

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

**Факультет прикладної математики
Кафедра системного програмування і
спеціалізованих комп'ютерних систем**

«На правах рукопису»
УДК 004.382.7

До захисту допущено:

Завідувач кафедри

_____ Віталій РОМАНКЕВИЧ

« ____ » _____ 2025 р.

Магістерська дисертація

на здобуття ступеня магістра

**за освітньо-професійною програмою «Системне програмування та
спеціалізовані комп'ютерні системи»**

зі спеціальності 123 «Комп'ютерна інженерія»

**на тему: «Засоби підвищення енергоефективності будівель на базі
концепції IoT»**

Виконав:

студент II курсу, групи КВ-41 мп

Орел Ілля Вадимович _____

Науковий керівник:

Професор кафедри СПіСКС, докт. техн. наук, професор

Романкевич Олексій Михайлович _____

Рецензент:

Засвідчую, що у цій магістерській дисертації
немає запозичень з праць інших авторів без
відповідних посилань.

Студент _____

Київ – 2025 року

**Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»**

Факультет прикладної математики

Кафедра системного програмування і спеціалізованих комп'ютерних систем

Рівень вищої освіти – другий (магістерський)

Спеціальність – 123 «Комп'ютерна інженерія»»

Освітньо-професійна програма «**Системне програмування та спеціалізовані комп'ютерні системи**»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Віталій РОМАНКЕВИЧ

«___» _____ 2024 р.

**ЗАВДАННЯ
на магістерську дисертацію студенту**

Орлу Іллі Вадимовичу

1. Тема дисертації «Засоби підвищення енергоефективності будівель на базі концепції IoT», науковий керівник дисертації професор кафедри СПіСКС, докт. техн. наук, професор Романкевич Олексій Михайлович, затверджені наказом по університету від «6» листопада 2025 р. №4836-с.
2. Термін подання студентом дисертації: 12 грудня 2025.
3. Об'єкт дослідження: процеси моніторингу та управління енергоефективністю будівель.
4. Предмет дослідження: програмно-апаратні засоби та методи побудови системи моніторингу та управління енергоспоживанням з використанням технологій інтернету речей.
5. Перелік завдань, які потрібно розробити:
 - 1) визначити проблему енергоефективності будівель та поняття «розумного» енергоспоживання;
 - 2) провести порівняльний аналіз існуючих пристроїв та систем інтернету речей, спрямованих на підвищення енергоефективності будівель, з метою виявлення недоліків та напрямів їх подальшого вдосконалення;
 - 3) визначити вимоги до системи розумного енергоспоживання;
 - 4) розробити архітектуру системи інтернету речей для моніторингу та управління енергоспоживанням;
 - 5) розробити структурну та функціональну схеми системи;

- 6) створити прототип системи;
- 7) розробити серверну та клієнтську частини програмного забезпечення;
- 8) протестувати роботу системи.

6. Орієнтовний перелік графічного (ілюстративного) матеріалу: презентація.

7. Орієнтовний перелік публікацій:

1) Романкевич О. М., Орел І. В. Система моніторингу та управління енергоспоживанням в приміщенні на основі технологій інтернету речей. *Прикладна математика та комп'ютинг ПМК 2025* : збірник тез доповідей XVIII наукової конференції магістрантів та аспірантів (м. Київ, лист. 2025 р.);

2) Романкевич О. М., Орел І. В. Архітектура IoT-системи розумного енергоспоживання. *Інноваційна наука: пошук відповідей на виклики сучасності* : збірник з матеріалами V міжнародної наукової конференції (м. Кривий Ріг, лист. 2025 р.).

9. Дата видачі завдання: 1 листопада 2024р.

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1.	Вивчення літератури за тематикою МД	11.12.2024	
2.	Розроблення та узгодження завдання	04.09.2025	
3.	Аналіз існуючих рішень за тематикою дослідження	15.09.2025	
4.	Розробка апаратного та програмного забезпечення системи	10.10.2025	
5.	Тестування розробленого апаратного та програмного забезпечення системи	10.11.2025	
6.	Підготовка та написання розділів МД	20.11.2025	
7.	Оформлення документації МД	30.11.2025	
8.	Попередній розгляд магістерської дисертації на кафедрі	02.12.2025	

Студент

Ілля ОРЕЛ

Науковий керівник

Олексій РОМАНКЕВИЧ

РЕФЕРАТ

Актуальність теми

Енергоефективність будівель має велике економічне, екологічне та соціальне значення. Це пов'язано з наявністю великої кількості старих будівель, постійним зростанням цін на енергоресурси, потребою знизити забруднення навколишнього середовища та одночасно забезпечити комфортне і безпечне життя людей.

Розвиток технологій інтернету речей відкриває нові підходи до управління енергоспоживанням будівель: інтелектуальні системи спроможні автономно моніторити та управляти енергоспоживанням у реальному часі, оптимізовувати роботу опалення, вентиляції, кондиціонування, освітлення тощо, і, як наслідок, зменшувати пікові навантаження в енергомережі, економити енерговитрати, підвищувати комфорт людей і навіть продуктивність праці. Крім того, з'являється можливість збирати та аналізувати дані для виявлення тенденцій та закономірностей, створення сценаріїв енергоспоживання.

Існуючі в сфері технологій інтернету речей рішення для підвищення енергоефективності будівель можна поділити на три групи: прості розумні енергозберігаючі пристрої, енергомонітори та складні системи управління розумним будинком. Кожна з цих груп має свої недоліки. Прості розумні пристрої є доступними, але націлені лише на один напрямок енергоспоживання і не реалізують моніторинг мікроклімату та енерговитрат. Енергомонітори виконують лише функції контролю за енергоспоживанням і теж не здійснюють моніторинг якості мікроклімату. Комплексні системи розумного будинку хоч і мають широкий функціонал, однак їх придбання, інтеграція та експлуатація коштують занадто дорого. Крім того, частина їх функціоналу часто виявляється надлишковою, а складний інтерфейс вимагає від користувача технічної обізнаності. Таким чином, для споживачів із помірними статками витрати на таку систему значно перевищують потенційну економію на енергетичних ресурсах, а отже стають невиправданими. Перелічені недоліки показують необхідність розробки рішень з повноцінним функціоналом та водночас більш доступних широкому колу споживачів.

Мета роботи

Створення доступної широкому колу споживачів ефективної IoT-системи моніторингу та управління енергоспоживанням у приміщенні з повноцінним аналітичним функціоналом, що поєднуватиме моніторинг енергоспоживання з відстеженням параметрів якості внутрішнього середовища для підтримки оптимального балансу між економією енергії та комфортом, і матиме простий, інтуїтивно зрозумілий інтерфейс.

Для досягнення поставленої мети вирішуються такі задачі:

- визначення проблеми енергоефективності будівель та поняття «розумного» енергоспоживання;
- порівняльний аналіз існуючих пристроїв та систем інтернету речей, спрямованих на підвищення енергоефективності будівель, з метою виявлення недоліків та напрямів їх подальшого вдосконалення;
- визначення вимог до системи розумного енергоспоживання;
- розробка архітектури системи інтернету речей для моніторингу та управління енергоспоживанням;
- проектування структурної та функціональної схем системи;
- створення прототипу системи;
- розробка серверної та клієнтської частин програмного забезпечення;
- тестування роботи системи.

Об'єкт дослідження – процеси моніторингу та управління енергоефективністю будівель.

Предмет дослідження – програмно-апаратні засоби та методи побудови системи моніторингу та управління енергоспоживанням з використанням технологій інтернету речей.

Методи дослідження

У роботі використовуються порівняльний аналіз, систематизація, аналогово-цифрове вимірювання електричних параметрів із фазовою корекцією, методи визначення показників мікроклімату (фотометричне, радіолокаційне, терморезистивне та інфрачервоне абсорбційне вимірювання), методи

моделювання, програмування, статистичного аналізу та візуалізації отриманих результатів.

Наукова новизна

Запропоновано структурну організацію системи моніторингу та управління енергоспоживанням, яка перевершує існуючі рішення тим, що поєднує такі переваги 4-рівневої архітектури, як низька вартість, простота розгортання та використання, з розширеними можливостями, які включають не лише моніторинг енергоспоживання, але й відслідковування параметрів якості внутрішнього середовища, зберігання великого обсягу статистичних даних без використання зовнішніх серверів, можливість заміни та калібрування датчиків, аналітичний функціонал та можливість налаштування сценаріїв використання електроприладів.

Практична цінність

Розроблено ефективну систему моніторингу та управління енергоспоживанням, яка спроможна збирати та зберігати у локальному сховищі дані про споживання електроенергії, параметри електромережі та якості внутрішнього середовища, автономно керувати навантаженням на основі визначених користувачем сценаріїв, представляти аналітику по зібраним даним у зручному форматі за допомогою веб-інтерфейсу для прийняття рішень щодо оптимізації енергоспоживання. Низька вартість впровадження та експлуатації системи робить її доступною широкому колу споживачів, а її використання у приміщеннях різного призначення підвищить їх енергоефективність.

Апробація результатів дисертації

Основні положення та результати роботи були представлені та обговорені на двох наукових конференціях, а саме: XVIII науковій конференції магістрантів та аспірантів «Прикладна математика та комп'ютинг» ПМК-2025 (Київ, 19-21 листопада 2025 р.) та V міжнародної наукової конференції «Інноваційна наука: пошук відповідей на виклики сучасності» (Кривий Ріг, 21 листопада 2025 р.).

Публікації

Основні результати досліджень викладені в 2 тезах доповідей у збірниках тез доповідей конференцій: Романкевич О. М., Орел І. В. Система моніторингу та управління енергоспоживанням в приміщенні на основі технологій інтернету речей. *Прикладна математика та комп'ютинг ПМК 2025* : збірник тез доповідей XVIII наукової конференції магістрантів та аспірантів, 19-21 лист. 2025 р. Київ : КПІ, 2025; та Романкевич О. М., Орел І. В. Архітектура IoT-системи розумного енергоспоживання. *Інноваційна наука: пошук відповідей на виклики сучасності* : збірник з матеріалами V міжнародної наукової конференції, 21 лист. 2025 р. Кривий Ріг : ГО «Міжнародний центр наукових досліджень», 2025.

Структура та обсяг роботи

Магістерська дисертація складається з вступу, трьох розділів, висновків по кожному розділу та загальних висновків по роботі в цілому, списку використаних літературних джерел (69 найменування). Повний обсяг дисертації – 152 сторінки, у тому числі 108 сторінок основного тексту, 52 рисунків, 8 таблиць.

Ключові слова

енергоефективність, інтернет речей, моніторинг, HVAC, освітлення, IEQ, BMS, ESP32, C++, JavaScript, SQLite3.

ABSTRACT

Relevance of the topic

Energy efficiency of buildings is of great economic, environmental and social importance. This is due to the presence of a large number of old buildings, the constant increase in energy prices, the need to reduce environmental pollution and at the same time ensure a comfortable and safe life for people.

The development of Internet of Things technologies opens up new approaches to managing energy consumption in buildings: intelligent systems are able to autonomously monitor and manage energy consumption in real time, optimize the operation of heating, ventilation, air conditioning, lighting, etc., and, as a result, reduce peak loads in the power grid, save energy costs, increase people's comfort and even labor productivity. In addition, it becomes possible to collect and analyze data to identify trends and patterns, and create energy consumption scenarios.

Existing solutions in the field of Internet of Things technologies for increasing the energy efficiency of buildings can be divided into three groups: simple smart energy-saving devices, energy monitors and complex smart home management systems. Each of these groups has its own drawbacks. Simple smart devices are affordable, but they are aimed at only one direction of energy consumption and do not implement monitoring of the microclimate and energy costs. Energy monitors perform only the functions of controlling energy consumption and also do not monitor the quality of the microclimate. Although complex smart home systems have wide functionality, their acquisition, integration and operation are too expensive. In addition, some of their functionality often turns out to be redundant, and the complex interface requires technical knowledge from the user. Thus, for consumers with moderate wealth, the costs of such a system significantly exceed the potential savings on energy resources, and therefore become unjustified. The listed shortcomings show the need to develop solutions with full functionality and at the same time more accessible to a wide range of consumers.

The purpose of the work

Creating an effective IoT system for monitoring and managing indoor energy consumption with full analytical functionality, accessible to a wide range of consumers, that will combine energy consumption monitoring with tracking of indoor environment quality parameters to maintain the optimal balance between energy savings and comfort, and will have a simple, intuitive interface.

To achieve the set goal, the following tasks are solved:

- defining the problem of energy efficiency of buildings and the concept of “smart” energy consumption;
- analysis of existing IoT devices and systems aimed at increasing the energy efficiency of buildings in order to identify shortcomings and areas for their further improvement;
- determining requirements for a smart energy system;
- development of an Internet of Things system architecture for monitoring and managing energy consumption;
- designing structural and functional schemes of the system;
- creating a system prototype;
- development of server and client parts of the software;
- testing the system operation.

The object of the research is the processes of monitoring and managing the energy efficiency of buildings.

The subject of the research is software, hardware and methods for building an energy consumption monitoring and management system using Internet of Things technologies.

Research methods

The work uses comparative analysis, systematization, analog-digital measurement of electrical parameters with phase correction, methods for determining microclimate indicators (photometric, radar, thermoresistive and infrared absorption measurements), methods of modeling, programming, statistical analysis and visualization of the obtained results.

Scientific novelty

A structural organization of an energy monitoring and management system is proposed, which surpasses existing solutions in that it combines the advantages of a 4-tier architecture, such as low cost, ease of deployment and use, with advanced capabilities that include not only energy monitoring, but also tracking of internal environment quality parameters, storing a large amount of statistical data without using external servers, the ability to replace and calibrate sensors, analytical functionality, and the ability to configure scenarios for using electrical appliances.

Practical value

An effective energy monitoring and management system has been developed that is capable of collecting and storing data on electricity consumption, power grid parameters, and indoor environment quality in a local storage, autonomously managing the load based on user-defined scenarios, and presenting analytics on the collected data in a convenient format using a web-interface for making decisions on optimizing energy consumption. The low cost of implementing and operating the system makes it accessible to a wide range of consumers, and its use in premises for various purposes will increase their energy efficiency.

Approbation of dissertation results

The main provisions and results of the work were presented and discussed at two scientific conferences, namely: XV III Scientific Conference of master's and postgraduate students "Applied Mathematics and Computing" PMK-2025 (Kyiv, November 19-21, 2025) and V international scientific conference "Innovative Science: Searching for Answers to Modern Challenges" (Kryvyi Rih, November 21, 2025).

Publications

The main results of the research are presented in 2 theses of reports in the collections of theses of conference reports: Romankevych O. M., Orel I. V. Indoor energy consumption monitoring and management system based on Internet of Things technologies. *Applied Mathematics and Computing PMK 2025* : collection of theses of reports of the XVIII scientific conference of master's and postgraduate students,

November 19-21, 2025 Kyiv: KPI, 2025; and Romankevych O. M., Orel I. V. Architecture of IoT-systems of smart energy consumption. *Innovative science: Searching for Answers to Modern Challenges* : collection of materials of the V international scientific conference, November 21, 2025. Kryvyi Rih: PO "International Center for Scientific Research", 2025.

Structure and scope of work

The master's thesis consists of an introduction, three chapters, conclusions for each chapter and general conclusions for the work as a whole, a list of used literary sources (69 titles). The full volume of the thesis is 152 pages, including 108 pages of the main text, 52 figures, 8 tables.

Keywords

energy efficiency, Internet of Things, monitoring, HVAC, lighting, IEQ, BMS, ESP 32, C ++, JavaScript, SQLite 3.

ЗМІСТ

ВСТУП	3
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ ІОТ-РІШЕНЬ ДЛЯ ПІДВИЩЕННЯ ЕНЕРГОЕФЕКТИВНОСТІ	5
1.1 Проблема енергоефективності будівель та поняття «розумного» енергоспоживання	5
1.2 Концепція інтернету речей.....	9
1.3 Використання технологій інтернету речей для підвищення енергоефективності будівель.....	11
1.4 Порівняльний аналіз існуючих пристроїв та систем інтернету речей, спрямованих на підвищення енергоефективності будівель	13
ВИСНОВКИ ДО РОЗДІЛУ 1	21
2 РОЗРОБКА СИСТЕМИ МОНІТОРИНГУ ТА УПРАВЛІННЯ ЕНЕРГОСПОЖИВАННЯМ.....	23
2.1 Вимоги до системи розумного енергоспоживання	23
2.2 Моделі архітектури та технології інтернету речей	27
2.3 Архітектура системи розумного енергоспоживання	35
2.4 Система управління базою даних.....	38
2.5 Вибір протоколу бездротового зв'язку.....	41
2.6 Апаратне забезпечення системи моніторингу та управління енергоспоживанням.....	44
2.6.1 Вибір мікроконтролера	44
2.6.2 Вибір модуля енергомоніторингу	50
2.6.3 Вибір датчика освітленості	54
2.6.4 Вибір датчика присутності людини	57
2.6.5 Вибір датчика температури.....	61
2.6.6 Вибір датчика вуглекислого газу	65
2.7 Проєктування функціональної схеми та створення прототипу системи	

моніторингу та управління енергоспоживанням	69
2.8 Програмне забезпечення системи моніторингу та управління енергоспоживанням.....	73
2.8.1 Огляд мов програмування та середовищ розробки для мікроконтролерів	73
2.8.2 Серверна частина програмного забезпечення.....	75
2.8.3 Огляд варіантів реалізації прикладного рівня системи.....	78
2.8.4 Клієнтська частина програмного забезпечення	79
ВИСНОВКИ ДО РОЗДІЛУ 2	83
3 ОСОБЛИВОСТІ ФУНКЦІОНУВАННЯ СИСТЕМИ.....	85
3.1 Алгоритм роботи системи.....	85
3.2 Робота з веб-інтерфейсом.....	86
ВИСНОВКИ ДО РОЗДІЛУ 3	97
ВИСНОВКИ.....	98
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	101
ДОДАТОК А ПРЕЗЕНТАЦІЯ	
ДОДАТОК Б ФРАГМЕНТИ ЛІСТИНГУ ПРОГРАМИ.....	
ДОДАТОК В ПУБЛІКАЦІЇ ЗА ТЕМОЮ ДИСЕРТАЦІЇ	

ВСТУП

Сучасний розвиток світової економіки супроводжується зростаючим попитом на енергоресурси та необхідністю скорочення викидів парникових газів. Будівлі та споруди є одними із найбільших споживачів енергії: на них припадає до 40% первинного енергоспоживання та значна частина викидів CO₂. Постійне зростання цін на енергоресурси, наявність застарілого житлового фонду, що витрачає у 2–3 рази більше енергії, ніж сучасні будівлі, потребують проведення масштабної модернізації житлових та громадських будівель для покращення їх енергозберігаючих характеристик. Підвищення енергоефективності стає не лише важливим фактором економії ресурсів та зниження витрат домогосподарств та організацій, а й умовою сталого розвитку та кліматичної безпеки.

Розвиток технологій інтернету речей (Internet of Things, IoT) відкриває нові підходи до управління енергоспоживанням будівель. Різні пристрої та системи IoT, що складаються з мережі датчиків та підключених приладів, можуть забезпечувати безперервний моніторинг параметрів внутрішнього середовища у приміщеннях, присутності людей та інших факторів. На основі цих даних інтелектуальні системи автоматизації будівель, націлені на моніторинг та управління енергоспоживанням у реальному часі, можуть оптимізувати роботу опалення, вентиляції, кондиціонування повітря, освітлення та інших інженерних систем.

Практика показує, що застосування IoT-технологій дає змогу знижувати річне енергоспоживання будівель на 15–40 %, зменшувати пікові навантаження в енергомережі, підвищувати комфорт перебування людей у приміщенні та навіть продуктивність праці. Використання аналітики великих даних та алгоритмів машинного навчання сприяє ранньому виявленню неполадок обладнання та попереджає нераціональні втрати енергії.

Незважаючи на значний потенціал, існуючі IoT-рішення мають цілу низку проблем. Ключовим негативним фактором, що перешкоджає поширенню енергоефективних технологій інтернету речей, є те, що комплексні системи автоматизації, які, на відміну від простих пристроїв, надають своїм споживачам

широкі та різнобічні функціональні можливості, мають дуже високу вартість. Це обмежує їх використання у звичайних житлових будинках та на об'єктах з обмеженим бюджетом. Крім того, ефективна експлуатація таких систем потребує кваліфікованого персоналу, який часто відсутній у невеликих організаціях та домогосподарствах. Доступні, прості енергозберігаючі IoT-пристрої не дозволяють вирішувати проблему енергоефективності комплексно і, як наслідок, дають недостатній енергозберігаючий результат.

Тому виникає необхідність у дослідженнях та розробках, спрямованих на створення системи інтернету речей, яка буде поєднувати в собі достатньо широкі функціональні можливості щодо підвищення енергоефективності будівель з доступністю широкому колу споживачів.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ ІОТ-РІШЕНЬ ДЛЯ ПІДВИЩЕННЯ ЕНЕРГОЕФЕКТИВНОСТІ

1.1 Проблема енергоефективності будівель та поняття «розумного» енергоспоживання

Енергетична ефективність будівлі – це показник, що характеризує обсяг енергії, необхідної для підтримки комфортного мікроклімату всередині будівлі [8]. Параметри енергетичної ефективності будівлі вимірюються як питоме енергоспоживання на квадратний метр відповідної площі [10].

Енергоефективність тісно пов'язана з економією енергоресурсів, та існує суттєва розбіжність між ними. Економія енергоресурсів (енергозбереження) – це зменшення споживання енергії за рахунок відмови або обмеження її використання, наприклад, можна вимкнути світло. У цьому випадку ми справді економимо електрику, але одночасно можемо втратити якусь можливість, наприклад, читати. Енергоефективність – це використання енергії більш раціонально, тобто виконання тієї ж роботи або отримання таких самих умов життя з меншими витратами ресурсів [13; 24]. Наприклад: замінити лампу розжарювання на світлодіодну.

Таким чином, енергозбереження – це обмеження споживання, а енергоефективність – це підвищення віддачі від кожної одиниці витраченої енергії. Підвищення енергоефективності будівлі означає використання меншої кількості енергії для досягнення таких самих належних умов для життєдіяльності людей. Тому при визначенні рівня енергоефективності треба не тільки встановити кількість спожитої енергії, але й оцінити параметри якості внутрішнього середовища (Indoor Environmental Quality, IEQ). IEQ – це комплексна оцінка умов усередині будівлі, що впливають на комфорт і здоров'я людей та складається з якості повітря, температурного комфорту, освітлення тощо [6; 44].

Енергоефективність будівель має велике значення через низку факторів, пов'язаних з економікою, екологією та соціальними аспектами [1; 2; 5; 13; 24].

По-перше, існує багато старих будівель, особливо житлових, що мають низькі стандарти теплоізоляції та енергоефективності. Ці будівлі споживають у 2-

3 рази більше енергії, ніж сучасні будівлі в розвинутих країнах. Це призводить до величезних втрат тепла та високих витрат на опалення. Модернізація існуючого житлового фонду на основі використання енергоефективних технологій інтернету речей могла б суттєво знизити споживання енергії та покращити умови проживання.

По-друге, суттєва залежність від імпорту енергоресурсів, таких як природний газ чи нафтопродукти, робить країну вразливою до змін на світових ринках та політичних чинників. Підвищення енергоефективності будівель може знизити потребу в енергоресурсах, скоротити залежність від імпорту та підвищити енергетичну безпеку країни.

По-третє, постійне зростання цін на енергоресурси робить комунальні послуги значним фінансовим навантаженням для населення. Підвищення енергоефективності будівель дозволить одночасно заощадити на оплаті комунальних послуг та покращити загальний рівень життя, оскільки вивільняє кошти для інших потреб домогосподарств.

По-четверте, споживання енергії у будинках пов'язане з викидами вуглекислого газу та інших парникових газів, особливо якщо використовуються корисні копалини. Зниження енергоспоживання сприяє зменшенню викидів, що робить внесок у боротьбу зі зміною клімату та допомагає знизити навантаження на навколишнє середовище. Дотримання стандартів ЄС у галузі енергоефективності покращить інвестиційну привабливість країни [1; 2; 13].

І нарешті, енергоефективні будинки забезпечують стабільний та комфортний мікроклімат. Це позитивно позначається на здоров'ї людей. Модернізація енергоефективності будинків також підвищує безпеку життя, оскільки впливає на загальну стійкість будівель та всіх комунікацій.

Таким чином, підвищення енергоефективності будівель – це не лише засіб знизити енерговитрати та покращити комфорт, а й шлях до енергетичної незалежності, сталого економічного зростання та дотримання міжнародних зобов'язань щодо екології. Все перераховане робить розробки у сфері підвищення енергоефективності приміщень не просто актуальними, а й вкрай необхідними.

Для того, щоб визначити можливі напрямки підвищення енергоефективності, розглянемо складові споживання енергії в будівлі будь-якого призначення (рисунок 1.1) [18; 25; 56]:

- опалення, вентиляція та кондиціювання повітря (Heating, Ventilation, and Air Conditioning, HVAC). Це найбільша стаття енерговитрат у більшості будівель (40-50% від загального енергоспоживання), особливо у регіонах з холодним кліматом;
- освітлення. Залежно від типу приміщення та систем освітлення, частка може суттєво змінюватись. У комерційних будинках освітлення становить більшу частину витрат (приблизно 20-30% енергоспоживання), тоді як у житлових воно може бути менше (до 10%);
- навантаження, що підключаються. Ця стаття витрат залежить від використання будівлі. В офісах значна кількість енергії споживається комп'ютерами, принтерами та іншою технікою і частка цих витрат становить 20-30% енергоспоживання, тоді як у житлових будинках ці енерговитрати припадають на побутову техніку (близько 15-20% загального енергоспоживання);
- гаряче водопостачання. Частка цих витрат у загальному енергоспоживанні близько 10-20%. Вона особливо висока в житлових та готельних приміщеннях, де потрібний постійний підігрів води для побутових потреб.



Рисунок 1.1 – Структура споживання енергії в будівлях

Ці цифри є орієнтовними і можуть відрізнятися залежно від конкретних умов експлуатації та типу будівлі. Для точного визначення частки видатків на кожен складову енергоспоживання проводять енергетичний аудит конкретного об'єкта та визначають клас енергетичної ефективності будівлі [3; 8-11].

Останні досягнення інформаційних технологій вивели поняття енергоефективності на новий рівень. З'явилося поняття інтелектуальної або «розумної» енергоефективності.

«Розумна» енергоефективність (її ще називають «активна ефективність») – це сучасний підхід до управління енергоспоживанням, який спирається на інформаційно-комунікаційні технології, такі як інтернет речей (Internet of Things, IoT), великі масиви даних (Big Data) тощо [40; 41; 51]. Суть у тому, що економія енергії досягається не лише за рахунок використання більш ефективного обладнання, але і завдяки розумному управлінню системами та процесами.

Концепція «розумної» енергоефективності робить акцент не просто на тому, щоб насос або кондиціонер були більше економічними, а ще й на тому, щоб вся система працювала ефективніше. Наприклад: «розумний» термостат знижує температуру, коли мешканців немає вдома; електромобіль заряджається вночі, коли навантаження на мережу та вартість електроенергії нижчі. На відміну від традиційної енергоефективності, яка пов'язана в основному з конструкцією та характеристиками обладнання, інтелектуальна енергоефективність використовує дані та автоматизацію, щоб постійно підтримувати оптимальний баланс між комфортом, витратами та енергоспоживанням.

Дослідження Американської ради з енергоефективної економіки (American Council for an Energy-Efficient Economy, ACEEE) засвідчили, що засоби інтернету речей, такі як датчики присутності та якості повітря, розумні лампи та вимикачі, засновані на веб-технологіях системи управління та віддаленого контролю, програмні засоби аналітики тощо, можуть дати до 40% економії енергетичних ресурсів (таблиця 1.1) [40; 51; 55; 57].

Таблиця 1.1 – Ефективність використання технологій IoT для підвищення енергоефективності будівель за даними Американської ради з енергоефективної економіки (ACEEE)

Призначення будівлі	Середня площа будівлі, м ²	Середнє енергоспоживання за рік, кВт·год	Середня економія енергії при використанні IoT технологій у порівнянні зі звичайною будівлею за рік	
			кВт·год	%
Освітня установа	9 290	190 000	20 900	11
Офіс	4 650	850 000	200 000	23
Готель	18 580	4200000	260 000	6
Лікувальна установа	11 150	7 900 000	1 400 000	18
Науково-дослідна лабораторія	6 500	980 000	390 000	40

1.2 Концепція інтернету речей

Хакім Кассімаллі та Едріан Мак'юен, автори книги «Designing the Internet of Things», виражають сутність інтернету речей за допомогою простої формули (рисунок 1.2): це поєднання фізичних об'єктів, сенсорів, контролерів, виконуючих механізмів та інтернету [48].

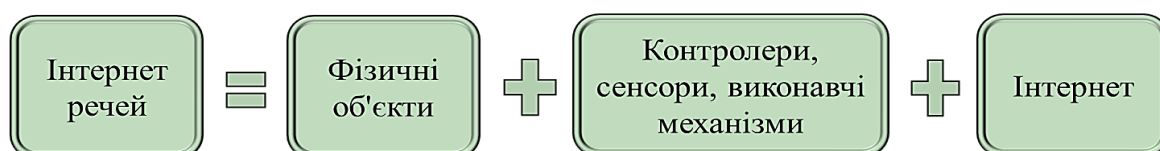


Рисунок 1.2 – Формула інтернету речей Касімалі Х. і Мак'юена Е.

Рекомендації Міжнародного союзу електрозв'язку МСЕ-Т Y.2060 (ITU-T Y.2060) дають значно ширше визначення поняття «інтернет речей» – це глобальна інфраструктура інформаційного суспільства, що забезпечує можливість надання

найсучасніших послуг через організацію зв'язку між різними фізичними та віртуальними речами на основі вже існуючих і тих, що в процесі розвитку, сумісних інформаційних і комунікаційних технологій [54].

Таким чином, концепція інтернету речей полягає в підключенні фізичних та віртуальних об'єктів у єдину мережу, де вони можуть взаємодіяти між собою, збирати інформацію та обмінюватися даними у режимі реального часу без прямої участі людини [31].

Згідно зі звітом IoT Analytics "State of IoT – Summer 2024" [66], до кінця 2023 року було 16,6 млрд. підключених пристроїв IoT (порівняно з 2022 роком зростання склало понад 15%). Очікується, що до кінця 2024 року ця кількість зросте ще на 13% до 18,8 млрд. (рисунок 1.3). Ця статистика враховує підключені вузли, пристрої або шлюзи, які концентрують кінцеві датчики, а не кожен окремий кінцевий датчик або виконавчий механізм.

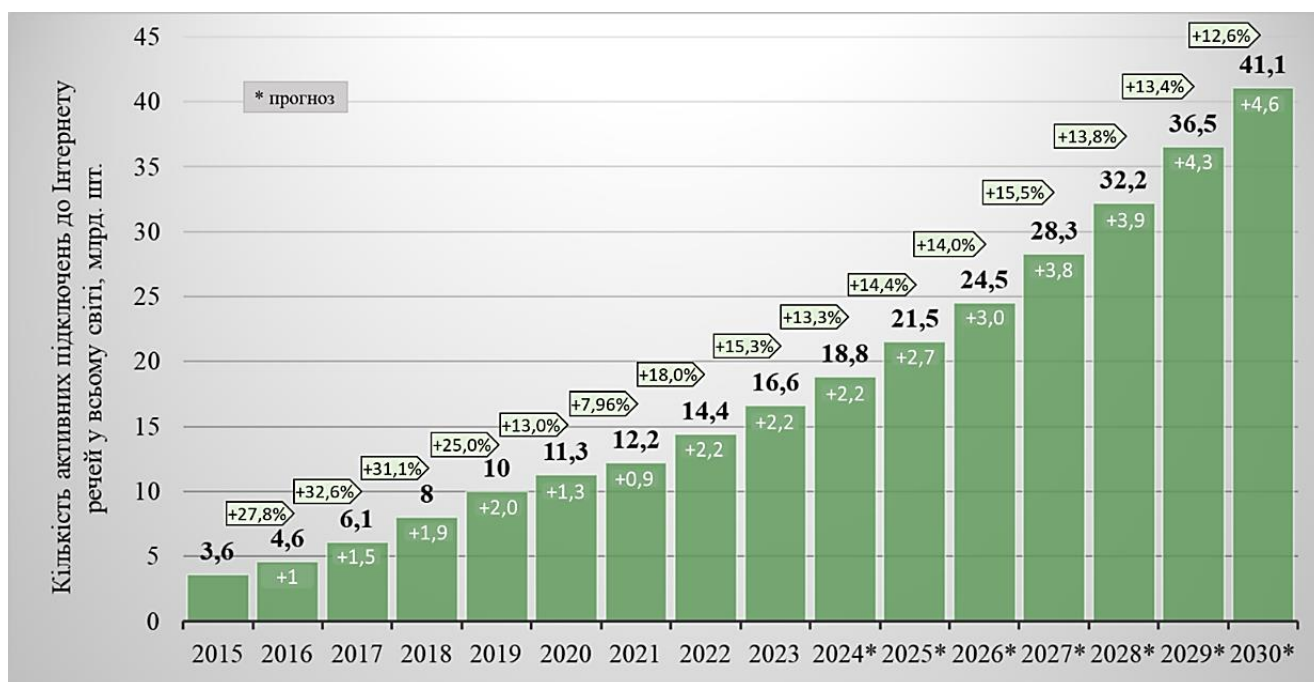


Рисунок 1.3 – Розвиток світового ринку інтернету речей за оцінкою IoT Analytics [64-66]

Загалом прогнозується, що щорічний приріст кількості підключених пристроїв інтернету речей становитиме 3-4 млрд. і в 2030 році буде понад 40 млрд.

При цьому корпоративні користувачі спланували у 2025-2030 роках збільшувати свої витрати на технології IoT на 14% кожного року і у 2030 році довести загальну суму інвестицій у сферу інтернету речей до 649 млрд. доларів [65]. Ця статистика підтверджує, що інтерес до досліджень та розробок у цій галузі не тільки не згасає, а й набирає обертів.

Одним із найперспективніших напрямків використання технологій інтернету речей є підвищення енергоефективності будівель. Технології IoT здатні значно підвищити енергоефективність приміщень за рахунок автоматизації управління, моніторингу та аналізу енергоспоживання у реальному часі.

1.3 Використання технологій інтернету речей для підвищення енергоефективності будівель

Розглянута вище концепція «розумної» енергоефективності передбачає, що система інтернету речей, спрямована на підвищення енергоефективності будівель, повинна не просто охоплювати всі складові енергоспоживання (HVAC, освітлення, різні інші навантаження), а забезпечувати ефективне управління енергоспоживанням [39]. Отже, можна виділити такі напрямки використання технологій IoT для підвищення енергоефективності будівель (рисунок 1.4):



Рисунок 1.4 – Напрямки використання технологій IoT для підвищення енергоефективності будівель

1. Управління опаленням, вентиляцією та кондиціонуванням повітря (HVAC). IoT-датчики температури та якості повітря можуть відстежувати умови в кожному приміщенні та передавати дані системі управління HVAC. На основі цих даних система може автоматично регулювати температуру та рівень вентиляції, підлаштовуючись під поточні умови, присутність та кількість людей. Це не тільки знижує енергоспоживання, а й покращує здоров'я та комфорт користувачів.

2. Інтелектуальне освітлення. Датчики руху та освітленості можуть включати, виключати та регулювати світло в залежності від наявності людей та рівня денного світла. Це скорочує енерговитрати на освітлення, особливо у великих офісах та громадських будинках, де світло часто горить постійно. IoT-системи можуть також налаштовувати інтенсивність і колірну температуру світла для підвищення комфорту, що додатково знижує витрати.

3. Управління побутовим та офісним обладнанням. IoT-платформи можуть автоматично відключати електроприлади, що не використовуються за відсутності людей, наприклад, комп'ютери, принтери, кавоварки тощо, що особливо важливо для офісів і підприємств.

4. Автоматизований енергоменеджмент у реальному часі. IoT може забезпечити постійний моніторинг усіх систем будівлі, оперативно виявляючи неполадки через аномальне енергоспоживання.

5. Аналітика, оптимізація та прогнозування енергоспоживання. IoT-системи, оснащені інструментами аналітики та машинного навчання, можуть:

- збирати дані про споживання енергії кожним приладом та системою будівлі. Це дозволяє виявляти неефективні пристрої та режими роботи. На основі цих даних можуть бути вироблені рекомендації щодо оптимізації роботи обладнання, розроблені ефективні сценарії енергоспоживання;

- прогнозувати майбутні потреби на основі аналізу даних про енергоспоживання. Це допомагає планувати навантаження та впроваджувати превентивні заходи для запобігання енергетичним перевантаженням, що зменшує витрати на електроенергію, запобігає аваріям обладнання та подовжує його термін служби.

Для досягнення максимального ефекту енергозбереження система інтернету речей повинна охоплювати всі ці напрямки.

1.4 Порівняльний аналіз існуючих пристроїв та систем інтернету речей, спрямованих на підвищення енергоефективності будівель

Дослідження існуючих технологій у сфері розумного енергоспоживання дозволило виділити декілька рішень, які можна охарактеризувати як найбільш розповсюджені та типові, за якими можна проаналізувати сучасний стан та тенденції розвитку обраної галузі. Розглянемо їх функціонал та особливості.

Велику долю ринку займають так звані «розумні» пристрої – розумні лампи, розумні розетки, розумні термостати тощо. Вони всі мають схожі функціональні можливості.

Розумна лампа WiZ A60 White 806 lm E27 (зовнішній вигляд зображений на рисунку 1.5) є типовим представником простого «розумного» пристрою, що спрямований на енергозбереження [23; 58].



Рисунок 1.5 – Зовнішній вигляд лампи WiZ A60 White 806 lm E27

Для початку роботи її треба просто вкрутити у патрон E27 та підключити до стандартної електромережі. Взаємодія з нею відбувається через Wi-Fi за допомогою мобільного додатку WiZ. Серед запропонованих функцій даного пристрою є налаштування температури світла у діапазоні від 2700 до 6500 K, зниження яскравості, вимикання по налаштованому інтервалу та інтеграція із голосовими помічниками на кшталт Google Assistant, Amazon Alexa тощо.

У використанні автономних розумних пристроїв, таких як WiZ A60 White 806 lm E27, у якості рішення проблеми розумного енергоспоживання є наступні недоліки. По-перше, вони є лише виконавчими пристроями, і не забезпечують можливості зчитування та накопичення даних про параметри IEQ та параметрів електромережі. Тобто, такі розумні виконавчі пристрої в першу чергу спрямовані на комфорт у використанні, а не на розумне енергоспоживання із детальною аналітикою. Вони можуть працювати в екосистемі із розумними маршрутизаторами, які в свою чергу здатні зберігати історію використання кожного підключеного пристрою, але для вимірювання та накопичення показників IEQ та параметрів електромережі необхідно купляти відповідні модулі вимірювання, що сумісні із екосистемою. Загальна вартість такої системи буде високою, особливо в масштабах немалих об'єктів. По-друге, кожен з таких пристроїв націлений лише на один напрямок споживання енергії в будівлі: розумні лампи – на освітлення, термостат – на температуру у приміщенні. Вони не вирішують проблему енергоефективності комплексно.

Наступною групою пристроїв є енергомонітори, основною функцією яких є відстеження енергоспоживання. Розглянемо декілька з них.

На рисунку 1.6 зображений зовнішній вигляд енергомонітора Shelly Pro 3EM від компанії Shelly Group SE [62].



Рисунок 1.6 – Зовнішній вигляд енергомонітора Shelly Pro 3EM

До недоліків даного рішення можна віднести те, що цей прилад вимірює лише споживання електроенергії. На основі таких вимірів неможливо зробити висновки про доцільність та ефективність використання виконавчих приладів. Можна лише відстежити тенденції по споживанню енергії. Також, він сам по собі не має можливості виводу зчитаних даних і потребує додаткової зовнішньої платформи для подання аналітики у зручній для користувача формі.

Далі розглянемо Sense Energy Monitor (рисунок 1.7) [37].



Рисунок 1.7 – Зовнішній вигляд енергомонітора Sense Energy Monitor

Як і попередній аналог, даний пристрій встановлюється в електрощиті та вимірює спожиту електроенергію. Для нього створено спеціальний мобільний додаток, що забезпечує можливість користувача аналізувати споживання електроенергії. Особливістю Sense Energy Monitor є функція у додатку із розпізнаванням підключених пристроїв за такими параметрами, як форма хвилі струму, частота, пусковий стрибок та тривалість роботи. Ця функція працює на основі алгоритмів машинного навчання.

Найголовнішим недоліком, як і у попереднього рішення, є те, що даний інструмент відслідковує лише енергоспоживання, а тому не надає інформації для проведення ретельного аналізу роботи пристроїв. Особлива функція із розпізнаванням приладів за сигнатурами працює через закриту хмару і для роботи застосунку комп'ютер або телефон користувача обов'язково повинен мати доступ до інтернету. Також, сам процес навчання займає декілька тижнів і залежить від

частоти використання виконуючих пристроїв. Не менш важливим негативним фактором є достатньо висока ціна.

Ще одним прикладом енергомонітору є EmonPi2 від проєкту The OpenEnergyMonitor [26]. Зовнішній вигляд даного пристрою зображений на рисунку 1.8.

EmonPi2 побудовано на основі мікрокомп'ютера Raspberry Pi. Він має можливість розширення свого функціоналу шляхом встановлення додаткових датчиків та модулів, що повинні бути придбані окремо. Без розширень він виконує роль звичайного лічильника електроенергії. Встановлювати EmonPi2 можна у будь-якому місці в будинку, підключивши до нього дроти з трансформаторними кліпсами, одягненими на фазні дроти в електрощиті, та підключивши до мережі датчик напруги. Що важливіше, він може приймати дані від інших пристроїв, зберігати їх та показувати накопичені дані у зручному для користувача вигляді за допомогою онлайн-засобів EmonCMS. Іншими пристроями, що можуть надсилати EmonPi2 свої дані, можуть виступати сумісні датчики газу, температури, вологості та багато інших.



Рисунок 1.8 – Зовнішній вигляд енергомонітора EmonPi2

До недоліків, в першу чергу, варто віднести доволі високу ціну. EmonPi2 хоч і підходить для створення відповідної системи із комплексним аналізом використання енергії та інших ресурсів, але додатковий функціонал окрім обліку електроенергії передбачає придбання відповідних пристроїв, що суттєво збільшує вартість такої системи.

Ще однією групою рішень у сфері розумного енергоспоживання є комплексні системи управління розумними будинками. Розглянемо дві з них – Desigo CC від німецького конгломерату Siemens AG та EcoStruxure Building Operation від французької енергомашинобудівної корпорації Schneider Electric.

Система Desigo CC – це інтегрована платформа управління будівлею, що забезпечує централізований моніторинг і контроль усіх технічних систем будівлі [20]. У цю систему входять контроль опалення, вентиляції, кондиціонування, автоматизація освітлення, керування денним світлом, підтримка сценаріїв, збір та аналіз даних лічильників, енергетична аналітика, створення звітів, виявлення перевитрат. Також вона підтримує можливість інтеграції сторонніх IoT-пристроїв та підтримує відкриті протоколи, такі як BACnet, MQTT, KNX, SNMP тощо. Аналітика реалізована шляхом збору даних з підсистем, накопичення цих даних у SQL базі даних та візуалізації за допомогою графіків та діаграм. Зовнішній вигляд додатку, створеного для взаємодії з системою Desigo CC, зображено на рисунку 1.9.

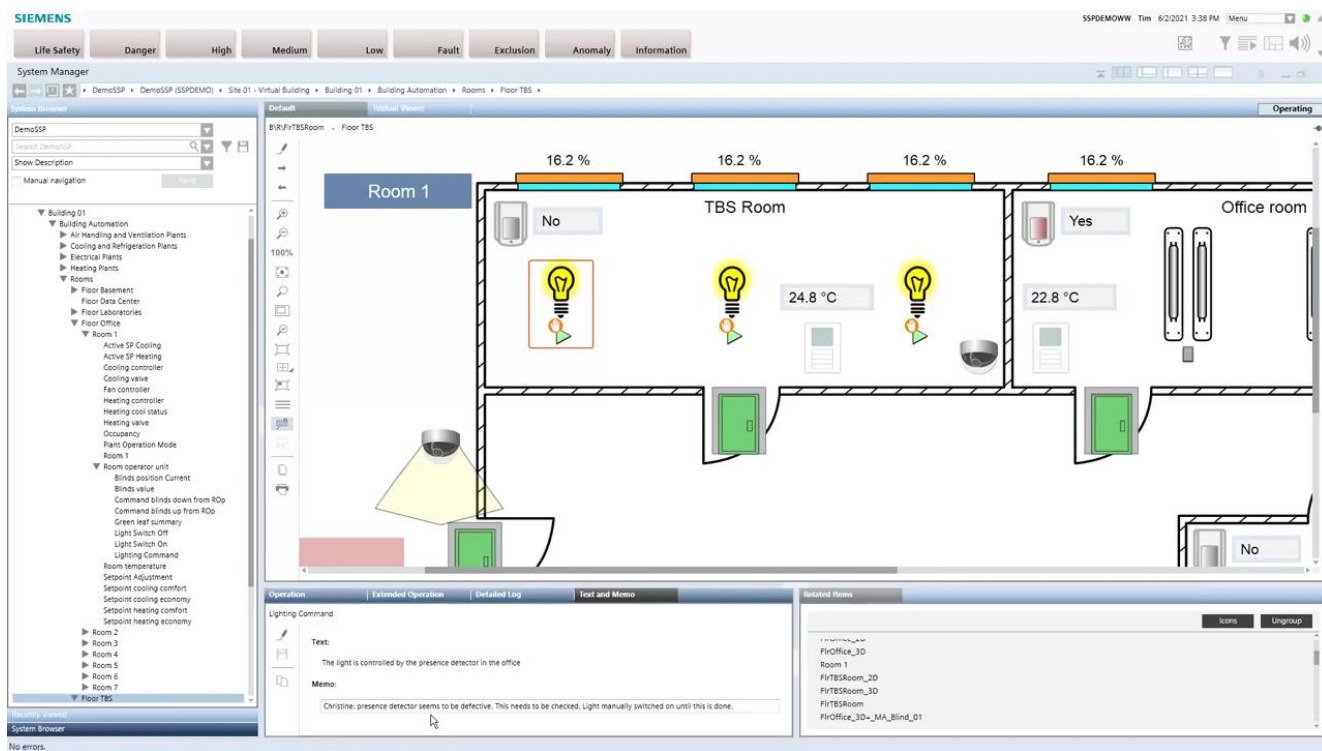
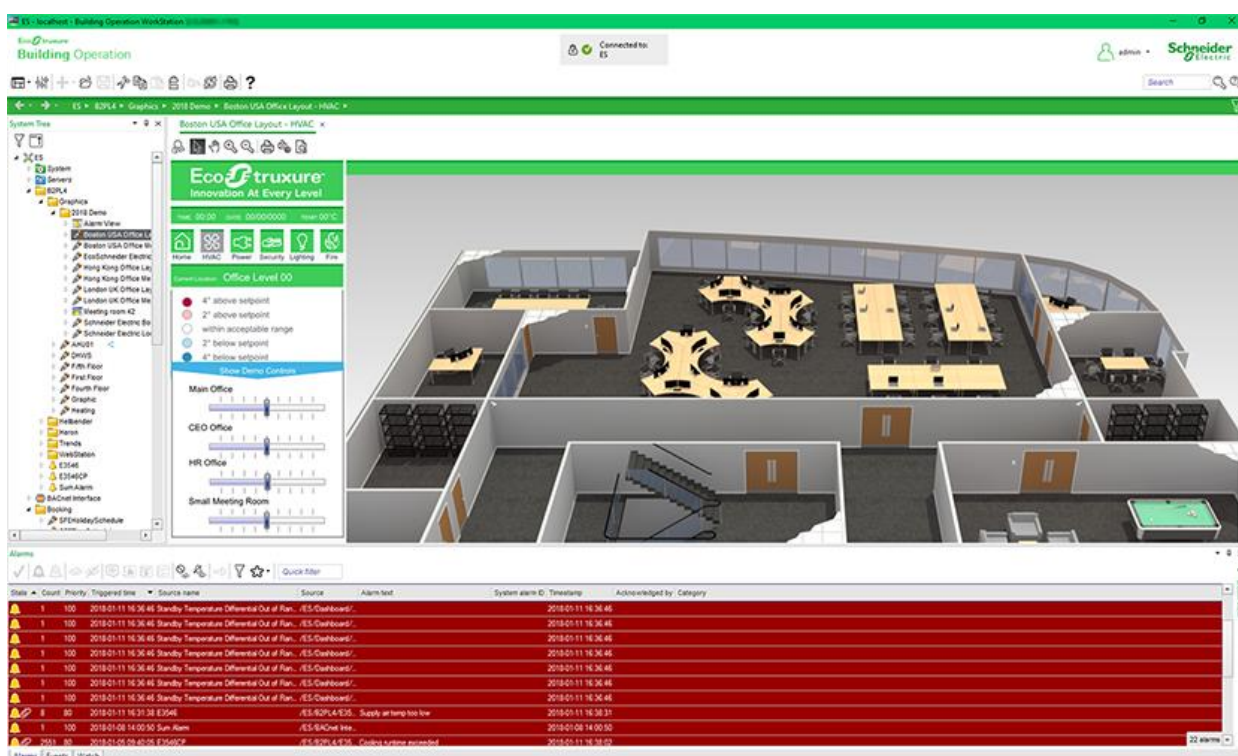


Рисунок 1.9 – Зовнішній вигляд додатку Desigo CC

Навіть враховуючи широкий функціонал даної системи, головним недоліком Desigo CC є вартість придбання та встановлення у будівлі. У мінімальній конфігурації на невеликому об'єкті запровадження Desigo CC буде коштувати приблизно 35 тисяч доларів. У великій будівлі ціна може вирости до 250 тисяч доларів. Це робить таку систему недоступною для широкого кола споживачів.

EcoStruxure Building Operation – це відкрита та масштабована система BMS наступного покоління, що дозволяє централізовано моніторити, керувати та оптимізувати інфраструктуру будівлі [21]. Вона інтегрує дані від сенсорів, виконавчих пристроїв та підсистем (HVAC, освітлення, безпека, електроживлення) в єдину платформу. Система підтримує зберігання трендів показників, локальну роботу без інтернету та можливість підключення до хмарних сервісів для аналітики. Завдяки модульній архітектурі вона підходить як для невеликих офісів, так і для великих бізнес-кампусів. Особливий акцент зроблено на кібербезпеці та стандартизованих протоколах інтеграції (BACnet, Modbus, OPC). Інтерфейс дозволяє користувачу переглядати історичні дані та аналітику. Зовнішній вигляд додатку, створеного для взаємодії з даною системою, зображений на рисунку 1.10.



Аналогічно до попередньої системи, EcoStruxure Building Operation має зовелику ціну впровадження. Вартість стандартного рішення становить 50-150 тисяч доларів, в залежності від кількості пристроїв, модулів та інтегрованих підсистем. Великі комерційні або промислові проєкти можуть перевищувати 500 тисяч доларів за своєю вартістю. Така система не розрахована на пересічного споживача.

У таблиці 1.2 наведено основні властивості усіх розглянутих пристроїв та систем інтернету речей, орієнтованих на енергозбереження.

Таблиця 1.2 – Основні характеристики розглянутих рішень розумного енергоспоживання

Рішення	Параметри, що вимірюються		Зберігання даних		Ціна, \$
	Споживання електроенергії	IEQ	Період	Інтервал	
WiZ A60 White 806 lm E27	—	—	—	—	7
Shelly Pro 3EM	+	—	60 днів	1 хвилина	102
Sense Energy Monitor	+	—	у хмарі без обмежень, локально 6 годин	1 секунда	300 – 350
EmonPi2	+	треба купувати додатково	налаштовуються		275 – 300
Desigo CC	+	+	налаштовуються		35 – 50 тис.
EcoStruxure Building Operation	+	+	налаштовуються		60 – 500 тис.

Таким чином, дослідження існуючих пристроїв та систем інтернету речей, спрямованих на підвищення енергоефективності будівель, дає можливість зробити низку висновків.

По-перше, значну долю ринку займають прості доступні розумні пристрої, які націлені лише на один напрямок енергоспоживання, наприклад, лише на

освітлення або лише на опалення, та не реалізують моніторинг якості внутрішнього середовища у приміщенні (IEQ) та енерговитрат із збереженням показників для подальшої аналітики. Більшість пристроїв із сегменту, що доступний широкому колу споживачів, потребують додаткового придбання маршрутизаторів або контролерів для зберігання зчитаних показників. Такі пристрої не спроможні вирішувати проблему підвищення енергоефективності комплексно.

По-друге, присутні на ринку енергомонітори виконують лише функції контролю за енергоспоживанням і теж, як і перша група пристроїв, не здійснюють моніторинг якості внутрішнього середовища. Крім того, у своїй більшості, енергомонітори залежать від доступу до хмари, а деякі навіть додатково потребують оформлення підписки на використання функцій пов'язаних із віддаленим сервером.

Третя група – це професійні системи розумного будинку, що створюються під замовлення. Хоча вони надають користувачу широкі функціональні можливості, їх придбання та інтеграція коштують занадто дорого. Крім того, частина функціоналу професійних систем розумного будинку часто виявляється надлишковою, а складний інтерфейс вимагає від користувача певного рівня технічної обізнаності. Використання такої системи потребує постійної технічної підтримки, а це додаткові видатки на експлуатацію будинка. Таким чином, для широкого кола споживачів із середніми статками витрати на таку систему значно перевищують отриману економію на енергетичних ресурсах та будуть невиправданими.

Перелічені недоліки існуючих пристроїв та систем інтернету речей у галузі енергоефективності підкреслюють гостру потребу в розробці економічно доцільного, доступного широкому колу споживачів рішення IoT з достатньо широким функціоналом, який би поєднував моніторинг енергоспоживання з відстеженням якості внутрішнього середовища й аналітикою, та мав простий, інтуїтивно зрозумілий інтерфейс.

ВИСНОВКИ ДО РОЗДІЛУ 1

Енергетична ефективність характеризує обсяг енергії, необхідної для підтримки комфортного мікроклімату всередині будівлі та визначається як питоме споживання енергії на квадратний метр відповідної площі. Цей показник – це не просто зменшення споживання енергії, це використання енергії більш раціонально для виконання тієї ж роботи або отримання таких самих умов життя.

Енергоефективність будівель має велике економічне, екологічне та соціальне значення. Це пов'язано з наявністю великої кількості старих будівель, залежністю від імпорту енергоресурсів, постійним зростанням цін на них, потребою знизити забруднення навколишнього середовища та одночасно забезпечити комфортне і безпечне життя людей.

Стрімкий розвиток інформаційних технологій став новим етапом у сфері енергоефективності. З'явилося поняття «розумна» енергоефективність (або «активна ефективність») – це сучасний підхід до управління енергоспоживанням, який спирається на інформаційно-комунікаційні технології, такі як IoT, Big Data тощо. Суть у тому, що економія енергії досягається не лише за рахунок використання більш ефективного обладнання, але і завдяки розумному управлінню системами та процесами.

Технології інтернету речей охоплюють всі аспекти енергоефективності будівель: управління опаленням, вентиляцією та кондиціонуванням повітря; інтелектуальне освітлення; управління побутовим та офісним обладнанням; автоматизований енергоменеджмент у реальному часі; аналітика, оптимізація та прогнозування енергоспоживання. Для досягнення максимального ефекту енергозбереження система інтернету речей повинна охоплювати всі ці аспекти. Це дасть можливість комплексно реалізувати необхідний функціонал системи.

Проведене дослідження найбільш розповсюджених та типових рішень в сфері використання технологій інтернету речей для підвищення енергоефективності будівель, за якими можна охарактеризувати сучасний стан та тенденції розвитку обраної галузі, дає можливість зробити низку висновків.

Існуючі IoT-рішення можна поділити на три групи: прості розумні енергозберігаючі пристрої, енергомонітори та складні великі системи управління розумним будинком. Кожна з цих груп має свої недоліки.

Прості розумні пристрої доступні широкому колу споживачів, але не спроможні вирішувати проблему підвищення енергоефективності комплексно, тому що націлені лише на один напрямок енергоспоживання, наприклад, лише на освітлення, і не реалізують моніторинг якості внутрішнього середовища у приміщенні (IEQ) та енерговитрат із збереженням показників для подальшої аналітики. Крім того, більшість таких пристроїв потребують додаткового придбання маршрутизаторів або контролерів для зберігання зчитаних показників.

Енергомонітори виконують лише функції контролю за енергоспоживанням і теж, як і перша група пристроїв, не здійснюють моніторинг якості внутрішнього середовища. Крім того, у своїй більшості, енергомонітори залежать від доступу до хмари, а деякі навіть додатково потребують оформлення підписки на використання функцій, пов'язаних із віддаленим сервером.

Професійні системи розумного будинку хоч і надають користувачу широкі функціональні можливості, їх придбання та інтеграція коштують занадто дорого. Експлуатація такої системи потребує постійної технічної підтримки, а це додаткові витрати. Крім того, частина функціоналу професійних систем розумного будинку часто виявляється надлишковою, а складний інтерфейс вимагає від користувача певного рівня технічної обізнаності. Таким чином, для широкого кола споживачів із помірними статками витрати на таку систему значно перевищують отриману економію на енергетичних ресурсах та будуть невиправданими.

Перелічені недоліки існуючих пристроїв та систем інтернету речей у галузі енергоефективності виявили потребу в розробці економічно доцільного, доступного широкому колу споживачів рішення IoT з достатньо широким функціоналом, який би поєднував моніторинг енергоспоживання з відстеженням якості внутрішнього середовища й аналітикою, та мав простий, інтуїтивно зрозумілий інтерфейс.

2 РОЗРОБКА СИСТЕМИ МОНІТОРИНГУ ТА УПРАВЛІННЯ ЕНЕРГОСПОЖИВАННЯМ

2.1 Вимоги до системи розумного енергоспоживання

Першим кроком у розробці будь-якої системи інтернету речей є визначення ключових вимог до неї. У нашому випадку це: масштабованість та гнучкість архітектури, здатність до інтеграції та сумісність, достатня потужність апаратного забезпечення, можливість автономної роботи, надійність, точність та оперативність моніторингу, безпека, зручність використання, економічність.

Масштабованість означає здатність системи працювати стабільно та ефективно при збільшенні кількості підключених пристроїв (датчиків, лічильників, модулів керування тощо) і обсягів даних, не потребуючи при цьому радикальної модернізації інфраструктури.

Гнучкість архітектури полягає у здатності системи адаптуватися до нових вимог користувача і змін умов експлуатації без повного перепроєктування. Це досягається за рахунок модульної побудови, коли кожен функціональний блок (моніторинг HVAC, освітлення, електромережі тощо) може додаватися або модернізуватися окремо. Це дозволяє швидко інтегрувати нові сенсори та виконавчі пристрої, оновлювати програмне забезпечення, підключати аналітичні сервіси чи змінювати алгоритми керування відповідно до потреб або вподобань користувачів.

Поєднання масштабованості та гнучкості робить IoT-систему сприятливою до розвитку технологій і зростання потреб без надмірних витрат на оновлення та підтримку, що забезпечує її довготривалу ефективність і придатність.

Сумісність забезпечує можливість взаємодії пристроїв різних виробників і поколінь у межах єдиної мережі. Це важливо для систем, що працюють у будівлях, де вже встановлено певне обладнання – системи HVAC, освітлення, лічильники електроенергії, засоби безпеки. Використання стандартних протоколів зв'язку, таких, наприклад, як HTTP/HTTPS, дозволяє уникнути проблем із несумісністю та забезпечує безперебійну комунікацію між усіма компонентами системи.

Інтеграція полягає у здатності системи об'єднувати дані з різних джерел і керувати різнорідними підсистемами через єдину платформу. Завдяки цьому забезпечується узгоджена робота всіх елементів – від сенсорів температури, освітленості та концентрації CO₂ до систем автоматичного керування освітленням, HVAC та електроприладами.

Високий рівень інтеграції та сумісності робить систему відкритою до майбутніх оновлень і розширень, зменшує витрати на модернізацію та продовжує термін її ефективної експлуатації. Це дозволяє будівлям поступово переходити на більш розумне й енергоефективне керування без необхідності повної заміни наявної інфраструктури.

Вимога щодо забезпечення необхідної потужності апаратної складової (процесора, оперативної та постійної пам'яті) є однією з ключових для IoT-системи, оскільки від неї залежить стабільність роботи всієї платформи та можливість ефективної обробки даних. Частота центрального процесора визначає швидкість обробки інформації, виконання алгоритмів керування та роботи аналітичних модулів. Не менш важливою є наявність достатнього обсягу оперативної пам'яті (RAM) для обробки поточкових даних у режимі реального часу та постійної пам'яті (Flash, EEPROM або SD-карти) для збереження налаштувань, локальних журналів і даних, які генерують сенсори й пристрої. Це дозволяє уникнути втрати інформації у випадках тимчасових збоїв зв'язку.

Автономність означає здатність системи виконувати свої функції без постійного підключення до мережі Інтернет або зовнішніх серверів. Це особливо важливо для об'єктів, де зв'язок може бути нестабільним або перериватися. У таких випадках система повинна зберігати та обробляти дані локально, використовуючи вбудовану пам'ять чи зовнішні носії, а також продовжувати керувати обладнанням за заданими алгоритмами.

Надійність системи полягає у здатності працювати цілодобово та безперервно, забезпечуючи стабільний збір і передачу даних, а також стійкість до відмов обладнання. Для цього важливими є використання якісних апаратних компонентів, наявність захисту від перевантажень і збоїв живлення, а також

резервування критичних вузлів – наприклад, застосування безперебійних джерел живлення або дублюючих датчиків. Крім того, надійність передбачає стійкість до зовнішніх впливів: система повинна коректно функціонувати в широкому діапазоні температур і вологості, а датчики мають зберігати точність вимірювань протягом тривалого часу без частого обслуговування.

Точність моніторингу визначає, наскільки достовірно система вимірює параметри середовища та енергоспоживання. Високоточні сенсори температури, концентрації CO₂, рівня освітленості чи електричних характеристик (напруги, струму, потужності) дають можливість отримувати дані з мінімальною похибкою. Наприклад, для контролю мікроклімату у будівлях важливо забезпечити точність вимірювання температури на рівні $\pm 0,5$ °C. Висока точність вимірювань дозволяє уникати помилкових рішень, запобігати перевитратам енергії та підтримувати оптимальні умови роботи обладнання і комфорт користувача.

Оперативність моніторингу стосується швидкості збору, обробки та передачі даних. Система повинна працювати в реальному часі. Частота оновлення даних має становити не більше 0,5-1 хвилини, що дає змогу своєчасно виявляти відхилення від норми та швидко виконувати коригувальні дії, наприклад змінювати режим роботи HVAC або вимикати непотрібні електроприлади.

Однією з найважливіших вимог до IoT-системи є безпека, оскільки сучасні системи інтернету речей збирають і передають великі обсяги даних: параметри мікроклімату, інформацію про споживання енергії, а іноді й персональні відомості користувачів та дані про доступ до приміщень. Система повинна використовувати сучасні алгоритми шифрування, а також надійні механізми аутентифікації та авторизації користувачів і пристроїв. Система має збирати лише необхідну інформацію та повідомляти користувачів про мету її використання.

Зручність використання є однією з визначальних вимог до IoT-системи. Система має бути доступною як для фахівців, так і для людей без технічної підготовки. Це передбачає простий та інтуїтивно зрозумілий інтерфейс, який дозволяє отримувати інформацію про споживання енергії, умови мікроклімату та стан обладнання, а також легко змінювати налаштування без потреби у спеціальних

знаннях. Важливим елементом є графічна візуалізація даних – відображення показників у вигляді діаграм, графіків і таблиць, що спрощує аналіз та допомагає швидко оцінювати ситуацію. Зручність використання також включає простоту встановлення та налаштування системи. Все має бути максимально автоматизовано й супроводжуватися покроковими інструкціями. Не менш важливим є забезпечення дистанційного доступу до управління та моніторингу. Користувачі повинні мати можливість контролювати параметри мікроклімату та енергоспоживання і налаштовувати роботу обладнання через зручні мобільні додатки або веб-інтерфейси з будь-якого місця та у будь-який час.

Таким чином, зручність використання визначає комфорт і швидкість взаємодії людини із системою, робить її доступною широкому колу користувачів і сприяє підвищенню ефективності управління енергоспоживанням завдяки своєчасним та обґрунтованим рішенням.

Економічність і довговічність є важливими вимогами до IoT-системи, оскільки вони безпосередньо впливають на вартість впровадження та подальшу експлуатацію системи, а також визначають її привабливість для широкого кола користувачів у довгостроковій перспективі.

Економічність системи передбачає оптимальне співвідношення вартості системи та її функціональних можливостей. При проєктуванні слід обирати доступні й надійні компоненти, які забезпечують необхідну продуктивність і точність, але не потребують надмірних фінансових витрат. Важливу роль відіграє використання обладнання з низьким енергоспоживанням, що дозволяє суттєво зменшити витрати на експлуатацію та обслуговування системи. Застосування бездротових технологій зв'язку й відсутність необхідності у складній серверній інфраструктурі також знижують витрати на встановлення й підтримку.

Довговічність означає здатність системи ефективно працювати протягом тривалого часу без частих замін чи оновлень обладнання. Для цього важливо використовувати сучасні та технічно прогресивні компоненти, які мають тривалий термін служби та підтримуються виробниками. Це гарантує можливість оновлення

програмного забезпечення й сумісність з новими пристроями у майбутньому без необхідності повної заміни системи.

2.2 Моделі архітектури та технології інтернету речей

Архітектура IoT-проекту – це “скелет” системи, це її функціональна структура, що окреслює, які компоненти використовуються та як вони взаємодіють для досягнення кінцевих цілей системи [38; 42; 68].

Ключовими характеристиками архітектури інтернету речей є:

- скільки рівнів має архітектура;
- які функції та конкретні завдання виконує кожен елемент архітектури;
- як здійснюється взаємозв’язок між елементами системи.

Розглянемо існуючі підходи до формування архітектури систем інтернету речей (рисунок 2.1) [4; 33; 35; 38; 46; 68].

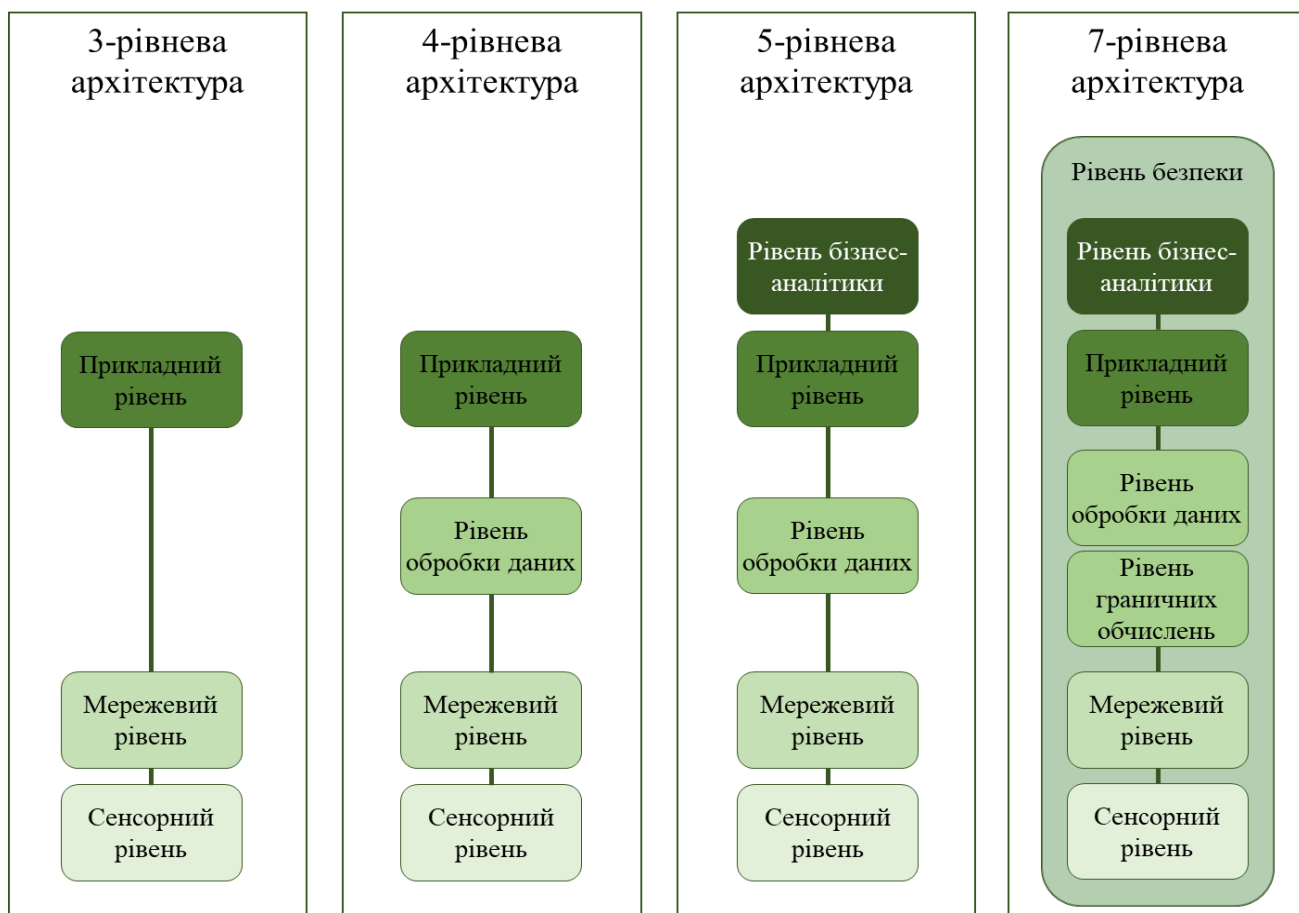


Рисунок 2.1 – Моделі архітектури систем інтернету речей

Історично першою, що сформувалася ще на ранніх етапах розвитку IoT, є трирівнева архітектура. Вона складається з сенсорного, мережевого та прикладного рівнів. Ці рівні є базовими для всіх типів архітектури, що використовуються для систем інтернету речей.

Рівень сприйняття або сенсорний рівень – це базовий, фундаментальний рівень архітектури системи інтернету речей, який забезпечує зв'язок цифрового світу з фізичним середовищем. Основне завдання цього рівня – збір первинної інформації про стан об'єкта чи середовища та її цифрування.

Цей рівень охоплює пристрої збору даних та виконавчі механізми: різноманітні датчики (температури, вологості, тиску, світла, руху, газів, біометричні тощо), камери, RFID-мітки, лічильники, а також актуатори, які реагують на команди системи, наприклад, вмикають освітлення.

Другим ключовим компонентом архітектури системи інтернету речей є мережевий рівень. Його ще називають рівнем зв'язку, рівнем підключення або транспортним рівнем. Основна функція мережевого рівня полягає в організації зв'язку між усіма компонентами IoT-системи. Він забезпечує передавання даних від пристроїв і сенсорів до вузлів обробки, хмарних платформ чи прикладних сервісів для подальшої обробки й аналізу. Мережевий рівень відповідає не лише за сам факт передавання даних, а й за стабільність, швидкість, пропускну здатність та захист цього процесу [14].

Для роботи мережевого рівня використовують широкий спектр технологій зв'язку [4; 69]:

- бездротові мережі короткого радіусу дії (Bluetooth, Zigbee, NFC);
- бездротові мережі середнього та великого радіусу дії (Wi-Fi, LoRaWAN, NB-IoT, LTE-M, 4G/5G);
- дротові мережі (Ethernet, оптоволоконні лінії);
- протоколи обміну повідомленнями та маршрутизації (MQTT, CoAP, HTTP/HTTPS, AMQP), які формують структуру та правила передачі даних у мережі IoT.

У складі мережевого рівня також працюють шлюзи (gateways) та системи збору даних (Direct-attached storage, DAS), які виконують роль проміжних вузлів. Вони можуть фільтрувати або агрегувати дані перед передачею на наступні рівні, а також забезпечують сумісність різних протоколів і підключення локальних пристроїв до глобальної мережі інтернет чи хмари.

Прикладний рівень, який ще називають рівнем застосунку або програмним інтерфейсом, є верхнім рівнем трирівневої архітектури системи IoT, який забезпечує взаємодію користувача з усією системою. Саме на цьому рівні результати роботи сенсорів, мереж та аналітики стають доступними для людей або зовнішніх програмних систем. Головна функція прикладного рівня – інтерпретація та візуалізація даних, що надходять від нижчих рівнів архітектури. Тут використовують аналітичні алгоритми, програмні платформи, сервіси візуалізації й інтерфейси користувача, які дозволяють приймати обґрунтовані рішення або автоматизувати процеси без безпосереднього втручання людини.

Трирівнева архітектура інтернету речей дуже розповсюджена у зв'язку з цілою низкою переваг. Вона має просту, зрозумілу структуру, яку можна швидко і легко впровадити, без надлишкових витрат. Такі системи зручні для застосунків із обмеженим обсягом даних (наприклад, локальний моніторинг температури, вологості). Однак, вони не мають достатніх ресурсів для обробки великих обсягів даних чи складних аналітичних інструментів. Крім того, питання безпеки вирішуються лише локально на рівні пристроїв або програм. Тому в сучасних рішеннях 3-рівневу архітектуру зазвичай застосовують як базову модель або основу для подальшого розширення до 4-, 5- чи 7-рівневих структур.

Чотирирівнева архітектура систем IoT є удосконаленою версією 3-рівневої, головна відмінність якої полягає у введенні додаткового рівня обробки даних [33; 42].

Рівень обробки даних (ще одна назва – рівень проміжного програмного забезпечення) є центральною ланкою у 4-, 5- та більш багаторівневих архітектурах інтернету речей. Його основне призначення – збирати, зберігати, фільтрувати та

аналізувати дані, отримані від пристроїв на рівні сприйняття, і передавати їх у зручному та стандартизованому вигляді на прикладний рівень.

Рівень обробки даних виконує низку важливих функцій:

- попередня обробка даних з метою очищення, нормалізації, фільтрації шумів та дублюючих записів;
- уніфікація, агрегація та буферизація даних, зібраних із різних пристроїв, у зручному форматі для подальшої обробки;
- локальна аналітика та обчислення на периферії (edge/fog computing) для виконання частини алгоритмів безпосередньо на шлюзах чи локальних серверах, що зменшує затримки і навантаження на центральні сервери;
- зберігання даних, робота з базами даних і сховищами даних для забезпечення доступу до історичних записів.

Ключовою відмінністю та перевагою 4-рівневої архітектури у порівнянні з 3-рівневою є те, що додатковий рівень забезпечує обробку даних ближче до джерела. Це знімає частину навантаження з прикладного рівня, який завдяки цьому може зосередитися на взаємодії з користувачами та бізнес-процесами. Крім того, попередня фільтрація та аналітика на рівні шлюзів прискорює реакцію системи в реальному часі. У 3-рівневій архітектурі дані з сенсорного рівня через канал зв'язку передаються безпосередньо до прикладних програм, які самі мають зберігати, обробляти й аналізувати інформацію.

Однак, 4-рівнева модель складніша й дорожча у впровадженні, вимагає інвестицій у додаткове програмне забезпечення, сервери або хмарні сервіси. Вона може бути надлишковою для простих локальних систем, де достатньо прямого каналу від сенсорів до прикладного рівня.

П'ятирівнева архітектура є наступним кроком розвитку інтернету речей і відрізняється введенням додаткового, верхнього рівня – бізнес-рівня або рівня бізнес-аналітики, який перетворює технічні дані, зібрані й оброблені нижчими рівнями, на інформацію, придатну для прийняття управлінських рішень і стратегічного планування.

На цьому рівні відбувається глибокий аналіз великих обсягів даних (Big

Data), які надходять із сенсорів, шлюзів і систем обробки. Використовуються інструменти бізнес-аналітики, статистичні методи, алгоритми машинного навчання, моделі прогнозування та інші технології, які дозволяють:

- виявляти закономірності та тенденції у даних;
- прогнозувати потреби або можливі проблеми;
- оцінювати ефективність виробництва, споживання ресурсів, логістики чи сервісного обслуговування;
- формувати показники результативності (Key Performance Indicators, KPI), звіти, графіки, інформаційні панелі для керівників та менеджерів;
- оптимізувати витрати, планувати інвестиції та модернізацію на основі реальних даних.

Рівень бізнес-аналітики також відповідає за інтеграцію IoT-даних із корпоративними інформаційними системами і тим самим забезпечує безперервний цикл: дані – аналітика – рішення – дія. Завдяки цьому компанії можуть своєчасно реагувати на зміни у виробничих або зовнішніх умовах, поліпшувати якість продукції чи послуг, підвищувати безпеку та екологічну стійкість.

Отже, головна перевага 5-рівневої архітектури полягає в інтеграції отриманих даних у бізнес-процеси, що дозволяє компаніям і організаціям не лише бачити дані, а й приймати на їх основі стратегічні та оперативні рішення. Недоліками таких систем є значна складність, зростання витрат за рахунок додаткових інвестицій у спеціалізоване аналітичне програмне забезпечення, фахівців із бізнес-аналітики, високі вимоги до кібербезпеки, оскільки бізнес-дані часто мають стратегічну або конфіденційну цінність. 5-рівнева архітектура економічно виправдана тільки для впровадження систем інтернету речей на середніх і великих підприємствах або організаціях, але не підходить для проєктів з автоматизації на рівні звичайних домогосподарств та малого бізнесу.

Подальшим розвитком 5-рівневої моделі інтернету речей стала 7-рівнева архітектура, яка використовується для великих, складних і критично важливих систем, наприклад, «розумних» міст чи транспортної інфраструктури. Її головна

відмінність полягає у додатковій деталізації обробки даних та введенні окремого рівня безпеки.

Рівень безпеки пронизує всю систему і діє як “захисна оболонка” для всієї архітектури, інтегруючись у кожний рівень [46]. Так, на рівні сприйняття відбувається автентифікація та захищена реєстрація пристроїв у мережі, захист прошивки та безпечні оновлення, шифрування даних ще на рівні сенсора, фізичний захист пристроїв від підключення сторонніх датчиків або витоку даних. На мережевому рівні реалізовані шифрування трафіку, безпечна маршрутизація й автентифікація вузлів мережі, захист від мережеских атак, контроль пропускну здатності та моніторинг мережевої активності. На рівні граничних обчислень передбачається захист шлюзів і edge-серверів від несанкціонованого доступу, локальна перевірка достовірності даних та запобігання їх підробці, використання безпечних апаратних модулів для зберігання ключів і сертифікатів, виявлення аномалій і реагування на вторгнення в реальному часі. На рівні обробки інформації здійснюється автентифікація користувачів та пристроїв, шифрування даних під час передавання та зберігання у хмарі, резервне копіювання, відновлення після збоїв і моніторинг журналів подій, захист API-інтерфейсів та безпечні інтеграції з іншими системами. Прикладний рівень потребує захисту веб- і мобільних інтерфейсів, керування правами доступу користувачів, запобігання витоку конфіденційних даних. На рівні бізнес-аналітики відбувається контроль доступу та шифрування до даних, моніторинг та аудит усіх бізнес-процесів, захист моделей машинного навчання й алгоритмів від маніпуляцій та витоку.

Додаткова деталізація обробки даних реалізована через розподіл обробки й аналізу даних між рівнем граничних обчислень (Edge/Fog) та рівнем обробки інформації (Processing/Cloud).

Рівень граничних (або периферійних) обчислень виконує швидкі та локальні операції, що вимагають миттєвої реакції або не потребують передавання повного масиву даних. Цей рівень складається з локальних обчислювальних блоків та розташований максимально близько до пристроїв: на шлюзах, контролерах чи навіть на самих сенсорах [17; 48; 52]. Він виконує:

- первинну фільтрацію та очищення даних (видалення шумів, дублювань, помилок вимірювань);
- попередню агрегацію, тобто об'єднання даних із кількох сенсорів для зменшення обсягу передачі;
- локальні розрахунки в реальному часі;
- застосування попередньо навчених моделей машинного навчання, наприклад, виявлення дефектів на виробничій лінії або прогноз виходу з ладу пристрою;
- буферизацію даних для тимчасового зберігання при збоях зв'язку із хмарою.

Рівень обробки інформації бере на себе ресурсомісткі, довготривалі та глибокі аналітичні завдання, що вимагають великої обчислювальної потужності, сховищ даних та інтеграції з іншими системами [52]. Цей рівень реалізований у центрах обробки даних або хмарних платформах і виконує:

- глибоку аналітику та візуалізацію, побудову складних моделей, тренування нейромереж, аналіз довгих часових рядів;
- інтеграцію великих обсягів даних (Big Data) для комплексного аналізу;
- прогнозування та оптимізацію ресурсів, попиту, виявлення закономірностей у великих історичних наборах даних;
- довгострокове зберігання та керування даними (формування архівів, журналів безпеки, звітів для бізнес-аналітики);
- забезпечення масштабованості, тобто централізованої підтримки тисяч або мільйонів пристроїв одночасно.

Перевагами такого розподілу обробки й аналізу даних є мінімальні затримки у відклику системи; зменшення навантаження на мережу та раціональне використання ресурсів, тому що передаються лише потрібні дані, а потужна, але дорожча хмарна інфраструктура використовується лише для тих завдань, що справді її потребують. Крім того, локальні вузли здатні працювати автономно у випадку збою зв'язку, що підвищує стійкість системи.

Отже, 7-рівнева архітектура інтернету речей – це найповніша та найбільш

деталізована модель побудови IoT-систем, яка має суттєві переваги з точки зору продуктивності обробки даних, масштабованості, надійності, безпеки та інтеграції з бізнес-процесами. Це робить 7-рівневу архітектуру оптимальною для масштабних, критично важливих систем [22].

Однак, її впровадженню в локальних, невеликих проєктах перешкоджають такі недоліки: висока складність розгортання, інтеграції та управління, що потребує досвідчених фахівців; значні витрати на обладнання, хмарні ресурси, навчання персоналу, кіберзахист і технічну підтримку. У простих локальних системах (наприклад, моніторинг будинку або теплиці) більшість можливостей цієї архітектури не використовується, що робить її надмірно складною й дорогою, а отже – недоцільною.

За результатами проведеного аналітичного огляду існуючих підходів до формування архітектури систем інтернету речей можна зробити висновок, що вибір архітектури для конкретного проєкту IoT залежить від масштабу системи, кількості пристроїв, обсягів даних і вимог до швидкості реакції та безпеки. Чим більший масштаб і складність проєкту, тим більше рівнів повинна мати архітектура. Для невеликих проєктів зайві рівні лише підвищують витрати та ускладнюють підтримку. Для пілотних або маленьких локальних проєктів, навчальних і тестових рішень достатньо 3-рівневої архітектури. 4-рівнева архітектура оптимальна для невеликих і середніх систем, які потребують базової автоматизації (наприклад, розумне освітлення будівлі). 5-рівнева – найбільш придатна для корпоративних рішень, де важлива бізнес-аналітика й інтеграція з управлінськими процесами. 7-рівнева архітектура потрібна там, де важливі висока продуктивність у реальному часі, безпека та масштабованість. Вона використовується для великих, критично важливих, територіально розподілених систем, що потребують швидкої реакції, захищеності та надійності.

Отже, для невеликого проєкту розумного енергоспоживання, що розрахований на масового споживача, оптимально обрати 4-рівневу архітектуру як найкращий компроміс між простотою, вартістю, безпекою та можливістю подальшого розвитку.

2.3 Архітектура системи розумного енергоспоживання

Метою даної роботи є розробка економічно доцільного, доступного широкому колу споживачів рішення IoT, спрямованого на підвищення енергоефективності, з достатньо широким функціоналом, що поєднує моніторинг енергоспоживання з відстеженням якості внутрішнього середовища й аналітикою, та має простий, інтуїтивно зрозумілий інтерфейс.

При побудові архітектури такої системи IoT треба враховувати, що:

- система повинна охоплювати всі складові енергоспоживання;
- система повинна комплексно реалізувати весь потрібний функціонал;
- повинні бути враховані ключові вимоги до архітектури.

Всі ці питання були ретельно розглянуті в підрозділах 1.1, 1.3 та 2.1.

Крім того, у попередньому підрозділі було визначено, що для невеликої, локальної системи інтернету речей оптимально обрати 4-рівневу архітектуру. Однак для розширення можливостей системи доцільно додати до такої інфраструктури аналітичні інструменти та можливість створювати сценарії енергоспоживання. Запропонована архітектура системи моніторингу та управління енергоспоживанням представлена на рисунку 2.2.

Перший, сенсорний рівень або рівень сприйняття, складається з засобів моніторингу параметрів електромережі (активної потужності, сили струму, напруги, частоти струму, коефіцієнту потужності), присутності людини у приміщенні та стану мікроклімату (освітленості, температури та концентрації CO₂). Такими приладами є датчики та вимірювальні модулі. Крім того, на цьому рівні архітектури розташовані актуатори – виконавчі механізми, що здійснюють керуючі дії у відповідь на зміни в навколишньому середовищі. Наприклад, вони будуть вмикати та вимикати світло в залежності від присутності людини, рівня освітленості або бажань користувача. В якості актуаторів використовуються реле.

Другим рівнем є рівень зв'язку або мережевий рівень. Його призначення – забезпечити надійний та ефективний обмін даними між різними елементами системи інтернету речей. Можливими опціями є дротовий або бездротовий зв'язок.

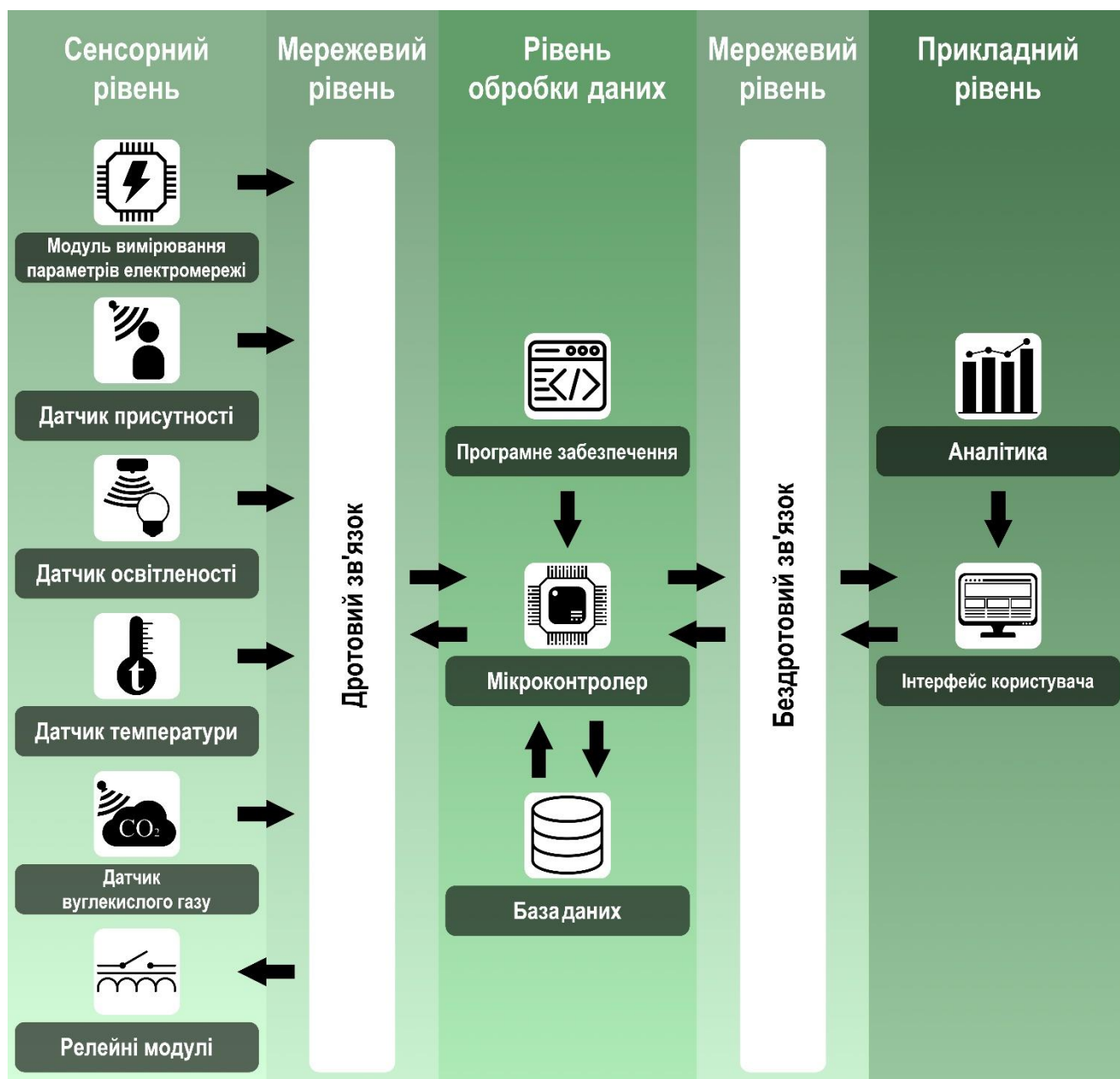


Рисунок 2.2 – Архітектура системи розумного енергоспоживання

Дротовий тип зв'язку забезпечує вищу надійність передачі даних і стабільність роботи порівняно з бездротовим, оскільки передача даних здійснюється по фізичному кабелю, який не піддається впливу радіоперешкод, сигнал складно перехопити або підробити. Однак, його використання в системах інтернету речей має низку суттєвих обмежень. Передусім, кабельна інфраструктура знижує гнучкість розгортання системи, ускладнює швидке та економічно доцільне масштабування мережі інтернету речей. Це особливо відчутно в умовах, коли

потрібно з'єднати пристрої, розташовані на значній відстані і дані мають передаватися віддалено. У таких випадках прокладання кабелів стає невиправдано дорогим, складним або навіть технічно неможливим. Крім того, дротові мережі схильні до фізичних пошкоджень і поступового зношення, що створює потребу в регулярному технічному обслуговуванні. Процес усунення проблем у таких системах зазвичай є складнішим, дорожчим і потребує більше часу, ніж у бездротових рішеннях. З іншого боку, бездротовий зв'язок більш гнучкий, дешевший і простіший у масштабуванні, проте може бути менш стабільним і потребує додаткового захисту.

Тому у локальній системі розумного енергоспоживання дротовий тип зв'язку доцільно використовувати між сенсорним рівнем та рівнем обробки даних, тому що всі датчики, модулі вимірювання та актуатори знаходяться поряд з мікроконтролером. Однак для передачі даних з рівня обробки інформації далі, на прикладний рівень, слід використовувати бездротові технології, що збільшить здатність системи до масштабування, підвищить її адаптивність, забезпечить віддалений доступ до даних.

Третім рівнем архітектури системи розумного енергоспоживання є рівень обробки даних (рівень проміжного програмного забезпечення). Він є центральною ланкою архітектури і виконує функції збору, зберігання, фільтрування та попереднього аналізу даних, отриманих від пристроїв сенсорного рівня, а також передає їх у зручному та стандартизованому вигляді на вищий рівень. Головним апаратним компонентом третього рівня є мікроконтролер, який забезпечує «інтелектуальність» системи. Програмним компонентом цього рівня є серверна частина програмного забезпечення – прошивка мікроконтролера.

Вищим рівнем архітектури системи моніторингу та управління енергоспоживанням є прикладний рівень або рівень застосування, що забезпечує взаємодію користувача з системою розумного енергоспоживання через клієнтську частину програмного забезпечення – інтерфейс користувача. Головна мета цього рівня – інтерпретація та візуалізація даних, що надходять від нижчих рівнів архітектури. На прикладному рівні користувач отримує інформацію стосовно

поточних значень параметрів мікроклімату в приміщенні, поточних значень параметрів електромережі, поточної активності підключених до системи приладів освітлення, кондиціонування, опалення тощо. Крім того, на цьому рівні користувач здійснює:

- загальне налаштування системи (час та дату вбудованого в систему модулю реального часу, IP адресу, шлюз, підмережу, доменне ім'я, SSID, пароль);
- визначення інтервалу, з яким система отримує інформацію, та інтервалу, з яким система записує інформацію в базу даних;
- налаштування датчиків та їх калібрування;
- створення сценаріїв використання підключених пристроїв (автоматична активація при подоланні зазначених користувачем порогових значень параметрів внутрішнього середовища, активні/неактивні періоди, «імітація присутності», вечірній режим тощо).

Прикладний рівень також надає користувачу додаткові можливості за рахунок аналітичного функціонала, призначення якого полягає у перетворенні даних, зібраних на першому сенсорному рівні, у корисну інформацію для прийняття рішень та створення сценаріїв. Це відбувається за допомогою статистичних методів та графічної візуалізації. Аналітика допомагає відповісти на питання скільки енергії витрачається, на що саме, чому і як це можна скоротити без втрати комфорту.

Наступними етапами розробки системи розумного енергоспоживання є підбір окремих елементів побудованої архітектури з урахуванням сформульованих у підрозділі 2.1 вимог та розробка відповідного програмного забезпечення.

2.4 Система управління базою даних

Найбільш поширеними підходами до зберігання даних є реляційні бази даних із використанням SQL (Structured Query Language, або мова структурованих запитів) та нереляційні або NoSQL-бази даних [7; 19]. Вони обидва мають власні принципи організації, методи доступу до інформації та оптимальні сценарії застосування. Вибір між ними визначається насамперед структурою даних, що

зберігаються, обсягом сховища та вимогами до швидкості обробки запитів.

SQL-бази даних засновані на реляційній моделі та зберігають інформацію у вигляді таблиць, що мають чітко визначені структури зі стовпцями та рядками. Їх основною характеристикою є використання мови запитів SQL для виконання CRUD-операцій (Create Read Update Delete, тобто додавання, пошук, оновлення та видалення даних). Цей підхід забезпечує строгу узгодженість і підтримує транзакції, що робить такі бази ідеальними для критично важливих систем, де мають значення точність і цілісність даних. Реляційні СУБД (системи управління базою даних) традиційно застосовуються у фінансових системах, бухгалтерії, управлінні ресурсами та багатьох інших сферах, де дані мають чіткі зв'язки між собою.

NoSQL-бази даних зі свого боку націлені на більш гнучке зберігання інформації. Вони можуть зберігати дані у форматах документів, графів, ключ–значення або колонкових структур. Такі СУБД добре підходять для роботи з великими обсягами інформації, що швидко змінюється, і для систем, де структура даних є динамічною або не може бути чітко визначеною заздалегідь. Перевагами NoSQL-рішень є масштабованість, відносно висока швидкість обробки запитів та простота роботи з даними. З іншого боку, вони зазвичай гірше узгоджені та мають обмежену підтримку транзакцій, що може стати суттєвим недоліком у деяких випадках.

З огляду на специфіку даного проєкту, що передбачає збереження показників датчиків з певним інтервалом та запис точного часу їх зняття, доцільніше обрати SQL-рішення. Пояснюється це тим, що у даних є чітка незмінна структура: кожен запис складається з фіксованого набору параметрів та часової позначки. Саме для таких випадків реляційна модель з таблицями і транзакційною підтримкою найкраще підходить, адже забезпечує прозору організацію інформації, простоту аналізу та мінімальні ризики втрати даних.

Серед SQL-СУБД доцільно розглянути три найбільш розповсюджених, а саме: SQLite, MySQL та PostgreSQL. Кожна з них має свої особливості, переваги та

обмеження, проте всі вони підтримують реляційну модель і забезпечують достатню функціональність для роботи зі зчитаними з датчиків показниками.

SQLite є однією з найпопулярніших СУБД, яка широко застосовується у мобільних додатках, IoT-пристроях та вбудованих системах. Вона не потребує розгортання серверної частини, оскільки база зберігається у вигляді єдиного файлу на локальному носії. Це особливо зручно для SD-карти. Перевагами SQLite є легкість, підтримка транзакцій, простота інтеграції та відсутність необхідності у складній конфігурації чи адмініструванні. Недоліком можна вважати обмежену продуктивність у масштабних проєктах із мільйонами записів та одночасним доступом до бази багатьох клієнтів, проте для локальної IoT-системи ці обмеження не є критичними [63].

MySQL є більш потужною реляційною СУБД, яка традиційно використовується для веб-додатків та серверних рішень. Вона підтримує багатокористувацький доступ, масштабованість та високий рівень надійності. Її перевагами є велика спільнота користувачів, активна підтримка та розвинуті можливості для оптимізації запитів. Недоліком у контексті даного проєкту є необхідність розгортання серверної частини та досить високі вимоги до апаратних ресурсів. Це ускладнює використання MySQL на мікроконтролерах із обмеженою пам'яттю та обчислювальними можливостями.

PostgreSQL вважається однією з найбільш насичених функціоналом та найгнучкіших реляційних СУБД. Вона підтримує складні запити, роботу з географічними даними та численні розширення. Вона добре підходить для систем, де важлива масштабованість та складний аналіз даних. Серед переваг PostgreSQL можна відзначити надійність, дотримання стандартів SQL та широкий спектр можливостей для адміністрування. Однак, як і MySQL, ця СУБД вимагає суттєвих ресурсів для ефективної роботи, що робить її менш придатною для невеликих систем, що використовують обмежені обчислювальні ресурси та обсяг пам'яті.

Для розробок у сфері локальних систем інтернету речей, остаточний вибір СУБД має ґрунтуватися на ресурсах апаратної частини. Враховуючи особливості розглянутих варіантів і те, що апаратну основу системи розумного

енергоспоживання складає мікроконтролер, оптимальною СУБД є SQLite. Вона є легкою та компактною, при цьому підтримує транзакції, що гарантує цілісність даних при одночасному доступі кількох процесів або потоків. Крім того, SQLite є простою у використанні та не вимагає розгортання серверної інфраструктури, що ідеально підходить для мікроконтролера із SD-картою як основним носієм даних.

2.5 Вибір протоколу бездротового зв'язку

У системах інтернету речей протокол бездротового зв'язку є одним з ключових елементів, що визначає, як саме пристрої обмінюються даними. Вибір протоколу напряду впливає на швидкість, енергоспоживання, радіус дії, масштабованість і надійність усієї системи. Тому, під час розробки необхідно враховувати апаратні можливості контролюючого модуля, особливості середовища, вимоги до безпеки та обсягу даних, які будуть передаватись. Розглянемо для порівняльного аналізу найпоширеніші протоколи бездротового зв'язку, що використовуються у IoT-проєктах, а саме: Wi-Fi, Bluetooth, ZigBee, Z-Wave, LoRa/LoRaWAN та стільникові технології [4; 22; 64].

Wi-Fi є одним із найпоширеніших протоколів бездротового зв'язку в системах інтернету речей. Він забезпечує передачу даних у діапазоні 2,4 ГГц та 5 ГГц у нових стандартах, пропонуючи високу швидкість – сотні мегабіт на секунду. Також перевагами Wi-Fi є відносна простота налаштування та використання стандартних протоколів TCP/IP, що забезпечує повну сумісність з інтернет-інфраструктурою. Недоліками виступають вище за інші бездротові технології енергоспоживання, що обмежує автономну роботу пристроїв, а також обмежений радіус дії без додаткових роутерів або ретрансляторів.

Bluetooth у свою чергу орієнтований на з'єднання на малих відстанях. Класичний Bluetooth дозволяє передавати дані зі швидкістю до кількох мегабіт за секунду та підходить для поточкових застосувань, наприклад аудіо. Натомість BLE (Bluetooth Low Energy) спроектовано спеціально для сенсорів і пристроїв із низьким енергоспоживанням, забезпечуючи швидкість до 1 Мбіт/с при значній

економії енергії [4; 16]. Окрім цього, Bluetooth простий в інтеграції з мобільними пристроями та дуже поширений. До його недоліків можна віднести ще менший, у порівнянні із Wi-Fi, радіус дії і неможливість забезпечити велику кількість одночасних з'єднань [46].

ZigBee від початку створювався як протокол для WSN-рішень (Wireless Sensor Network, тобто бездротова сенсорна мережа). Класичний сценарій його використання це комунікація датчиків, шлюзів та хабів всередині системи розумного будинку для передачі та збирання інформації. Отже, його головною перевагою є підтримка топології «mesh», де пристрої можуть передавати дані один через одного, забезпечуючи надійність і розширення зони покриття. Протокол енергоефективний і підходить для пристроїв, що працюють на батарейках. Водночас швидкість передачі даних серйозно обмежена, що робить ZigBee не дуже придатним для застосувань, де є необхідність у передачі великих обсягів даних або безпосередня інтеграція з веб-інтерфейсом [12; 35].

Z-Wave є пропрієтарним протоколом, який наразі належить Silicon Labs. Він має відкриті специфікації, але стек і сертифікація контролюються Silicon Labs, тому може використовуватись чипами лише від одного постачальника. Z-Wave орієнтований на IoT-мережі з низьким енергоспоживанням і працює у діапазоні близько 868 МГц у Європі та 908 МГц у США, що знижує рівень радіоперешкод порівняно з перевантаженим спектром 2,4 ГГц. Як і ZigBee, Z-Wave підтримує mesh-топологію, дозволяючи будувати розподілені мережі сенсорних-модулів і контролерів. Його перевагами є стабільність і простота інтеграції у розумні будинки із сумісною архітектурою, але закритість протоколу ускладнює його використання в кастомних проєктах. Також, Z-Wave має низьку швидкість передачі даних. Перелічені чинники роблять його малоприматним для потреб запропонованого проєкту локальної системи інтернету речей [4; 35].

LoRa та LoRaWAN є прикладами протоколів для наддовгих відстаней. Вони можуть забезпечити передачу даних на кілька кілометрів при дуже низькому енергоспоживанні. Водночас швидкість передачі становить лише кілька кбіт/с,

тому LoRa та LoRaWAN підходить для коротких повідомлень і не може забезпечити роботу веб-інтерфейсів чи передачу великих обсягів даних [4].

Стільникові IoT-технології, такі як NB-IoT та LTE-M, базуються на використанні мобільних мереж операторів. Вони забезпечують широке покриття, високу надійність та інтеграцію з існуючою інфраструктурою. Основною перевагою є можливість підключення пристроїв практично будь-де за наявності мобільного зв'язку. З іншого боку, рішення на основі стільникових технологій створює залежність від конкретних операторів та необхідність витратити кошти на SIM-карти та абонентську плату. Ще варто відзначити, що зв'язок через стільникові станції має більшу затримку передачі даних у порівнянні із локальною мережею [4; 46].

Порівняльний аналіз показує, що, враховуючи потребу у взаємодії з веб-інтерфейсом та роботі у реальному часі, для системи моніторингу та управління енергоспоживанням будівлі найбільш доцільно використовувати Wi-Fi. Протоколи ZigBee, Z-Wave і LoRaWAN добре підходять для автономних сенсорних мереж із малим трафіком, проте вони не здатні забезпечити потрібну швидкість для передачі робочих файлів веб-інтерфейсу. Bluetooth, попри свою енергоефективність, має занадто малий радіус дії, низьку швидкість та обмеження за кількістю з'єднань. Перевага стільникових технологій у вигляді високого радіуса дії перевищує потреби системи розумного енергоспоживання дому, бо для нього достатньо локальної мережі. При цьому вони додають залежність від операторів та підвищують грошові витрати на підтримання роботи системи. Wi-Fi в свою чергу забезпечує оптимальний баланс між швидкістю, простотою інтеграції та доступністю апаратних рішень, що робить його найбільш придатним протоколом зв'язку для локальної системи моніторингу та управління енергоспоживанням.

2.6 Апаратне забезпечення системи моніторингу та управління енергоспоживанням

2.6.1 Вибір мікроконтролера

В результаті огляду існуючих плат на основі мікроконтролерів було відібрано для порівняльного аналізу декілька рішень, в яких реалізована можливість бездротового зв'язку. Серед висунутих вимог та параметрів, на які треба звертати увагу в першу чергу, є кількість ядер процесора та підтримка багатозадачності, частота процесора, об'єми оперативної та флеш-пам'яті, кількість контактів та підтримувані на них інтерфейси, наявність декількох режимів роботи для контролю енергоспоживання та вартість придбання.

Першим розглянемо мікроконтролер ESP8266 NodeMCU V3. Даний мікроконтролер є одним із найдешевших серед тих, що пропонують вбудований бездротовий зв'язок, і тому є дуже розповсюдженим у сфері IoT-проектів. Він має одне ядро, працює на частоті 80 або 160 МГц, має обсяг оперативної пам'яті у 50 КБ і потребує зовнішньої флеш-пам'яті для зберігання користувацької програми. Ще однією слабкою стороною даного рішення є наявність лише базових периферійних можливостей, а саме обмежена кількість входів/виходів загального призначення (GPIO, General-Purpose Input/Output) та спеціалізованих виходів/контактів. ESP8266 обмежено реалізує багатозадачність через наявність лише 1 ядра процесора. З точки зору енергоспоживання даний мікроконтролер може бути дуже економним у режимі deep-sleep, але при активному використанні Wi-Fi-з'єднання витрати енергії стають значно помітніші. На рисунку 2.3 зображено одну з найбільш розповсюджених плат на основі ESP8266 NodeMCU V3. Вона надає можливість завантажувати програму користувача за допомогою USB-порту та пропонує стабілізатор напруги AMS1117 [29].

ESP32 схожий на ESP8266 своїми сильними сторонами – він також має вбудований модуль бездротового зв'язку і потрапляє у дешевий сегмент серед мікроконтролерів. Крім того, він має додаткове обладнання, що робить його досконалішим, ніж попередній аналог. Так, він має вже два ядра, робочу частоту у

240 МГц, обсяг оперативної пам'яті у 520 КБ та флеш-пам'яті у 4 МБ та більшу кількість периферії. Відповідно, через вищенаведені переваги, ціна даного мікроконтролера вища за ESP8266, проте його все ще можна віднести у дешевий сегмент. Енергоспоживання ESP32 у активному режимі роботи є вищим, ніж у ESP8266, проте у режимі сну – набагато нижчим. На рисунку 2.4 зображено ESP32 DEVKIT V1, який є одним із найбільш популярних комплектів розробки на базі ESP32. Він теж реалізує можливість завантажувати прошивку через USB та пропонує стабілізатор напруги [27; 28; 43; 45].

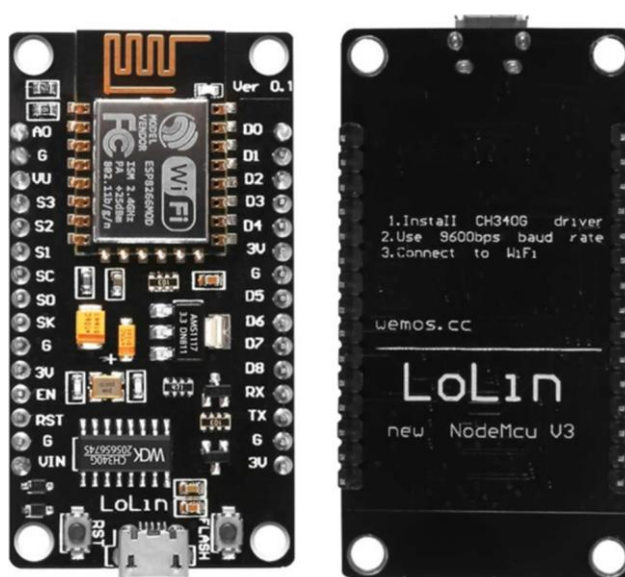


Рисунок 2.3 – Зовнішній вигляд NodeMCU V3 на основі ESP8266

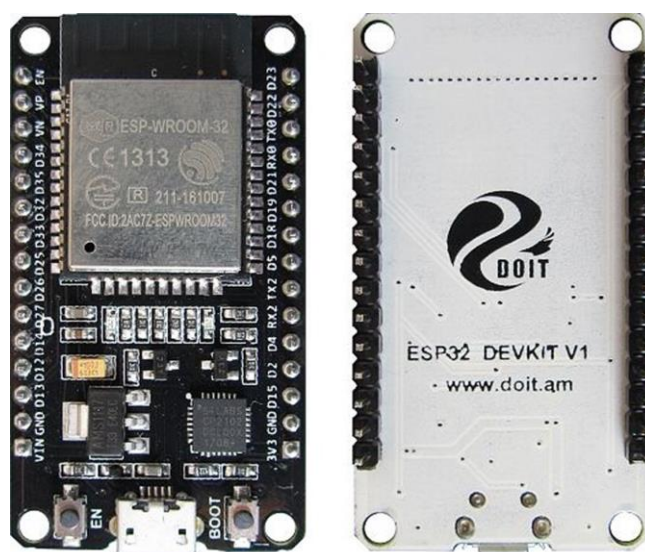


Рисунок 2.4 – Зовнішній вигляд ESP32 DEVKIT V1

Модуль Raspberry Pi Pico W, зображений на рисунку 2.5, базується на мікроконтролері RP2040, що має два процесорних ядра та частоту 133 МГц. Вона додає Wi-Fi за допомогою чипа Infineon CYW43439, під'єднаного до контролера через послідовний периферійний інтерфейс (SPI, Serial Peripheral Interface) та універсальний асинхронний приймач/передавач (UART, Universal Asynchronous Receiver/Transmitter). При цьому дана плата зберігає відносно низьку ціну. Raspberry Pi Pico W має підключену флеш-пам'ять об'ємом у 2 МБ та вбудовану оперативну пам'ять мікроконтролера об'ємом у 264 КБ. Основна перевага даної плати – програмовані входи/виходи (PIO, Programmed Input/Output), які надають виняткову гнучкість у реалізації нестандартних інтерфейсів і високошвидкісних протоколів на будь-яких контактах. Вони працюють незалежно від основних ядер Cortex-M0+ і дозволяють реалізувати будь-який цифровий протокол, навіть якщо його немає апаратно в чипі [61].

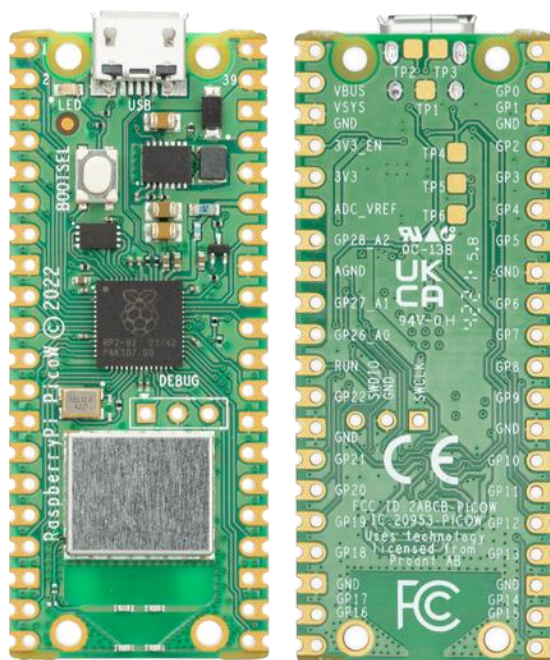


Рисунок 2.5 – Зовнішній вигляд Raspberry Pi Pico W

Наступною розглянемо модель L4 лінійки STM32, що була створена для роботи від автономних джерел живлення і тому орієнтована на якомога нижче енергоспоживання. Серед мікроконтролерів, що порівнюються, STM32L4

споживає найменший об'єм енергії в активному режимі роботи. Проте варто зазначити, що цей мікроконтролер не має вбудованого бездротового зв'язку, що суттєво знижує його показник енерговитрат у порівнянні із аналогами, де відповідний функціонал присутній. STM32L4 пропонує одне ядро процесора із частотою 80 МГц, 128 КБ оперативної пам'яті та 1 МБ флеш-пам'яті. Плата STM32L475E-IOT01A (рисунок 2.6) є собою готовим IoT-набором на базі STM32L4, що пропонує додатковий модуль бездротового зв'язку, інтерфейси USB та microUSB, слот для microSD, велику кількість датчиків і багато іншого. Головним недоліком такого рішення є висока ціна, хоча вона й виправдана широтою охопленого функціоналу [53; 67].

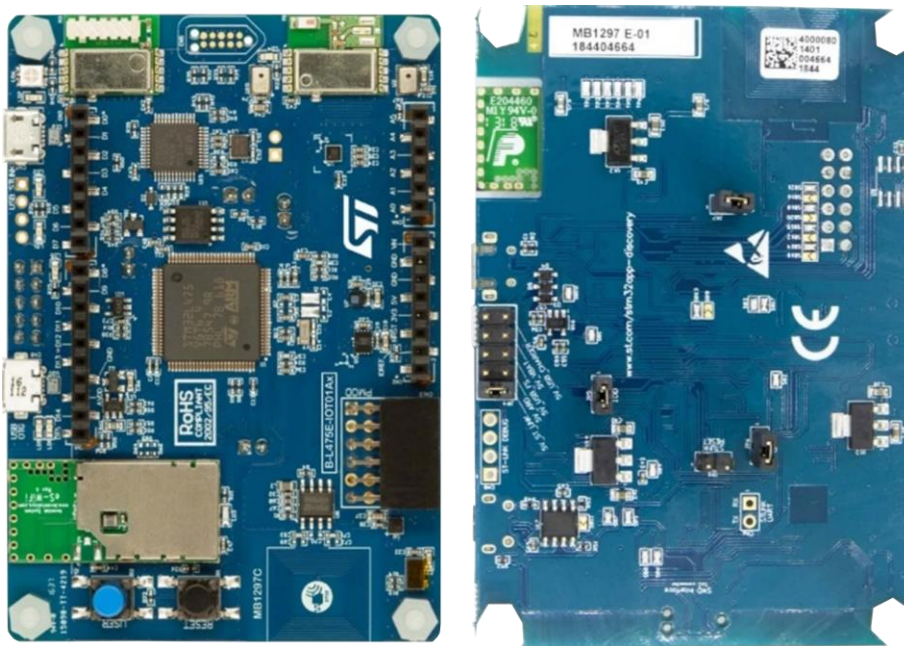


Рисунок 2.6 – Зовнішній вигляд STM32L475E-IOT01A

Модуль Arduino MKR1000, зовнішній вигляд якого зображений на рисунку 2.7, поєднує мікроконтролер SAMD21, який має одне ядро процесора із частотою 48 МГц, 32 КБ оперативної пам'яті та 256 КБ флеш-пам'яті, з Wi-Fi-модулем Atmel WINC1500 підключеним по SPI. Також плата для розробки додає USB-порт, зарядний контролер для Li-Po батарей та стабілізатор напруги 3,3 В. Дана модель акцентується на зручності взаємодії із нею та спрямована на користувачів, які не

хочуть заглиблюватись в низькорівневі налаштування. Недоліками цієї плати є висока ціна та водночас менш потужний процесор у порівнянні з аналогами [50].

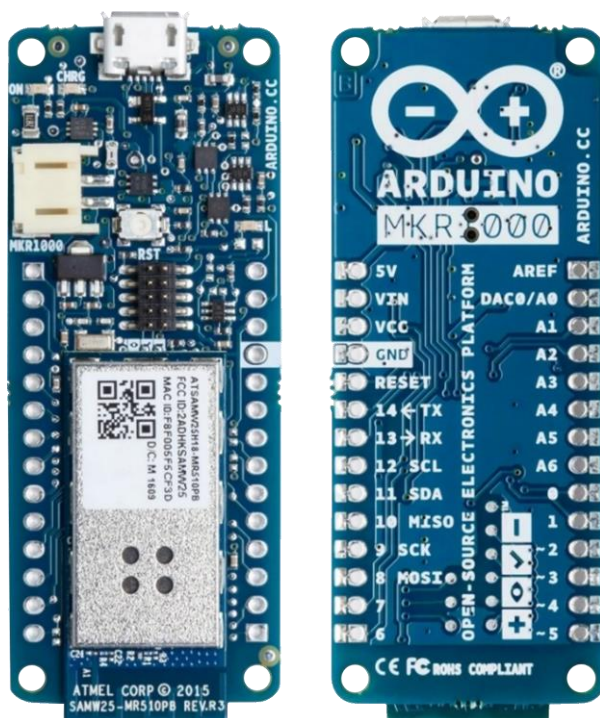


Рисунок 2.7 – Зовнішній вигляд Arduino MKR1000

У таблиці 2.1 наведено основні параметри усіх розглянутих плат.

В результаті порівняльного аналізу можна зробити висновок, що із розглянутих плат краще всього задовольняє висунутим вимогам ESP32 DEVKIT V1. Плати STM32L475E-IOT01A та Arduino MKR1000 мають суттєво більшу ціну, але пропонують лише 1 ядро та значно нижчі частоту процесора, обсяги оперативної та флеш-пам'яті. Це робить відповідне IoT-рішення малодоступним та не привабливим для широкого кола споживачів. Певною перевагою модуля STM32L4 є значно більша кількість контактів GPIO (82 проти 34 у ESP32), однак це є надлишковим для потреб даного проєкту. Модуль Raspberry Pi Pico W знаходиться приблизно у тій самій ціновій категорії, що й DEVKIT V1 на основі ESP32, і вони обидві мають 2 ядра, однак друга плата пропонує вищу частоту процесора, об'єми оперативної і флеш-пам'яті та більшу кількість інтерфейсів. І,

нарешті, мікроконтролерний модуль ESP8266 NodeMCU V3 у порівнянні із ESP32 DEVKIT V1 не набагато дешевший, проте має суттєво гірші параметри.

Таблиця 2.1 – Основні характеристики плат мікроконтролерів

Назва	Частота процесора, МГц	Ядра	Пам'ять		Контакти	Ціна, \$
			RAM	FLASH		
ESP8266 NodeMCU V3	80-160	1	50 КБ	512 КБ - 4 МБ	17 × GPIO 1 × ADC 1 × UART 1 × SPI 1 × I2C 1 × I2S	~4
ESP32 DEVKIT V1	240	2	520 КБ	4 МБ	34 × GPIO 18 × ADC 2 × DAC 16 × PWM 3 × UART 3 × SPI 2 × I2C 2 × I2S	~6
Raspberry Pi Pico W	133	2	264 КБ	2 МБ	26 × GPIO 4 × ADC 16 × PWM 2 × UART 2 × SPI 2 × I2C	~7
STM32L475E-IOT01A	80	1	128 КБ	1 МБ	82 × GPIO 3 × ADC 2 × DAC 6 × USART 3 × SPI 3 × I2C 2 × SAI	~54
Arduino MKR1000	48	1	32 КБ	256 КБ	22 × GPIO 7 × ADC 12 × PWM 2 × UART 1 × USB 1 × SPI 1 × I2C	~45

Отже, серед розглянутих мікроконтролерів найкращим вибором для системи інтернету речей, що моніторить і керує енергоспоживанням у приміщенні, є ESP32. Він забезпечує оптимальний баланс між вартістю та функціональними можливостями.

2.6.2 Вибір модуля енергомоніторингу

Після дослідження ринку існуючих пристроїв для вимірювання параметрів електроспоживання було відібрано декілька типових моделей для проведення порівняльного аналізу. В першу чергу треба звертати увагу на такі характеристики: які параметри дозволяє вимірювати модуль; частота оновлення даних; який інтерфейс використовується для передачі даних; напруга живлення та вартість.

Першим розглянемо M5Stack AC Measure Unit. Зовнішній вигляд AC Measure Unit зображений на рисунку 2.8.



Рисунок 2.8 – Зовнішній вигляд M5Stack AC Measure Unit

Це готовий модуль для вимірювання параметрів змінного струму, побудований на базі інтегральної мікросхеми HLW8032. Цей чип є спеціалізованим енергометром, який може вимірювати напругу, силу струму та активну потужність у мережі змінного струму. Він передає результати у вигляді пакетів даних через послідовний інтерфейс, а також має вбудовані цифрові фільтри, що забезпечують стабільність показників. Сам по собі HLW8032 видає дані через UART зі швидкістю 4800 бод, але робота з ним вимагає знання протоколу обміну та

додаткової обробки. Повноцінний модуль від M5Stack додає мікроконтролер STM32 та інтерфейс I²C (Inter-Integrated Circuit), що робить підключення значно простішим. Модуль також містить гальванічну розв'язку, завдяки якій вимірювальна частина, що підключена до мережі 220 В, електрично відділена від логіки управління. Це підвищує безпеку використання. Даний пристрій вимірює напругу, силу струму, активну потужність та спожиту енергію, проте не надає значення частоти мережі чи коефіцієнта потужності. Його оновлення відбуваються кожні 50 мс, що із запасом задовольняє вимоги даного проєкту. Діапазон допустимої напруги живлення – від 4,5 до 5,5 В.

Ще одним популярним модулем для моніторингу параметрів змінного струму, який часто використовується у проєктах з Arduino та ESP32, є PZEM-004T v3 (рисунок 2.9).



Рисунок 2.9 – Зовнішній вигляд PZEM-004T v3 без та з ізоляційним корпусом

На відміну від попереднього аналога, він додатково вимірює частоту мережі та коефіцієнт потужності. Має версію на 10 А, яка використовує вбудований шунт

для вимірювання, та версію на 100 А, яка використовує зовнішній струмовий трансформатор. Для передачі даних застосовується UART із протоколом Modbus-RTU. Як і попередній пристрій, PZEM-004T v3 живиться від напруги у діапазоні від 4,5 до 5,5 В. Теоретично даний модуль можна опитувати десятки разів за секунду відповідно до пропускну здатності UART, але на практиці він стабільно працює приблизно на частоті 1 Гц. При збільшенні частоти опитування можуть з'являтися затримки та помилки [60].

Останнім розглянемо SDM120-Modbus-MID. Це компактний однофазний лічильник електроенергії для монтажу на DIN-рейку, який розроблено для професійного використання. На рисунку 2.10 зображено його зовнішній вигляд.



Рисунок 2.10 – Зовнішній вигляд SDM120-Modbus-MID

Даний пристрій вимірює напругу, силу струму, активну потужність, коефіцієнт потужності, частоту мережі та спожиту енергію. На відміну від вищерозглянутих модулів, SDM120 живиться безпосередньо від мережі 230 В змінного струму і не потребує окремого джерела постійного струму. Пристрій підтримує інтерфейс RS-485 з протоколом Modbus-RTU, що робить його сумісним із промисловими системами автоматизації. Проте для підключення до ESP32 необхідно додатково придбати перетворювач рівнів з RS-485 на TTL. Він коштує значно більше у порівнянні із менш професійними модулями, однак забезпечує набагато вищу точність вимірювання. Теоретично цей лічильник має мінімальний період відповіді 80 мс, але на практиці оптимальна частота опитування не перевищує 5 Гц.

У таблиці 2.2 наведено основні параметри всіх розглянутих модулів вимірювання параметрів мережі змінного струму.

Таблиця 2.2 – Основні характеристики модулів моніторингу параметрів електромережі

Характеристика		Модуль		
		M5Stack AC Measure Unit	PZEM-004T v3	SDM120-Modbus-MID
Вимірювані параметри	Напруга	+	+	+
	Сила струму	+	+	+
	Активна потужність	+	+	+
	Спожита енергія	+	+	+
	Частота	—	+	+
	Коефіцієнт потужності	—	+	+
Напруга живлення, В		від 4,5 до 5,5	від 4,5 до 5,5	від 184 до 276
Частота оновлення даних, Гц		~20	~1	1–5
Інтерфейси		I ² C	UART / TTL	RS-485
Ціна, \$		~10,5	~11	~34

В результаті проведеного аналізу модулів енергомоніторингу можна зробити наступні висновки. Реалізація моніторингу електромережі через SDM120-Modbus-MID та перетворювач рівнів, хоч і пропонує дещо вищу точність вимірювання, та є суттєво дорожчою у порівнянні із двома іншими варіантами. Енергомонітори M5Stack AC Measure Unit та PZEM-004T v3 коштують приблизно однаково. При цьому перший пропонує суттєво вищу частоту опитування, однак не вимірює частоту та коефіцієнт потужності. Другий здатен забезпечити стабільне опитування щосекунди, і це є достатнім для даного проєкту, а спроможність вимірювати більшу кількість показників є суттєвою перевагою. Отже, кращим варіантом реалізації моніторингу електромережі є використання модуля PZEM-004T. Варто відзначити, що через діапазон напруги живлення 4,5-5,5 В, його підключення до ESP32 має бути реалізованим через VIN-контакт.

2.6.3 Вибір датчика освітленості

В результаті огляду існуючих датчиків освітленості було відібрано декілька найбільш розповсюджених моделей для проведення порівняльного аналізу: МН Sensor Series LM393, BH1750 та OPT3001. Для визначення найкращого варіанту мають значення такі характеристики, як частота оновлення даних, інтерфейс, що використовується для передачі даних, напруга живлення та вартість.

МН Sensor Series LM393, зовнішній вигляд якого зображено на рисунку 2.11, є простим і дешевим модулем для визначення рівня освітленості, що побудований на основі фоторезистора GL5528 [34] та компаратора LM393.

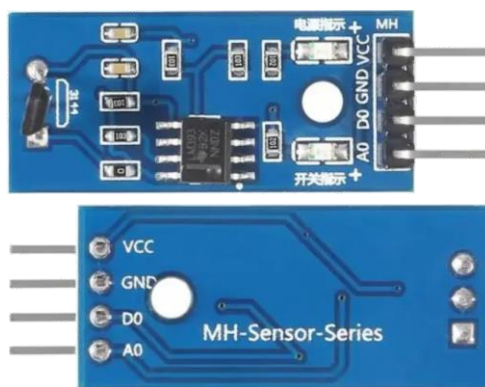


Рисунок 2.11 – Зовнішній вигляд МН Sensor LM393 (версія із чотирма контактами)

Він має два виходи: аналоговий, що відображає зміну опору LDR залежно від інтенсивності світла, та цифровий, який спрацьовує при досягненні встановленого порогового рівня. Датчик не є каліброваним, тому його діапазон вимірювання та величина похибки вимірювання не має чітко визначених значень. Допустима напруга живлення модуля знаходиться у діапазоні від 3,3 до 5 В, що дозволяє використовувати його з ESP32 та Arduino. Частота опитування обмежується швидкодією світлочутливого резистора, яка становить десятки-сотні мілісекунд, тому сенсор може забезпечити частоту опитування у 10-20 Гц. Калібрування порогового рівня для цифрового виходу датчика відбувається фізично за допомогою потенціометра. Напруга, що надходить з аналогового виходу датчика,

перетворюється на АЦП у відносну величину, яка після цього потребує додаткової логіки переведення у освітленість [47; 59].

BH1750 – це цифровий датчик освітленості, який на відміну від попереднього повертає показник освітленості у люксах. На рисунку 2.12 зображено зовнішній вигляд даного модуля на платі GY-302. Він підключається до мікроконтролера через I²C. Сенсор підтримує кілька режимів роботи: високої роздільності, низької роздільності та безперервних вимірювань, що дозволяє балансувати між роздільністю та частотою. Налаштування режимів відбувається програмно. Діапазон вимірювання становить від 0,11 (у режимі роботи із найбільшою роздільною здатністю) до приблизно 65535 люкс, а живлення сенсора відбувається у діапазоні 2,4 – 3,6 В. Точність вимірювань залежить від умов, однак для типових сценаріїв помилка складає приблизно 20 відсотків. Типовий час одного вимірювання становить 120 мс, але він може змінюватися у діапазоні 31–254 мс, відповідно частота опитування може сягати від 8 до 32 Гц.

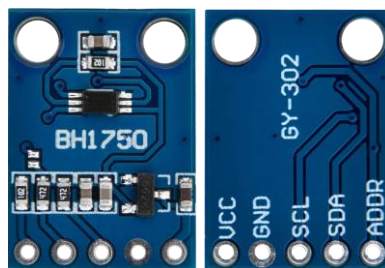


Рисунок 2.12 – Зовнішній вигляд BH1750 на платі GY-302

Як і попередній, датчик OPT3001 повертає одразу значення освітленості у люксах через інтерфейс I²C. Має ширший діапазон вимірювання – від 0,01 до приблизно 83 тисяч люкс, що дозволяє йому працювати і під сонячним світлом. Калібрування йде з коробки та включає вбудовану спектральну характеристику, дуже близьку до кривої чутливості людського ока, що робить вимірювання більш точними. Так, типова похибка у вимірюванні становить приблизно 2%. Суттєвою перевагою даного модуля є вбудована функція автоматичного вибору діапазону в залежності від освітленості середовища, що запобігає перенасиченню. Потребує

напруги живлення в діапазоні від 1,6 до 3,6 В. Максимальна частота опитування OPT3001 визначається його внутрішнім аналого-цифровим перетворювачем і дорівнює приблизно 10 Гц у стандартному режимі та 1,25 у високоточному. Даний датчик підтримує програмне налаштування, яке включає вибір режиму роботи, значення часу інтеграції, перемикання між ручною фіксацією діапазону та автоматичним його вибором. Зовнішній вигляд сенсора зображено на рисунку 2.13.

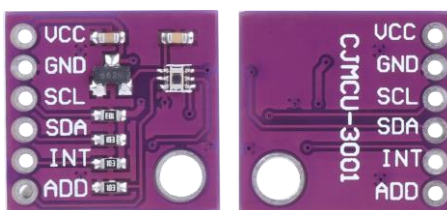


Рисунок 2.13 – Зовнішній вигляд OPT3001

У таблиці 2.3 наведено основні параметри розглянутих модулів вимірювання рівня освітленості.

Таблиця 2.3 – Основні характеристики модулів вимірювання освітленості

Назва	Діапазон, Люкс	Похибка, %	Напруга живлення, В	Частота, Гц	Інтерфейси	Ціна, \$
MH Sensor LM393 (4 контакти)	Некалібрований		від 3,3 до 5	~10-20	цифровий вихід + аналоговий вихід	~0,5
BH1750 (GY-302)	від 0,11 до 65535	~20	від 2,4 до 3,6	~8-32	I ² C	~3
OPT3001	від 0,01 до ~83000	~2	від 1,6 до 3,6	~1,25- 10	I ² C	~5

Їх порівняльний аналіз дає можливість зробити наступні висновки. Модуль LM393 доцільно використовувати лише для простих задач автоматизації, наприклад, вмикання світла при настанні темряви. Його недоліком є відсутність точних вимірювань у люксах, адже LDR має нелінійну характеристику та значні похибки залежно від температури і спектру світла. OPT3001 та BH1750 є

дорожчими, проте мають суттєво вищу точність вимірювань та зручність використання.

VN1750 може повертати зчитане значення із набагато більшою частотою у порівнянні із OPT3001, проте останній пропонує ширший діапазон та вищу точність вимірювання. Тому у якості модуля вимірювання освітленості було обрано OPT3001. Його допустима частота опитування, навіть у високоточному режимі, є достатньою для запропонованої системи розумного енергоспоживання, а енергоспоживання у активному режимі є суттєво нижчим у порівнянні із VN1750.

2.6.4 Вибір датчика присутності людини

В результаті огляду існуючих датчиків присутності людини було відібрано для порівняльного аналізу декілька найбільш популярних із використанням різних технологій. Серед параметрів, на які в першу чергу треба звертати увагу, є можливість розпізнавати нерухому людину, можливість налаштовувати чутливість, радіус дії, інтерфейси, якими датчик комунікує та вартість.

Першим розглянемо HC-SR501. Його зовнішній вигляд представлено на рисунку 2.14.

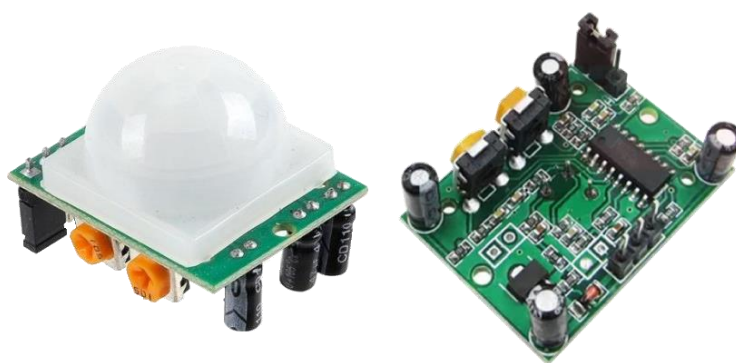


Рисунок 2.14 – Зовнішній вигляд HC-SR501

Даний датчик працює за принципом вимірювання інфрачервоного випромінювання, що означає, що він реєструє присутність, реагуючи на тепло живих істот. Радіус дії цього модуля знаходиться у діапазоні 3–7 метрів. Для

живлення йому необхідна напруга від 4,5 В до 20 В, а для передачі вихідного сигналу він використовує цифровий вихід. Налаштування HC-SR501 проводиться апаратно із використанням потенціометра. Цей датчик є найдешевшим із розглянутих.

Наступним розглянемо RCWL-0516, що працює за рахунок ефекту Доплера, тобто він реагує на переміщення у його радіусі дії предметів, особливо тих, що добре відбивають мікрохвильове випромінювання. Має оптимальний радіус дії 5–7 метрів. Діапазон напруги живлення складає від 4 до 28 В. Як і попередній датчик, RCWL-0516 має цифровий вихід для подання сигналу і конфігурується апаратно вручну. Для цього треба підключати пасивні компоненти у відповідні контакти, розташовані на задній частині плати. Так можна змінити діапазон та час спрацьовування, додати фоторезистор для спрацьовування лише в темряві та відрегулювати його чутливість. Зовнішній вигляд RCWL-0516 зображено на рисунку 2.15.

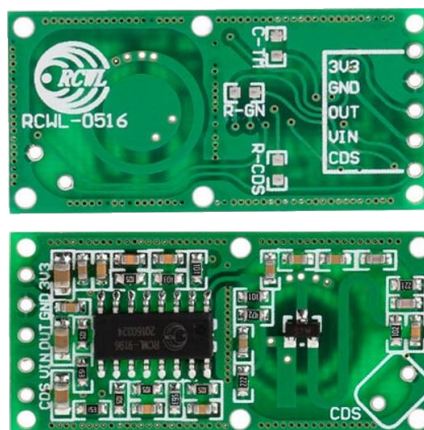


Рисунок 2.15 – Зовнішній вигляд RCWL-0516

Далі розглянемо датчик HLK-LD2410 (рисунок 2.16). Даний датчик посиляє неперервні хвилі із частотою 24 ГГц та оброблює отриманий радіолокаційний сигнал за допомогою вбудованих високоточних алгоритмів. Він працює від напруги у діапазоні 4,5–6 В та здатен точно виявляти людську присутність, навіть якщо людина не рухається, у діапазоні радіусу від 75 см до 5 м. HLK-LD2410 має, окрім цифрового виходу, інтерфейс UART, який дозволяє також вимірювати

відстань до виявленого об'єкта та програмно налаштовувати чутливість. Також для зручної взаємодії з ним створено мобільну та комп'ютерну версії додатків [36].

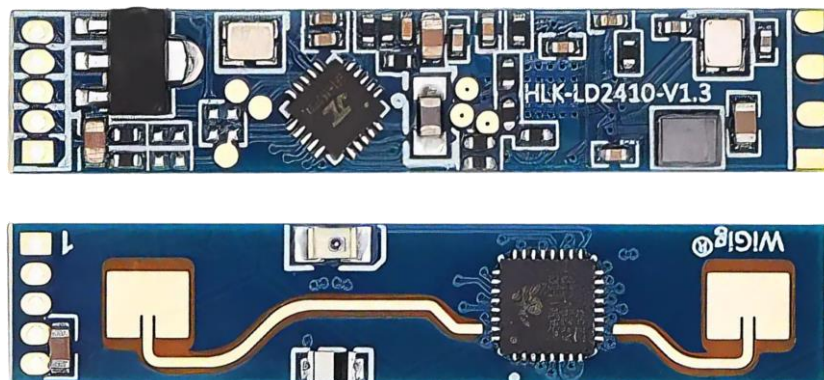


Рисунок 2.16 – Зовнішній вигляд HLK-LD2410

Ще одним варіантом може бути VL53L0X – датчик, що працює на основі інфрачервоного лазера та вимірює час, за який відбитий сигнал повертається назад. Лазер має частоту хвилі 940 нм і є безпечним для людських очей. Сприймає напругу живлення в діапазоні від 2,6 до 5,5 В. Максимальна відстань, з якої він стабільно зчитує зворотній сигнал, становить 2 м. VL53L0X використовує шину I²C для передачі числових значень обчисленої відстані та налаштування. Налаштування виконуються програмно через регістри датчика та включають період вимірювання, мінімальний рівень зворотнього сигналу, період імпульсів інфрачервоного лазера, і, насамкінець, режим вимірювання. Зовнішній вигляд готової плати Gy-VL53L0X V2 з розведенням та стабілізатором напруги на основі VL53L0X зображено на рисунку 2.17.

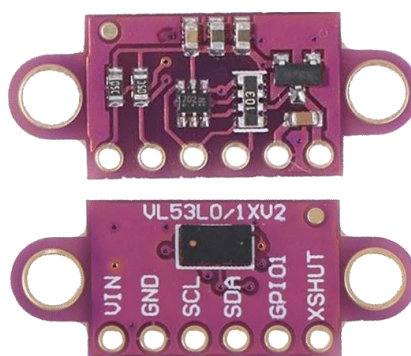


Рисунок 2.17 – Зовнішній вигляд плати Gy-VL53L0X V2

Останнім ми розглянемо JSN-SR04T. Даний датчик випромінює ультразвуковий сигнал частотою близько 40 кГц, чекає на повернення цієї хвилі та вимірює час проходження сигналу, після чого обчислюється відстань до об'єкта, що опинився на шляху сигналу. JSN-SR04T має мінімальний радіус дії 20 см, що означає, що він не реагує на об'єкти, що знаходяться ближче. Максимальний оптимальний радіус дії становить 6 м. Потребує напругу живлення у діапазоні 4,5–5,5 В. Взаємодія із JSN-SR04T може відбуватися за допомогою таймінгового механізму Trig/Echo, або через UART. У першому випадку контролюючий пристрій подає короткий логічний імпульс на вхід Trig, після чого на логічному виході Echo сигнал переходить у активний стан до моменту відбитого сигналу. Ширина отриманого з виходу Echo сигналу має бути переведена у відстань за простою формулою. У випадку взаємодії через UART модуль надсилає вже готове числове значення дистанції в сантиметрах. На рисунку 18 зображено зовнішній вигляд JSN-SR04T.

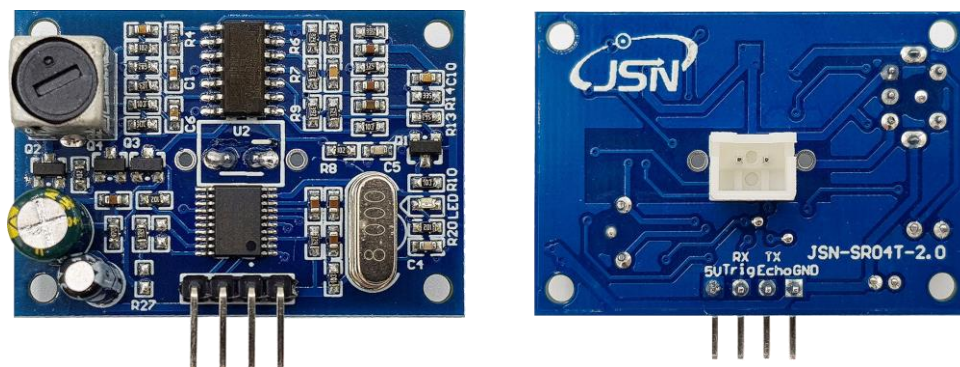


Рисунок 2.18 – Зовнішній вигляд JSN-SR04T

Усі основні параметри розглянутих датчиків присутності людини зведені у таблиці 2.4.

В результаті порівняльного аналізу можна зробити наступні висновки. Кожен із розглянутих датчиків підходить для дуже конкретної ситуації. Тому важливо брати до уваги недоліки технології, що використовується у кожному них. Так, наприклад, VL53L0X та JSN-SR04T лише вимірюють відстань до найближчого об'єкта у радіусі своєї дії, і, тому можуть реагувати на рух неживих об'єктів.

RCWL-0516 відслідковує лише рухомі об'єкти, тому не визначає нерухому присутність. HC-SR501 в свою чергу може хибно спрацьовувати, реагуючи на нагріті або ж освітлені неживі об'єкти. Таким чином, для системи розумного енергоспоживання у якості датчику присутності людини найкращим варіантом є HLK-LD2410, адже він має найточнішу із розглянутих технологію визначення присутності людини. З точки зору сумісності з ESP32 діапазон напруги живлення даного модуля є не найзручнішим, адже не включає 3,3В, проте обрана плата ESP32 DEVKIT V1 має вбудований стабілізатор напруги, що дозволяє безпечно жити LD2410 від 5 В. При цьому виходи UART (TX/RX) та цифровий вихід OUT працюють на рівні 3.3 В, що повністю сумісно з GPIO ESP32 без додаткових перетворювачів рівнів.

Таблиця 2.4 – Основні характеристики датчиків присутності людини

Назва	Напруга живлення, В	Радіус, м	Інтерфейси	Можливість конфігурації	Ціна, \$
HC-SR501	від 4,5 до 20	від 3 до 7	цифровий вихід	апаратно	~1
RCWL-0516	від 4 до 28	від 5 до 7	цифровий вихід	апаратно	~3
HLK-LD2410	від 4,5 до 6	від 0,75 до 5	UART + цифровий вихід	програмно	~5
VL53L0X	від 2,6 до 5,5	від 2 до 2	I ² C	програмно	~3
JSN-SR04T	від 4,5 до 5,5	від 20 до 6	цифрові Trigger/Echo контакти	—	~8

2.6.5 Вибір датчика температури

Для проведення порівняльного аналізу існуючих сенсорів температури розглянемо типові моделі: NTC, LM35, TMP36 та DS18B20. Основними характеристиками таких датчиків є діапазон та похибка вимірюваної температури, частота оновлення даних, інтерфейс, що використовується для передачі даних, напруга живлення та вартість.

Сенсори NTC є простими терморезисторами із негативним температурним коефіцієнтом (Negative Temperature Coefficient), що виробляються безліччю компаній по усьому світу. Вони виготовляються у різних формах, розмірах та класах точності вимірювання. Так, наприклад, найточніший клас має похибку при вимірюванні температури в 1% опору, що транслюється приблизно у $0,25\text{--}0,3\text{ }^{\circ}\text{C}$ за кімнатної температури, а найменш точний клас – 5% опору та відповідно – $1,2\text{--}1,5\text{ }^{\circ}\text{C}$. NTC вимірює в діапазоні від -55 до $+125\text{ }^{\circ}\text{C}$. Для запобігання самонагріванню при роботі такий датчик треба підключити в ділник напруги з відповідним резистором. Для переведення виміряного опору у температуру необхідно використовувати В-формулу або рівняння Steinhart–Hart. Також зазвичай такі сенсори потребують ручного калібрування, що включає замір зміщення та його врахування у формулі підрахунку зчитаної температури. NTC є найдешевшим рішенням із розглянутих сенсорів температури. Зовнішній вигляд даного датчика зображено на рисунку 2.19.



Рисунок 2.19 – Зовнішній вигляд NTC у епоксидній смолі

Датчики серії LM35 компанії Texas Instruments є аналоговими і, на відміну від попереднього сенсора, не потребують калібрування для більшості звичайних задач та важких обчислень для визначення зчитаної температури. Залежність напруги від температури є майже лінійною. Похибка у вимірюванні складає $0,5\text{ }^{\circ}\text{C}$ за кімнатної температури та може максимально вирости до $1\text{ }^{\circ}\text{C}$ на крайніх значеннях діапазону. Різні версії LM35 відрізняються діапазоном вимірювання та точністю. Так, наприклад, базова версія LM35 DZ (рисунок 2.20) вимірює лише в діапазоні від $0\text{ }^{\circ}\text{C}$ до $+100\text{ }^{\circ}\text{C}$, в той час як LM35 C / LM35 CA – від $-40\text{ }^{\circ}\text{C}$ до $+110\text{ }^{\circ}\text{C}$, а LM35 H – від $-55\text{ }^{\circ}\text{C}$ до $+150\text{ }^{\circ}\text{C}$. Для напруги живлення підходить діапазон від 4 до 30 В. Ці датчики мають аналоговий вихід, тому частота, з якою можна

розрядність вимірювання у діапазоні від 9 до 12 біт, що дозволяє обрати найбільш підходяще співвідношення між часом конверсії та роздільною здатністю вимірювання. Так, із обраною розрядністю у 9 бітів крок вимірювання становить 0,5 °C, проте час конверсії дуже короткий – 93,75 мс, а з розрядністю у 12 бітів роздільна здатність є найвищою – 0,0625 °C, проте час конверсії становить 750 мс. DS18B20 вимірює температуру у діапазоні від -55 °C до +125 °C та має похибку 0,5 °C у діапазоні від -10 °C до +85 °C, та 2 °C при крайніх значеннях діапазону вимірювання.

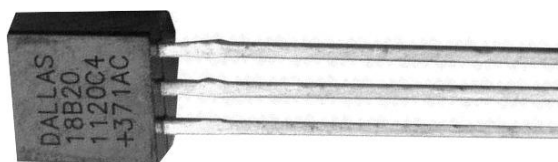


Рисунок 2.22 – Зовнішній вигляд DS18B20

У таблиці 2.5 наведено основні параметри всіх розглянутих сенсорів температури.

Таблиця 2.5 – Основні характеристики сенсорів температури

Назва	Діапазон, °C	Похибка, °C	Напруга живлення, В	Частота, Гц	Інтерфейс	Ціна, \$
NTC	від -55 до +125	0,5-3	—	аналоговий вихід		≤0,5
LM35C / LM35CA	від -40 до +110	0,5	від 4 до 30	аналоговий вихід		~1,5
TMP36	від -40 до +125	1	від 2,7 до 5,5	аналоговий вихід		~2
DS18B20	від -55 до +125	0,5	від 3 до 5,5	1,33-10,67	1-Wire	~2

Порівняльний аналіз датчиків температури дає можливість зробити низку висновків. NTC є найдешевшим рішенням, проте і найменш точним (похибка до 3°C), а також потребує лінеаризації, ручного калібрування та встановлення через

дільник напруги. Лінійка LM35 пропонує широкий асортимент сенсорів, серед яких найбільший інтерес викликають LM35C / LM35CA, адже LM35DZ не вимірює негативних температур, а широкий діапазон LM35H є надлишковим для вимірювання температури у приміщенні. Однак проблемою є те, що діапазон напруги живлення LM35 починається з 4 В, що унеможлиблює роботу через V3.3 у ESP32. Датчик TMP36 не має такої проблеми – він може житись від напруги 2,7-5,5 В. Однак він має вищу похибку вимірювання температури (1°C проти 0,5°C). Разом з тим DS18B20 є цифровим датчиком і потребує найменше додаткових обчислень. Має зручну робочу напругу для роботи з ESP32, широкий діапазон вимірювання температури, малу похибку та дає можливість налаштовувати розрядність. Крім того, інтерфейс 1-Wire дозволяє підключати до однієї шини ESP32 одразу велику кількість таких датчиків, що дозволяє здійснювати горизонтальне масштабування системи. Хоч DS18B20, на відміну від аналогових датчиків, і має обмеження частоти оновлення даних у вигляді часу, необхідного для конверсії температури, але це обмеження несуттєве: час, необхідний для конверсії у 12-розрядне число, становить 0,75 с, а необхідний для системи період оновлення даних – до 1 секунди. Таким чином, для системи моніторингу та управління енергоспоживанням у приміщенні оптимальним вибором буде сенсор DS18B20.

2.6.6 Вибір датчика вуглекислого газу

В результаті огляду існуючих модулів вимірювання концентрації летких органічних сполук та вуглекислого газу було відібрано декілька найбільш розповсюджених датчиків для проведення порівняльного аналізу, а саме CCS811, MH-Z19B та SCD30. До найбільш важливих характеристик таких датчиків відносяться: діапазон вимірювання, похибка, частота оновлення даних, інтерфейс передачі даних, напруга живлення та вартість.

CCS811 – це цифровий метал-оксидний датчик з вбудованим мікроконтролером, який вимірює концентрацію TVOC (Total Volatile Organic Compounds, загальна концентрація летких органічних сполук у повітрі) і eCO_2

(еквівалент вуглекислого газу) у робочому діапазоні від 400 до 8192 ppm. Важливо звернути увагу на те, що значення концентрації вуглекислого газу, що він повертає, є алгоритмічною оцінкою на основі TVOC, а не точним вимірним показником, тому похибка у визначенні концентрації вуглекислого газу може сягати і кількох десятків відсотків. Діапазон напруги живлення складає від 1,8 до 3,6 В, а типовим живленням у IoT-проектах є напруга у 3,3 В. CCS811 використовує інтерфейс I²C для передачі даних та дозволяє програмно налаштовувати режим роботи. Загалом існує чотири режими активної роботи цього датчика, а саме: вимірювання кожні 0,25 с, 1 с, 10 с та 60 с із поверненням сирих даних. Крім того, датчик підтримує режим сну. Максимальна частота, з якою даний модуль повертає оброблені значення, дорівнює 1 Гц і вона обмежена вбудованими режимами роботи. Опитувати його можна частіше, проте отримуючи при цьому повторні значення. Зовнішній вигляд CCS811 зображено на рисунку 2.23.

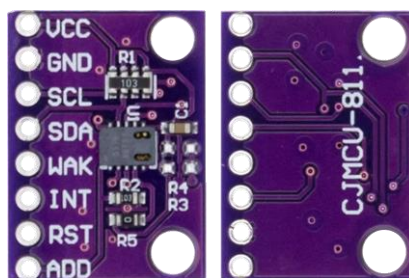


Рисунок 2.23 – Зовнішній вигляд CCS811

MH-Z19B є відносно недорогим модулем для вимірювання концентрації вуглекислого газу (рисунок 2.24). На відміну від CCS811 даний датчик функціонує за рахунок технології NDIR (Non-Dispersive Infrared, недисперсійне інфрачервоне випромінювання), тобто він посиляє інфрачервоне випромінювання частоти, що поглинається молекулами вуглекислого газу, вимірює інтенсивність світла, що не було поглинуте, та розраховує концентрацію газу за прямою пропорцією до цієї інтенсивності. Зазначений виробником робочий діапазон вимірювання становить від 400 до 5000 ppm, а величина похибки на ньому не перевищує 50 ppm + 5% від переведеного значення. Модуль потребує живлення в діапазоні 4,5-5,5 В. Він

обладнаний інтерфейсами UART та PWM, які пропонують альтернативні способи взаємодії. МН-Z19В має часову характеристику реакції приблизно у 120 секунд, проте звертатись до нього можна набагато частіше в рамках можливостей інтерфейсів виведення. UART дозволяє опитувати по декілька разів на секунду. У випадку підключення через PWM, частота звернення обмежується періодом виведення у 1004 мс, тобто вона спадає до приблизно 1 Гц [41].

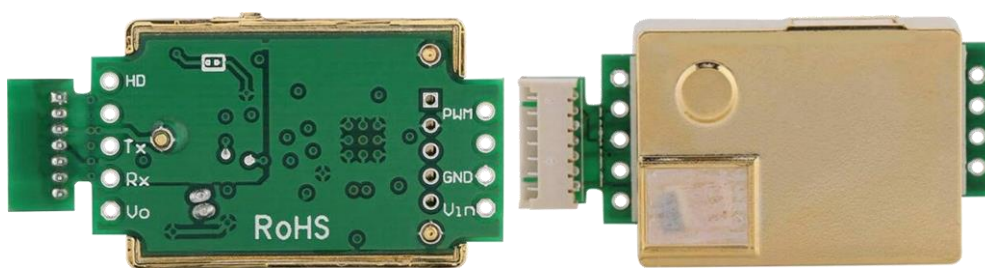


Рисунок 2.24 – Зовнішній вигляд МН-Z19В

Останнім розглянемо SCD30, зовнішній вигляд якого зображено на рисунку 2.25.

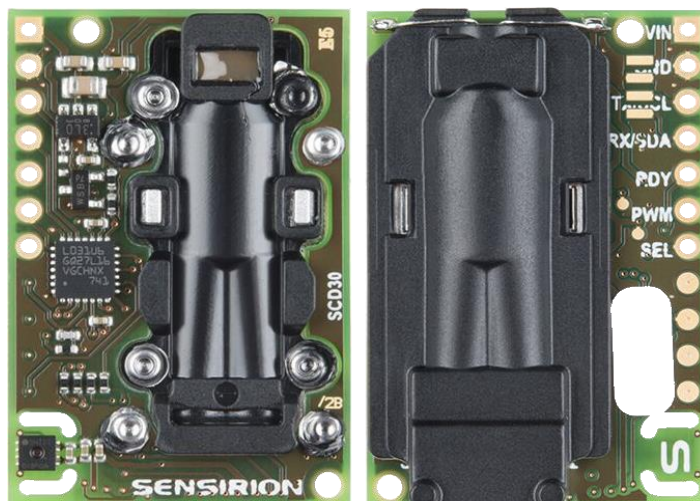


Рисунок 2.25 – Зовнішній вигляд SCD30

Це професійне і найдорожче рішення із розглянутих. Він, як і попередній, використовує NDIR-підхід для вимірювання концентрації вуглекислого газу, і крім цього має додаткові інтегровані термосенсор та датчик вологості. SCD30 має

повний діапазон вимірювання вуглекислого газу від 0 до 40000 ppm, з якого 400-10000 ppm є каліброваним відрізком. Датчик має точність величиною у 30 ppm + 3% від виміряного значення. Допустиме живлення даного модуля знаходиться у діапазоні від 3,3 до 5,5 В, що дає більше варіантів підключення до ESP32 у порівнянні із вищеописаними аналогами. Інтерфейси пристрою включають в себе I²C та UART для повноцінної роботи з вимірюваними параметрами та PWM, якщо цільовою є лише концентрація вуглекислого газу. За замовчуванням інтервал оновлення зчитаних значень встановлено у нижню межу, тобто 2 с, проте опитувати датчик можна із частотою, яку дозволить обраний інтерфейс. SCD30 пропонує широкий ряд програмних налаштувань, що включають величину інтервалу між вимірюваннями, значення висоти над рівнем моря, режим автокалібровки та багато інших.

У таблиці 2.6 наведено основні параметри розглянутих датчиків вуглекислого газу.

Таблиця 2.6 – Основні характеристики датчиків вуглекислого газу

Назва	Діапазон, ppm	Похибка	Напруга живлення, В	Частота, Гц	Інтерфейси	Ціна, \$
CCS811	від 400 до 8192	не визначена	від 1,8 до 3,6	≤ 1	I ² C	~9
MH-Z19B	від 400 до 5000	50 ppm + 5%	від 4,5 до 5,5	~0,0083	UART, PWM	~15
SCD30	від 400 до 10000	30 ppm + 3%	від 3,3 до 5,5	0,5	I ² C, UART, PWM,	~35

В результаті виконання порівняльного аналізу розглянутих датчиків концентрації вуглекислого газу можна зробити наступні висновки. Технологія, що використовується у модулі CCS811, дає занадто велику похибку у визначенні концентрації вуглекислого газу. SCD30 пропонує чудову гнучкість у використанні, є зручним та високоточним, проте висока вартість робить його малодоступним рішенням. Тому для системи розумного енергоспоживання, розраховану на широке

коло споживачів, доцільно у якості датчика вуглекислого газу обрати модуль МН-Z19В, адже він є прийнятним компромісом між точністю, ціною та простотою інтеграції. Хоч його частота оновлення даних і є найнижчою, однак для вимірювання зміни концентрації вуглекислого газу, що відбувається поступово, це є достатнім. Враховуючи діапазон напруги живлення даного модуля, його підключення до ESP32 відбувається через контакт VIN.

2.7 Проектування функціональної схеми та створення прототипу системи моніторингу та управління енергоспоживанням

В якості основи апаратного забезпечення системи моніторингу та управління енергоспоживанням була обрана плата DEVKIT V1 на основі мікроконтролера ESP32. Призначення її контактів наведено на рисунку 2.26.

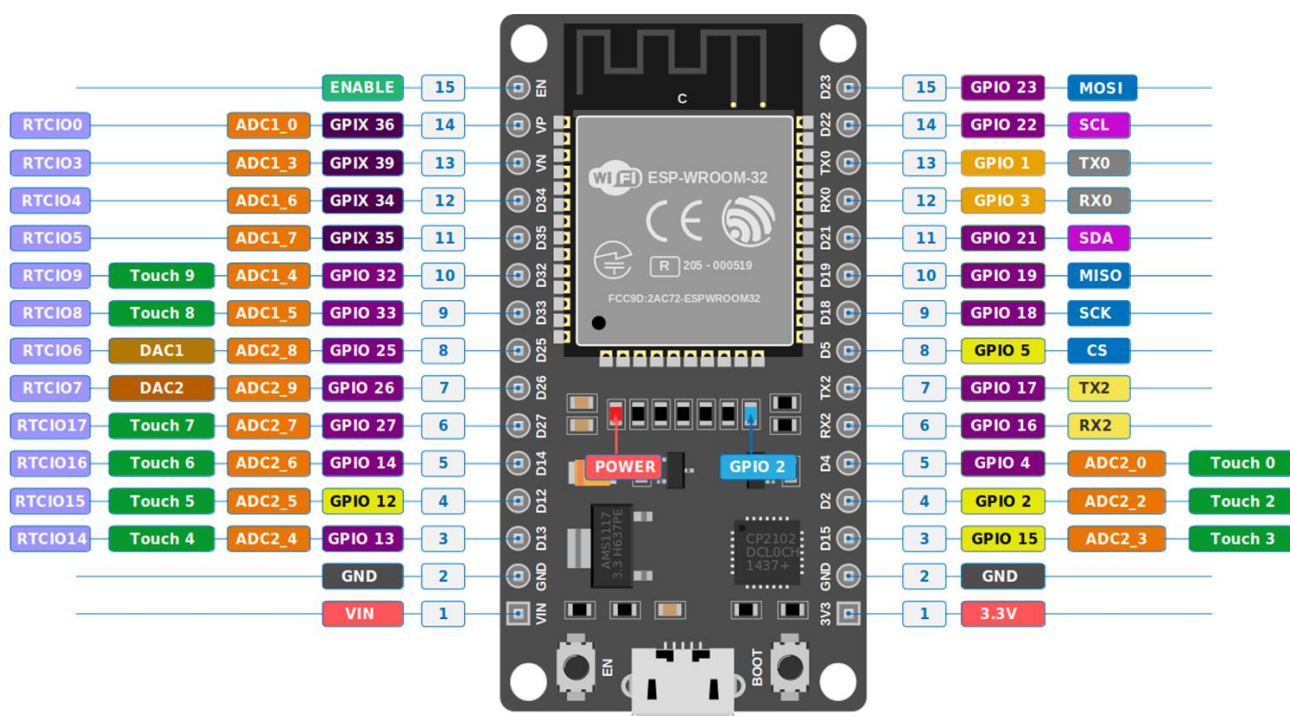


Рисунок 2.26 – Призначення контактів плати DEVKIT V1 на основі мікроконтролера ESP32

Для тестового стенду системи були використані наступні компоненти:

- макетна плата;

- плата DEVKIT V1 на основі мікроконтролера ESP32;
- перемички на 2,54 мм;
- слот для microSD;
- карта microSD;
- модуль реального часу DS3231SN;
- модуль енергомоніторингу PZEM-004T v3;
- датчик освітленості OPT3001;
- модуль контролю людської присутності HLK-LD2410;
- датчик температури DS18B20;
- датчик вуглекислого газу MH-Z19B;
- релейний модуль для управління виконавчими пристроями;

Під'єднання вищенаведених модулів до плати DEVKIT V1 відбувається таким чином:

- 1) Модуль енергомоніторингу PZEM-004T v3:
 - контакт GND до контакту GND мікроконтролера;
 - контакт VCC до контакту 3V3 мікроконтролера;
 - контакт TX до контакту RX2 мікроконтролера;
 - контакт RX до контакту TX2 мікроконтролера;
- 2) Датчик освітленості OPT3001:
 - контакт GND до контакту GND мікроконтролера;
 - контакт VCC до контакту 3V3 мікроконтролера;
 - контакт SCL до контакту D22 мікроконтролера;
 - контакт SDA до контакту D21 мікроконтролера;
- 3) Модуль контролю людської присутності HLK-LD2410:
 - контакт GND до контакту GND мікроконтролера;
 - контакт VCC до контакту VIN мікроконтролера;
 - контакт TX до контакту RX2 мікроконтролера;
 - контакт RX до контакту TX2 мікроконтролера;
- 4) Датчик температури DS18B20:
 - контакт GND до контакту GND мікроконтролера;

- контакт VDD до контакту 3V3 мікроконтролера;
- контакт DQ до контакту D4 мікроконтролера;

5) Датчик вуглекислого газу MH-Z19B:

- контакт GND до контакту GND мікроконтролера;
- контакт VIN до контакту VIN мікроконтролера;
- контакт TX до контакту RX2 мікроконтролера;
- контакт RX до контакту TX2 мікроконтролера;

6) Модуль реального часу DS3231SN:

- контакт GND до контакту GND мікроконтролера;
- контакт VCC до контакту 3V3 мікроконтролера;
- контакт SCL до контакту D22 мікроконтролера;
- контакт SDA до контакту D21 мікроконтролера;

7) Слот для microSD:

- контакт GND до контакту GND мікроконтролера;
- контакт 3V3 до контакту 3V3 мікроконтролера;
- контакт MOSI до контакту D23 мікроконтролера;
- контакт MISO до контакту D19 мікроконтролера;
- контакт CS до контакту D5 мікроконтролера;
- контакт CLK до контакту D18 мікроконтролера.

На рисунку 2.27 зображено функціональну схему системи, а на рисунку 2.28 – фотографія її прототипу.

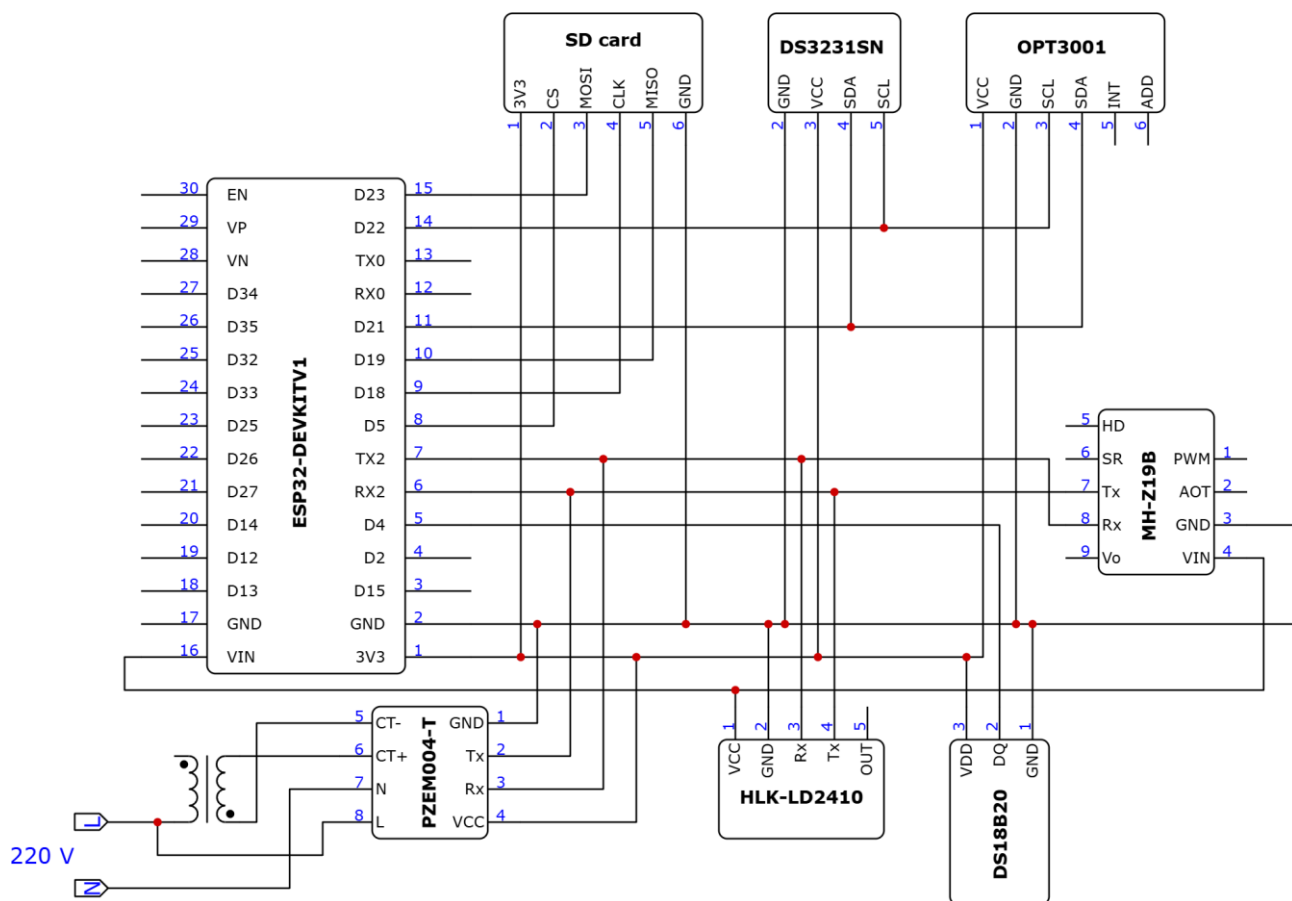


Рисунок 2.27 – Функціональна схема системи моніторингу та управління енергоспоживанням

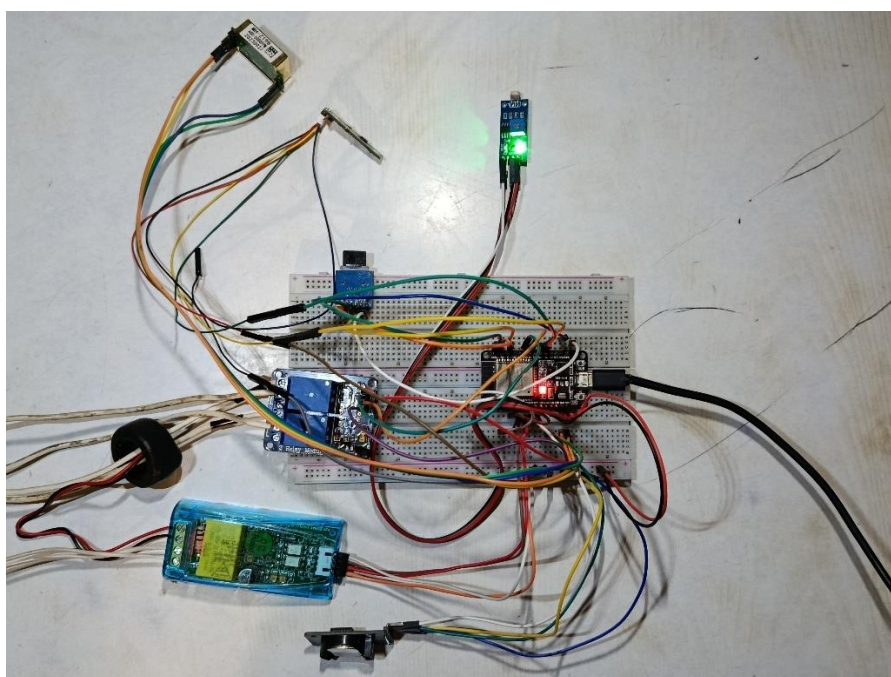


Рисунок 2.28 – Прототип системи моніторингу та управління енергоспоживанням

2.8 Програмне забезпечення системи моніторингу та управління енергоспоживанням

2.8.1 Огляд мов програмування та середовищ розробки для мікроконтролерів

У сучасних IoT-проєктах вибір інструментів розробки програмного забезпечення має ключове значення. Від цього залежить не лише швидкість створення прототипу, а й подальша підтримка коду, масштабованість системи та ефективність використання апаратних ресурсів.

У якості мов програмування найчастіше використовують C, C++ та MicroPython. Мова C є низькорівневою мовою, яка забезпечує максимальний контроль над апаратними ресурсами та високу продуктивність програм. Вона дозволяє безпосередньо працювати з пам'яттю, регістрами та периферією, проте працюючи із C розробнику доводиться вручну керувати багатьма аспектами, які у вищих мовах автоматизовані. C++ розширює можливості C об'єктно-орієнтованим підходом, чим полегшує створення структурованих проєктів та їх масштабування. MicroPython в свою чергу представляє підхід, орієнтований на простоту та швидке прототипування. Ця мова зосереджена на легкому синтаксисі та можливості швидко перевіряти ідеї без складної конфігурації, що робить її привабливою для навчання та експериментів. Проте MicroPython суттєво поступається C і C++ за продуктивністю [49]. Так, C++ займає проміжну позицію між продуктивністю, близькою до C, та зручним стилем програмування, тому C++ було обрано для розробки програмного забезпечення мікроконтролера у даному проєкті.

Для мікроконтролерів ESP32 існує декілька популярних середовищ розробки, що відрізняються широтою функціоналу, зручністю роботи та складністю освоєння. Розглянемо три таких рішення, а саме: Arduino IDE, PlatformIO та ESP-IDF.

Arduino IDE вважається найпростішим варіантом для початківців. Завантаження прикладів і бібліотек є інтуїтивно зрозумілим, а їх кількість у вільному доступі є великою. Для ESP32 вже існують сотні готових бібліотек, які

дають змогу швидко підключати датчики, модулі зв'язку чи периферійні пристрої. У даному середовищі використовується мова програмування C++, але у спрощеній формі, з прихованими від користувача низькорівневими деталями. З іншого боку Arduino IDE має обмежені можливості відлагодження та організації великих проєктів. При масштабуванні структура коду швидко стає незручною, а робота з кількома файлами одночасно ускладнена [15].

PlatformIO – це більш сучасний інструмент розробки, який окрім власного застосунку ще реалізований у вигляді розширення до середовища Visual Studio Code. Він підтримує C і C++, має зручну систему керування бібліотеками та інтеграцію з системами контролю версій, такими як Git. PlatformIO дозволяє працювати з різними фреймворками, включаючи Arduino Core та ESP-IDF, що робить його універсальним інструментом. На відміну від Arduino IDE, у ньому можна легко налаштовувати проєкти з багатьма файлами, а також використовувати автодоповнення коду та розширені інструменти для тестування. Серед недоліків можна виділити складніший процес початкового налаштування, а також вищі вимоги до обчислювальних ресурсів.

ESP-IDF – це офіційний фреймворк від компанії Espressif, розроблений спеціально для їх мікроконтролерів. Він надає доступ до всіх функцій мікроконтролера та дозволяє створювати максимально оптимізовані програми. Основними мовами програмування є C та C++, що дає можливість працювати з апаратурою на низькому рівні. ESP-IDF забезпечує повну офіційну підтримку, регулярні оновлення, приклади коду і детальну документацію. Проте дане середовище має високий поріг входу та написання коду займає більше часу, оскільки розробнику доводиться вручну керувати конфігурацією проєкту, розподілом пам'яті, ініціалізацією периферії та іншими аспектами, на відміну від Arduino IDE та PlatformIO [30].

Отже, кожне із середовищ має свої сильні та слабкі сторони. Arduino IDE є хорошим інструментом для початківців, ESP-IDF – найбільш професійне із розглянутих рішень. Однак, для збалансованого поєднання простоти, зручності використання та гнучкості оптимальним вибором є PlatformIO. Таким чином, для

проєкту системи розумного енергоспоживання, апаратною основою якого виступає плата DEVKIT V1 на основі мікроконтролера ESP32, у якості середовища розробки програмного забезпечення варто обрати PlatformIO, а в якості мови програмування – C++.

2.8.2 Серверна частина програмного забезпечення

При створенні серверної частини програмного забезпечення мікроконтролера було дотримано модульну структуру проєкту. Так, кожен файл містить окрему логічну частину функціоналу, що спрощує подальшу роботу над кодом. На рисунку 2.29 зображено схему взаємозв'язків модулів програми мікроконтролера.

Розглянемо кожен із наведених на схемі модулів.

Файл `main.cpp` містить обов'язкові точки входу програми – функції `setup` та `loop`. У функції `setup` описані одноразові команди, що мають бути виконані на початку роботи системи, а саме ініціалізація інтерфейсів UART, I²C та SPI, релейних модулів, датчиків та RTC, файлової системи та бази даних, вивантаження записаних у JSON-файлах налаштувань та сценаріїв у глобальні змінні, ініціалізація точки доступу, веб-сервера та вебсокет-каналу. Функція `loop` у свою чергу є головним циклом, у якому код виконується поки мікроконтролер не буде вимкнено. У ній з певними часовими інтервалами відбувається зчитування значень з датчиків, вимірювальних модулів та RTC, їх обробка та відповідна реакція в залежності від поточного часу, присутності людини, рівнів показників освітленості, концентрації вуглекислого газу та температури, збереження у базу даних останніх зчитаних значень показників та станів підключених пристроїв, а також їх надсилання у вебсокет-канал.

В модулі `setup.cpp` описано функції, що викликаються у файлі `main.cpp` в блоці `setup` і реалізують налаштування мікроконтролера для роботи із його внутрішніми системами. Вони містять ініціалізацію файлової системи, точки доступу Wi-Fi, DNS-сервера, веб-сервера та вебсокет-каналу. Даний функціонал

було винесено в окремий блок, щоб `main.cpp` не був громіздким і містив лише виклики описаних у даному модулі функцій.

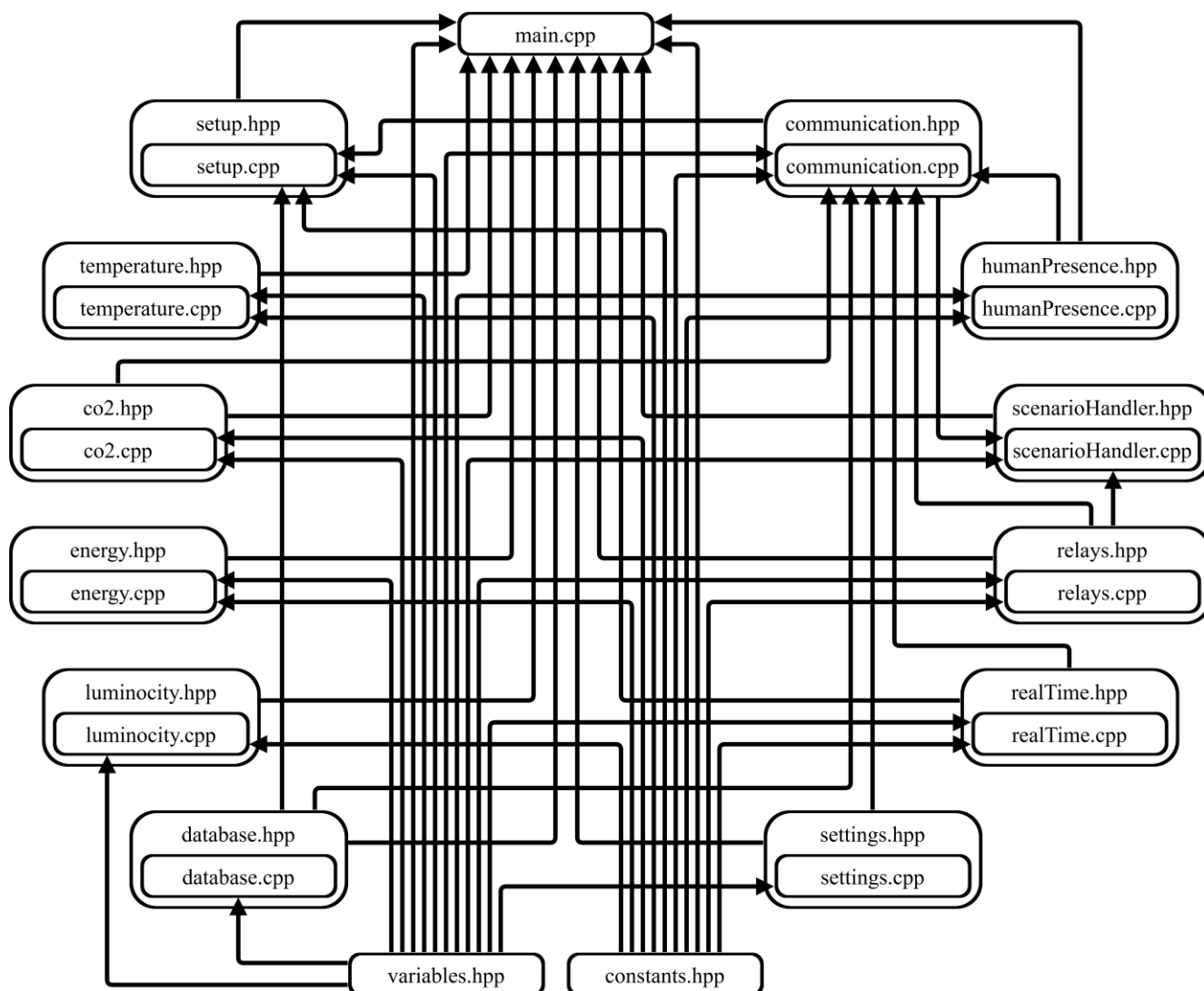


Рисунок 2.29 – Схема взаємозв'язків модулів серверної частини програмного забезпечення системи моніторингу та управління енергоспоживанням

Модуль `settings.cpp` описує функції для взаємодії між програмою мікроконтролера та файлами із налаштуваннями `settings.json` та `scenarios.json`. У файлі `settings.json` зберігаються загальні налаштування системи, такі як IP-адреса, шлюз, підмережа, DNS-адреса, SSID мережі та пароль від неї, а також інтервали, з якими система зчитує показники та зберігає їх у базу даних. У `scenarios.json` в свою

чергу записані налаштування, пов'язані із сценаріями, що пропонує система. За допомогою них користувач може налаштовувати автоматичну активацію виконавчих пристроїв при досягненні відповідним зчитаним показником зазначеного користувачем порогу, визначати активні та неактивні періоди роботи виконавчих пристроїв тощо.

Файл `communication.cpp` містить функцію-оброблювач подій, що надходять по вебсокет-з'єднанню з клієнтської сторони. Тобто, при надходженні запиту від клієнта цей оброблювач визначає, яку функцію необхідно викликати, і відповідно, яку дію виконає мікроконтролер. Після цього, якщо виконанню нічого не завадило, оброблювач надсилає у вебсокет-канал відповідь, яка несе код зворотньої команди та може містити файли. Надсилання повідомлення у вебсокет-канал реалізовано окремою функцією, яка теж описана у даному модулі. Вона необхідна, щоб оновлювати на вкладці «Home» індикатори зчитаних параметрів та станів підключених пристроїв при їх автоматичному перемиканні.

В файлі `database.cpp` описано функції, які виконують запити до бази даних на зовнішньому накопичувачі мікроконтролера. Вони періодично викликаються для регулярного запису зчитаних сенсорами показників та станів виконавчих пристроїв, або коли клієнт робить запит на перегляд графіків за певний період.

Заголовковий файл `variables.hpp` містить у форматі змінних інформацію, необхідну для швидкої роботи системи. У цих змінних продубльовано вищезгадані налаштування та сценарії, що містяться у файлах JSON, щоб не було необхідності проводити їх парсинг більше одного разу. При завантаженні системи або модифікації користувачем налаштувань та сценаріїв оновлений вміст JSON-файлів зберігається у змінних модуля `variables.hpp` і надалі робота програми відбувається саме з ними. Також у ньому зберігаються поточні стани релейних модулів для відслідковування часу роботи виконавчих пристроїв та останні значення, що передали вимірювальні модулі та годинник реального часу.

У заголовковому файлі `constants.hpp` у форматі констант збережено інформацію, необхідну для роботи програми, яка не передбачає свої зміни користувачем. Наприклад, до цієї інформації відносяться номери контактів

мікроконтролера для зручної роботи із підключеними модулями та кодові номери команд запитів та відповідей.

Модулі `realTime.cpp`, `humanPresence.cpp`, `co2.cpp`, `temperature.cpp`, `energy.cpp` та `luminosity.cpp` описують роботу із модулями реального часу, виявлення присутності людини, вимірювання концентрації вуглекислого газу, температури, параметрів електромережі та рівня освітленості відповідно. Окрім ініціалізації модулів та зчитування показників вони ще реалізують програмне налаштування та калібрування датчиків, що це дозволяють. Наприклад, датчик вуглекислого газу МН-Z19В надає можливості програмного управління калібруванням, а також налаштування його діапазону вимірювання, що реалізовано у функціях програмного модуля `co2.cpp`.

У модулі `relays.cpp` описано функції, за допомогою яких мікроконтролер може керувати підключеними пристроями. Так, при зміні стану пристрою за допомогою цих функцій відповідним чином оновлюється глобальна структура з модулю `variables.hpp`, що гарантує очікувану поведінку системи.

Файл `scenarioHandler.cpp` містить функцію, яка реалізує задані користувачем сценарії. Ця функція оброблює значення, що передали датчики, вимірювальні модулі та годинник реального часу, після чого на їх основі визначає нові стани підключених пристроїв та переводить їх у ці стани.

2.8.3 Огляд варіантів реалізації прикладного рівня системи

Для організації взаємодії з користувачем системи на базі мікроконтролера найчастіше розглядаються два підходи. Перший варіант полягає у створенні веб-інтерфейсу, доступного безпосередньо з браузера будь-якого пристрою, що має підключення до мережі. Другий варіант передбачає розробку окремого мобільного чи настільного додатку, який встановлюється на пристрій користувача і взаємодіє з мікроконтролером через мережу або локальні інтерфейси. Розглянемо особливості, переваги та обмеження обох підходів.

Рішення з веб-інтерфейсом є універсальним, адже він працює через браузер

незалежно від операційної системи пристрою користувача. Завдяки цьому така реалізація не потребує встановлення та витрат постійної пам'яті на стороні клієнта, а також гарантує кросплатформеність. Крім того, оновлення та підтримка такого інтерфейсу значно спрощуються: достатньо змінити код на серверній стороні, після чого всі користувачі одразу отримають актуальну версію.

Рішення з додатком, у свою чергу, дозволяє глибше інтегруватися з операційною системою пристрою користувача. Кожну версію додатку можна оптимізувати для конкретної платформи, беручи до уваги її ресурси та особливості, наприклад, системне кешування, локальне збереження даних або доступ до сенсорів. Завдяки цьому додаток може працювати швидше та стабільніше, а також з'являється можливість реалізувати його роботу при обмеженому підключенні до мережі.

Узагальнюючи можна зробити наступний висновок. Створення та підтримка версій додатків для різних платформ (Android, iOS, Windows тощо) потребує більше часу і ресурсів розробника, а встановлення додатка займає постійну пам'ять пристрою користувача. З іншого боку, рішення з веб-інтерфейсом забезпечить доступ до системи з будь-якого пристрою через браузер, мінімізує витрати на розробку та обслуговування і дозволить уникнути проблем із сумісністю. Таким чином, для реалізації прикладного рівня локальної системи моніторингу та управління енергоспоживанням кращим вибором буде веб-підхід, адже він поєднує в собі простоту використання, легкість оновлення та універсальність.

2.8.4 Клієнтська частина програмного забезпечення

Прикладний рівень системи було реалізовано із використанням базових інструментів веб-розробок, таких як HTML (HyperText Markup Language, або мова розмітки гіпертексту), CSS (Cascading Style Sheets, або каскадні таблиці стилів) та JavaScript [32]. Варто відзначити, що в цілях економії флеш-пам'яті мікроконтролера було зроблено вибір на користь розробки власних динамічних елементів замість використання готових бібліотек або фреймворків.

При створенні веб-інтерфейсу користувача було дотримано модульну структуру проєкту. Тобто під будь-який окремий за логікою функціонал виділено власний файл, а за необхідності його використання у іншому файлі він імпортується. Така структура є дуже простою для сприйняття, тому дозволяє розробнику швидко орієнтуватися у проєкті і, відповідно, пришвидшує подальший процес розробки. На рисунку 2.30 зображено схему взаємозв'язків між модулями клієнтської частини програмного забезпечення системи.

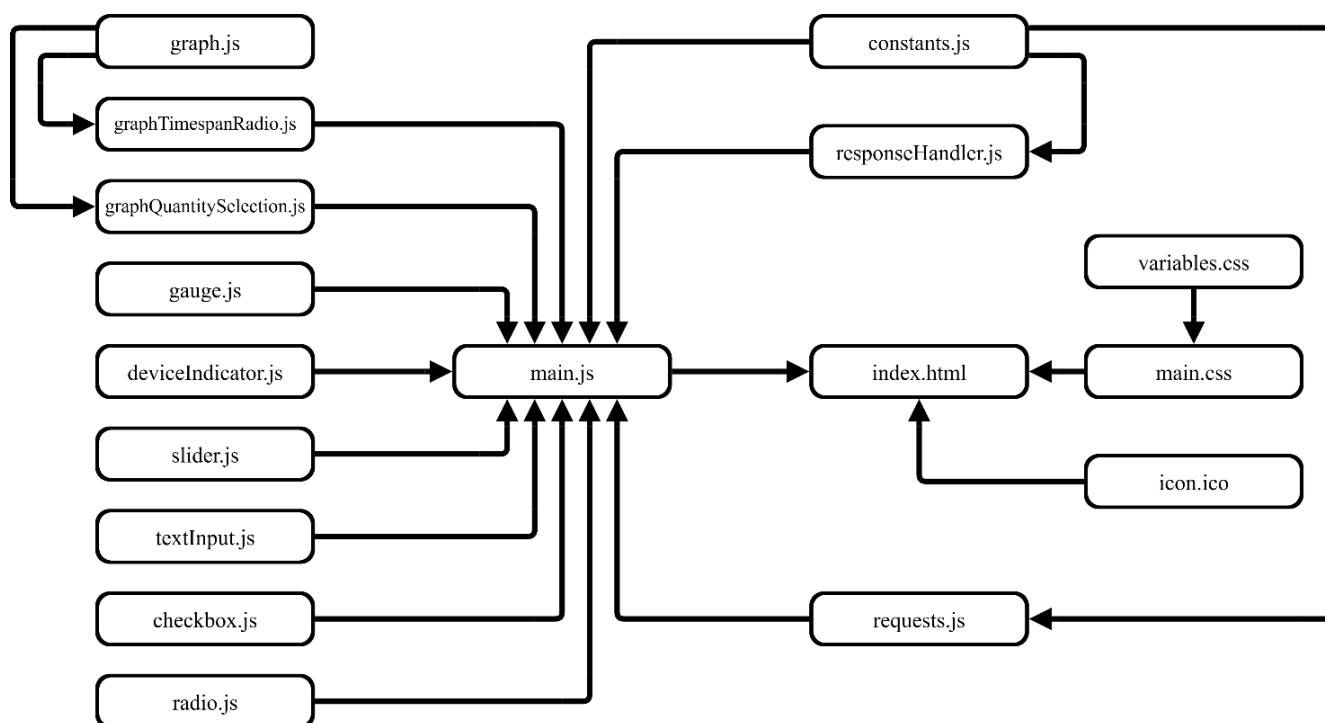


Рисунок 2.30 – Схема взаємозв'язків модулів клієнтської частини програмного забезпечення системи моніторингу та управління енергоспоживанням

Розглянемо функціонал кожного з модулів, наведених на схемі.

Задля вищої швидкості роботи інтерфейсу та кращого користувацького досвіду було використано підхід SPA (single page application, або односторінковий додаток), тому розмітку сторінки реалізовано у одному файлі index.html. Таким чином, при відкритті веб-інтерфейсу користувацький пристрій (комп'ютер, планшет або телефон) має одразу завантажити повний обсяг даних, але при

подальшому переході між вкладками додатку будуть відсутні затримки через довантаження файлів. Комунікація із серверною стороною відбувається через вебсокетне з'єднання і передбачає пересилання повідомлень із номерами команд та JSON-файлів.

У файлі `main.css` описано стилі, що застосовуються до розмітки у `index.html`, а в `variables.css` – змінні, що багаторазово використовуються у `main.css` для збереження однакових величин, шрифтів, кольорів, тощо. Файл `icon.ico` містить стиснутий логотип веб-сторінки, що відображається у панелі закладок браузера.

Модуль `constants.js` містить у вигляді констант інформацію, таку як кодові номери команд запитів та відповідей для їх розпізнавання у оброблювачі подій та граничні значення фізичних величин, що необхідні для відображення їх поточних показників та побудови графіків тощо.

У `requests.js` зібрано функції, що виконують запити до серверної сторони системи. Наприклад, у ньому описано: функцію, яка при завантаженні веб-інтерфейсу під'єднує до вебсокет-каналу оброблювач подій, які надходять з серверу; функцію, що робить запит на отримання записів від серверу з його бази даних за певний період для побудови графіку; функції, які вмикають та вимикають обраний пристрій тощо.

У файлі `responseHandler.js`, в свою чергу, записано функцію-оброблювач повідомлень, що надходять з серверу. Вона оброблює початково отримані налаштування системи, регулярно оновлює надіслані мікроконтролером показники датчиків та стани виконавчих пристроїв, отримує та оброблює записи бази даних для побудови графіків тощо.

Головною точкою входу інтерактивного функціоналу веб-інтерфейсу є файл `main.js`. У ньому відбувається початкова ініціалізація вебсокетного з'єднання та обробника серверних повідомлень, створення динамічних компонентів, таких як індикатори, поле для побудови графіку, поля введення, та прив'язка до них відповідних функцій, що очікують даних від серверу.

У файлах `graph.js`, `graphTimespanRadio.js`, `graphQuantitySelection.js`, `gauge.js`, `deviceIndicator.js`, `slider.js`, `textInput.js`, `checkbox.js` та `radio.js` описано функції для

ініціалізації графу, полів для введення часового проміжку та величини для побудови графу, індикаторів вимірних величин та станів підключених виконавчих пристроїв, а також повзункових, текстових, булевих полів та полів-перемикачів відповідно. Процедуральна генерація цих компонентів гарантує, що вони матимуть однакову структуру, що полегшує процес взаємодії із ними та захищає від неочікуваної поведінки. Для створення індикаторів та графів існують готові фреймворки та бібліотеки у відкритому доступі, проте для даної системи задля економії флеш-пам'яті було розроблено власні програмні модулі із відповідним функціоналом – `gauge.js` та `graph.js`.

ВИСНОВКИ ДО РОЗДІЛУ 2

В першу чергу під час розробки системи моніторингу та управління енергоспоживанням в приміщенні були визначені основні вимоги до неї, а саме: масштабованість та гнучкість архітектури, здатність до інтеграції та сумісність, достатня потужність апаратного забезпечення, можливість автономної роботи, надійність, точність та оперативність моніторингу, безпека, зручність використання, економічність.

Проведений аналітичний огляд існуючих підходів до формування архітектури систем інтернету речей показав, що для системи моніторингу та управління енергоспоживанням у приміщенні оптимальним вибором є 4-рівнева архітектура, адже вона пропонує такі якості, як простоту розгортання й використання, низьку вартість, достатній рівень безпеки та можливість подальшого розширення та розвитку.

Враховуючи зазначені вище вимоги був проведений порівняльний аналіз існуючих систем управління базою даних, протоколів зв'язку, апаратних модулів та засобів розробки. У якості бази даних було обрано SQLite3 через те, що вона є легким та компактним реляційним рішенням, яке не вимагає розгортання серверної інфраструктури. Серед протоколів бездротового зв'язку найбільш придатним для системи моніторингу та управління енергоспоживанням в приміщенні виявився Wi-Fi – він забезпечує оптимальний з точки зору потреб такої системи баланс між швидкістю, простотою інтеграції, а також пропонує велику кількість апаратних рішень на його базі.

До кожної складової апаратного забезпечення системи було висунуто релевантні вимоги щодо їх технічних характеристик та можливостей, на основі яких проводилось порівняння. При виборі мікроконтролерної плати у якості найважливіших критеріїв виступали вартість, потужність процесорного блоку, об'єми різних видів пам'яті, кількість контактів та інтерфейси, що вони підтримують. Серед розглянутих варіантів найкраще поєднання наведених

параметрів має плата DEVKIT V1 на основі ESP32, тому її було обрано за основу апаратного забезпечення системи.

Для вимірювальних модулів порівняльними факторами виступили діапазон та похибка вимірювання відповідних величин, інтерфейси, що вони підтримують, частота оновлення показників та вартість. Так, у якості модуля енергомоніторингу було обрано PZEM-004T v3, датчика освітленості – OPT3001, датчика присутності людини – HLK-LD2410, датчика температури – DS18B20, датчика вуглекислого газу – MH-Z19B. Кожен із цих модулів є оптимальним компромісом між низькою вартістю та потрібними характеристиками (точністю вимірювання, частотою опитування та зручністю взаємодії).

Задля створення програмного забезпечення (прошивки мікроконтролера та інтерфейсу користувача) були оглянуті можливі варіанти засобів розробки та мов програмування. Обґрунтованим вибором для написання серверної частини програмного забезпечення стали фреймворк PlatformIO та мова C++, а для реалізації прикладного рівня архітектури був обраний веб-підхід із чистим JavaScript.

На макетній платі з використанням вищеописаних апаратних модулів був зібраний прототип системи моніторингу та управління енергоспоживанням в приміщенні, а також перевірена її працездатність та відлагоджено програмне забезпечення мікроконтролера.

3 ОСОБЛИВОСТІ ФУНКЦІОНУВАННЯ СИСТЕМИ

3.1 Алгоритм роботи системи

Для початку роботи системи необхідно подати на її вхід живлення змінного струму напругою 220 В. Після цього мікроконтролер проводить дії, що описані у функції `setup` у прошивці, а саме ініціалізацію файлової системи, бази даних та модуля реального часу, парсинг файлів налаштувань та сценаріїв, ініціалізація точки доступу Wi-Fi, веб-сервера та вебсокетного каналу. Після того, як відпрацюють перелічені у `setup` функції, починає виконуватись основний цикл `loop`, який не зупинить своєї роботи до припинення подачі живлення.

У циклі `loop` є два окремих інтервали, за якими регулярно виконуються певні дії, а саме:

- перший інтервал, з яким відбувається виявлення присутності людини, зчитування поточного часу, показників стану мікроклімату в приміщенні (температури, освітленості, концентрації вуглекислого газу) та параметрів електромережі (активна потужність, сила струму, напруга, частота та коефіцієнт потужності), а також увімкнення або вимкнення під'єднаних до системи приладів в залежності від присутності людини, часу, перетинання показниками заданих сценаріями порогів;

- другий інтервал, з яким відбувається запис зчитаних параметрів та станів підключених пристроїв із міткою поточного часу у базу даних задля можливості подальшого перегляду статистики за відповідний період.

Комунікація між клієнтською та серверною сторонами відбувається шляхом надсилання кодів команд та JSON-файлів. Якщо клієнт робить запит на ручну зміну стану підключеного приладу, редагування сценаріїв або налаштувань або отримання записів з бази даних за певний період для побудови графіку, серверний оброблювач подій, що був прив'язаний до вебсокет-каналу, розпізнає цей запит і виконує відповідну команду. Після цього він повертає відповідь, яка сигналізує про успішність або ж неуспішність виконання відповідної дії та може містити JSON-файл із запитаними даними. Отримавши повідомлення від сервера по вебсокет-

з'єднанню, клієнтський оброблювач подій розпізнає його та в реальному часі виконує відповідні зміни у веб-інтерфейсі, наприклад, оновлює значення індикаторів на головній сторінці, будує графік, змінює активну опцію радіо-перемикача у налаштуваннях тощо.

3.2 Робота з веб-інтерфейсом

Веб-інтерфейс користувача поділено на чотири логічно-відокремлені вкладки: «Home», «Analytics», «Scenarios» та «Settings».

При завантаженні веб-сторінки користувача зустрічає вкладка «Home», зовнішній вигляд якої зображено на рисунку 3.1. На цій сторінці представлено індикатори величин, що зчитуються під'єднаними до мікроконтролера сенсорами та вимірювальними модулями (Sensor values), а саме показників стану мікроклімату в приміщенні (температури, освітленості, концентрації вуглекислого газу) та параметрів електромережі (активна потужність, сила струму, напруга, частота та коефіцієнт потужності); а також індикатори станів підключених до системи пристроїв (Device states), які можуть бути в активному або неактивному стані. На рисунку 3.2 наведено зовнішній вигляд вкладки «Home» під час одночасної роботи декількох пристроїв.

Дані оновлюються у реальному часі із інтервалом, який налаштовує користувач на вкладці «Settings». При наведенні на кожен індикатор з'являється підпис із назвою відповідної величини або ж пристрою. Індикатори станів підключених пристроїв додатково слугують кнопками, що дозволяють користувачеві вручну перемикати їх стан із пріоритетом вищим, ніж в автоматичних інструкцій системи.

Вкладка «Analytics» дає користувачу доступ до аналітичного функціоналу системи. Вона дозволяє переглядати збережену у базі даних статистику щодо присутності людини в приміщенні, показників якості внутрішнього середовища, параметрів електромережі та активності під'єднаних до системи пристроїв за обраний період часу.

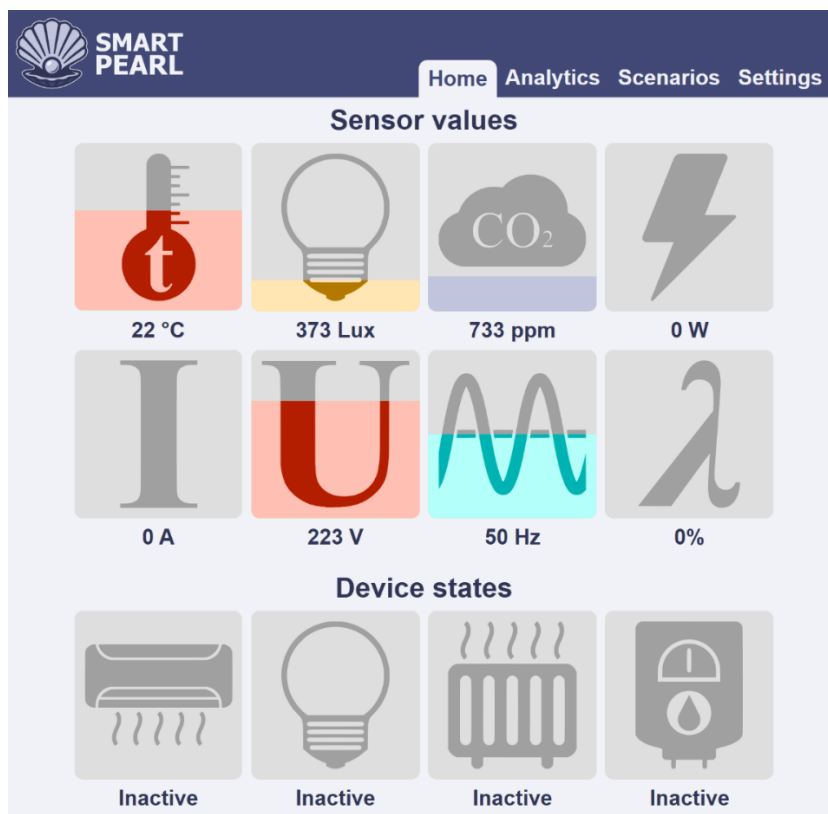


Рисунок 3.1 – Зовнішній вигляд вкладки «Home»

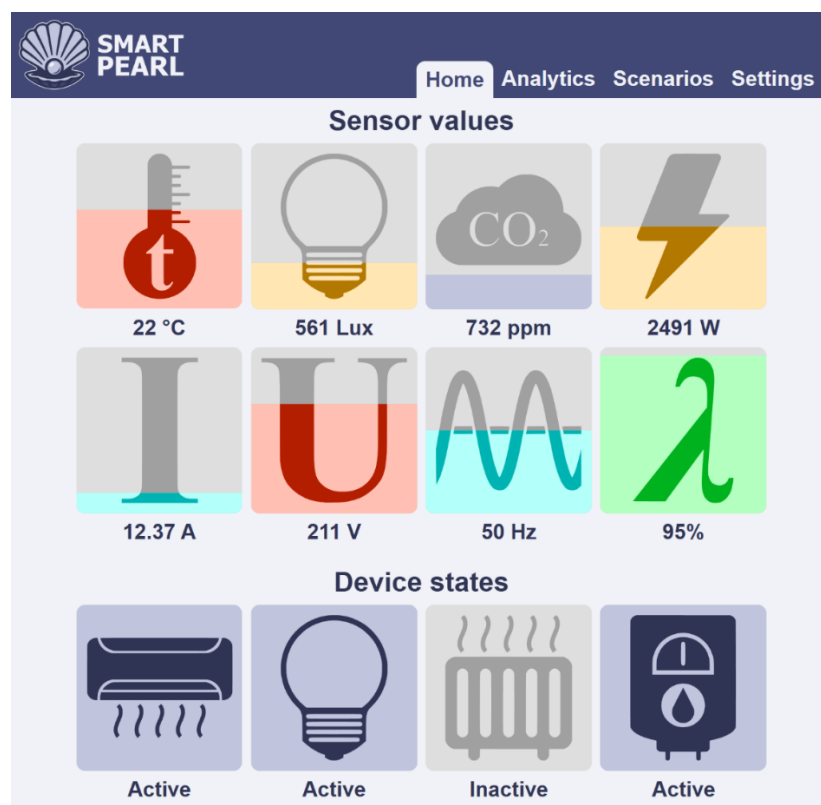


Рисунок 3.2 – Зовнішній вигляд вкладки «Home» під час роботи освітлювального пристрою, кондиціонування та бойлера

Для кращого сприйняття інформації вона подається у графічному вигляді. Під горизонтальною віссю наведені хронологічні мітки записів, а ліворуч від вертикальної осі – найближчі границі до мінімального та максимального значень показників на відповідному часовому проміжку.

До переліку інформації, яку може отримати користувач у графічному вигляді на вкладці «Analytics» входять:

- погодинна історія активності кожного з підключених пристроїв протягом останніх 24 годин;
- сумарна кількість годин активності підключених пристроїв щодобово протягом останніх 30 діб та щомісячно протягом останніх 12 місяців;
- середні значення показників якості внутрішнього середовища (температури, освітленості, концентрації вуглекислого газу) та напруги погодинно протягом останніх 24 годин, щодобово протягом останніх 30 діб та щомісячно протягом останніх 12 місяців;
- сумарна кількість витраченої енергії погодинно протягом останніх 24 годин, щодобово протягом останніх 30 діб та щомісячно протягом останніх 12 місяців;
- погодинна історія присутності людини протягом останніх 24 годин;
- сумарна кількість годин присутності людини щодобово протягом останніх 30 діб та щомісячно протягом останніх 12 місяців.

Для здійснення порівняльного аналізу на кожному з цих графіків додатково представлені дані за аналогічний попередній період. На рисунку 3.3 зображено побудований системою графік середнього щогодинного рівня освітленості за останні 24 години, а на рисунку 3.4 представлено графік активності освітлення за той самий період. Дані за аналогічний попередній період відображаються світлим кольором.

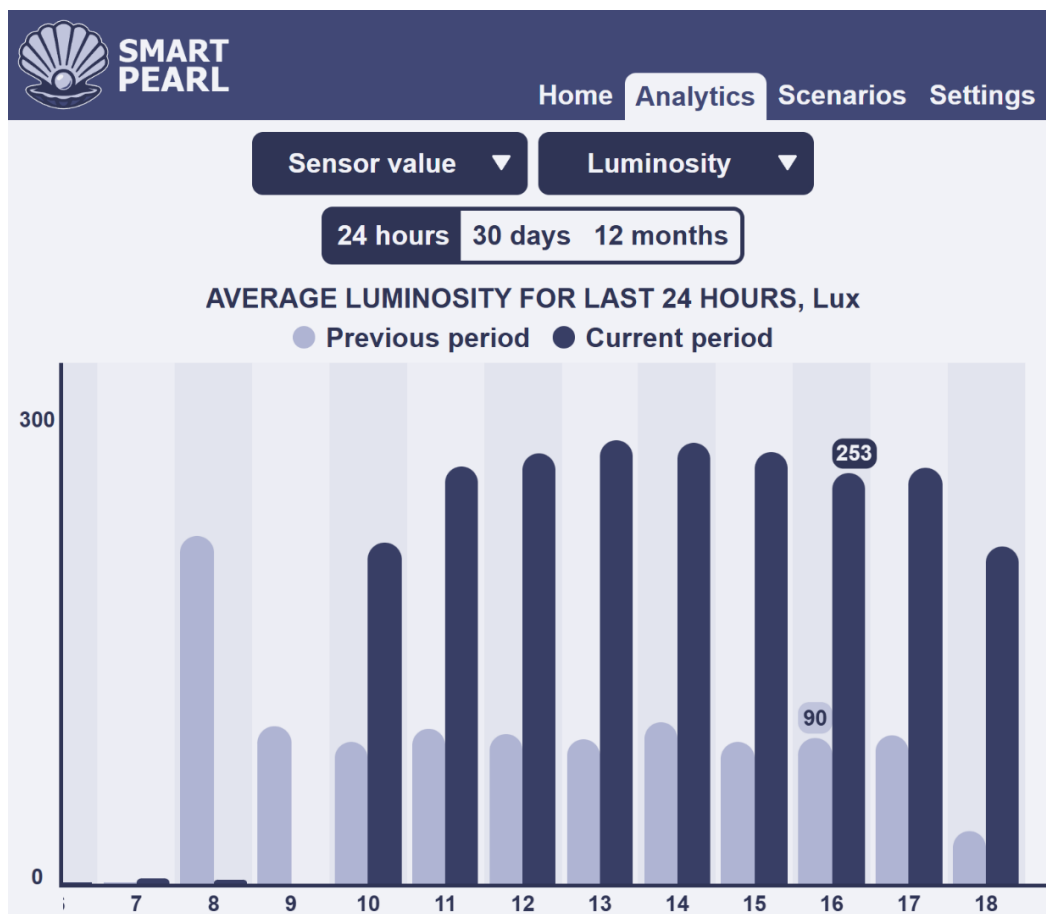


Рисунок 3.3 – Зовнішній вигляд вкладки «Analytics» із графіком середнього погодинного рівня освітленості за останні 24 години

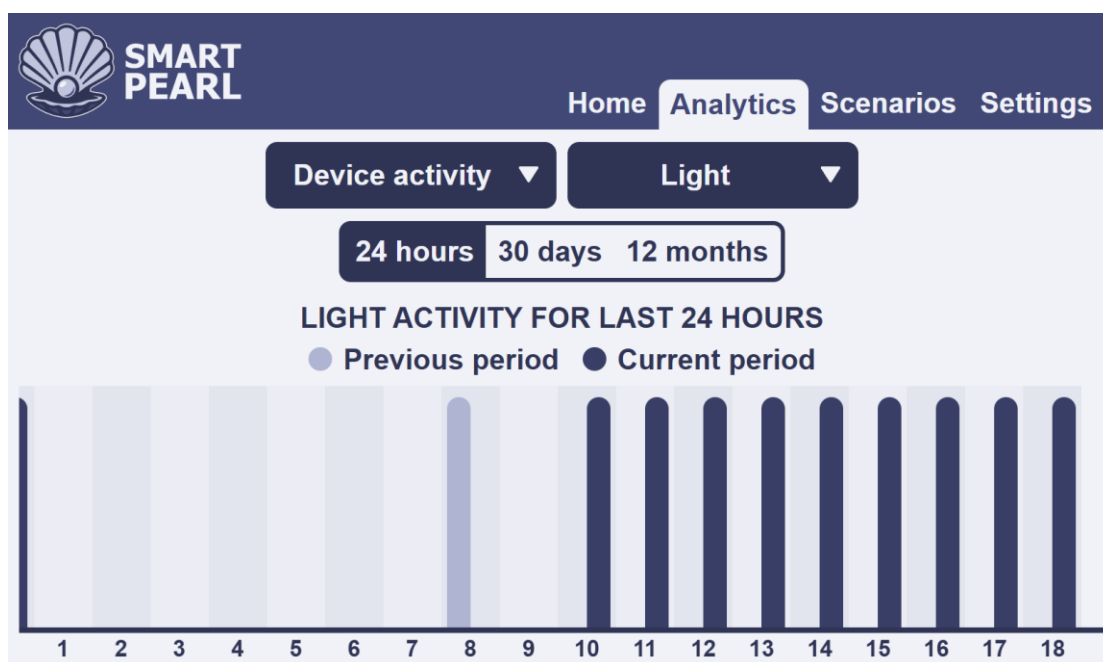


Рисунок 3.4 – Зовнішній вигляд вкладки «Analytics» із графіком активності освітлення за останні 24 години

Такий аналітичний функціонал надає користувачу широкі можливості щодо оптимізації енергоспоживання, а саме:

- виявляти пікові навантаження та час їх виникнення, щоб перерозподілити навантаження на інший час, наприклад, перенести прання на ніч;
- виявляти «аномалії» енергоспоживання, неефективні або старіючі пристрої, наприклад, опалювальний пристрій став споживати на 30% більше енергії;
- прогнозувати витрати енергії на основі звичок та сезону;
- оцінити, скільки електроенергії споживають «паразитні» навантаження (пристрої в режимі очікування);
- оцінити ефективність опалення та кондиціонування (наприклад, наскільки швидко зігрівається приміщення та скільки енергії це вимагає);
- побачити кореляцію між мікрокліматом та енергоспоживанням – допомагає зрозуміти, за яких умов система витрачає більше енергії;
- підлаштовувати роботу систем опалення, кондиціонування та освітлення під присутність або відсутність людини;
- виявляти «надлишкові» години роботи пристроїв під час відсутності людини;
- проводити аналіз поточного використання підключених пристроїв та обирати оптимальні сценарії для зниження енерговитрат (наприклад, знизити на 1°C порогове значення температури для спрацьовування опалювального пристрою, після чого проаналізувати графіки температури та витраченої енергії і зробити висновок щодо збереження або ж скасування такої зміни).

На вкладці «Scenarios» користувач має можливість переглядати та змінювати налаштування системи, які формують сценарії керування підключеними до неї пристроями. Під кожний з керованих пристроїв відведено окремий блок налаштувань: «Lighting» (рисунок 3.5) – для освітлення, «Air conditioning» (рисунок 3.6) – для кондиціонування, «Heater» (рисунок 3.7) – для опалювання та «Boiler» (рисунок 3.8) – для бойлера, а також є окремий блок «Manual control» (рисунок 3.9) для налаштувань, пов'язаних із ручним керуванням системою.

Lighting

Auto activation

☒ Enable

☐ While absent

0 **Luminosity threshold (lux): 300** 2000

☒ Special late time threshold

Late time start: 18:30

Late time end: 8:30

0 **Late time luminosity threshold (lux): 200** 2000

Inactive period

☒ Enable

Inactive period start: 00:00

Inactive period end: 06:00

Presense imitation

☒ Enable

0 **Absence period (h): 4** 20

Рисунок 3.5 – Зовнішній вигляд блоку вкладки «Scenarios» зі сценаріями освітлення

Air conditioning

Auto activation

☒ Enable

☐ While absent

400 **Carbon dioxide threshold (ppm): 800** 2000

-20 **Temperature threshold (°C): 26** 50

Рисунок 3.6 – Зовнішній вигляд блоку вкладки «Scenarios» зі сценаріями кондиціонування

Heater

Auto activation

☒ Enable

☒ While absent

-20 **Temperature threshold (°C): 20** 50

☒ Special absence threshold

-20 **Absence temperature threshold (°C): 10** 50

Рисунок 3.7 – Зовнішній вигляд блоку вкладки «Scenarios» зі сценаріями опалювального пристрою

Boiler

Active period

☒ Enable

Active period start: 22:30

Active period end: 23:30

Active days

☒ Monday

☒ Tuesday

☒ Wednesday

☒ Thursday

☐ Friday

☐ Saturday

☒ Sunday

Рисунок 3.8 – Зовнішній вигляд блоку вкладки «Scenarios» зі сценаріями бойлера

Manual control

☒ Manual control has higher priority

☒ Return to automatic control if human is absent

0 **Absence period (min): 0** 60

Рисунок 3.9 – Зовнішній вигляд блоку вкладки «Scenarios» з налаштуваннями ручного керування

У блоці «Manual control» можна зробити вищим пріоритет автоматичного керування або ж керування вручну, задати сценарій повернення у автоматичний режим при відсутності людини певний час та задати значення цього часового інтервалу. Для освітлення є можливість налаштувати автоматичну активацію при подоланні зазначеного користувачем порогу освітленості при відсутності або ж присутності людини, задати нижчий поріг для темнішого періоду дня, а також проміжок часу, протягом якого освітлення взагалі не має вмикатись незалежно від рівня освітленості або присутності людини. Також можна налаштувати сценарій «імітації присутності», який передбачає, що при відсутності людини заданий період часу система вмикатиме та вимикатиме з випадковими інтервалами освітлення, імітуючи присутність людини в приміщенні. У блоці для кондиціонування повітря присутні налаштування автоматичної активації в присутності або відсутності людини в залежності від подолання обраних користувачем порогів концентрації вуглекислого газу та температури. Блок, пов'язаний із опалювальним пристроєм, включає сценарій автоматичної активації при зниженні температури до зазначеного користувачем порогу та налаштування окремого порогу спрацьовування за умови відсутності людини. Сценарій використання бойлера передбачає задання часу та днів, коли він буде активним.

Вкладка «Settings», зовнішній вигляд якої зображено на рисунку 3.10, також містить налаштування, проте вони визначають не використання під'єднаних до системи пристроїв, а її загальні параметри та взаємодію із сенсорами або вимірювальними модулями, що підтримують програмне налаштування або калібрування. Як і попередня вкладка, ця поділена на окремі блоки: «General» – для загальних налаштувань системи, «Carbon dioxide sensor» – для датчика вуглекислого газу, «Human presence detector» – для датчика людської присутності.

До загальних налаштувань системи належать поточний час, IP-адреса, шлюз, підмережа, DNS-адреса, ім'я та пароль від мережі, а також інтервал, з яким система опитує модулі вимірювання, отримує стани керованих пристроїв, а також на підставі цих даних вмикає або вимикає пристрої; та інтервал, з яким система зберігає зчитані показники та стани підключених пристроїв до бази даних.

SMART PEARL

Home Statistics Scenarios **Settings**

General

Current time: 08.10.25 18:14
 IP address: 192.168.1.4
 Gateway: 192.168.1.2
 Subnet: 255.255.255.0
 Dynamic name: smartpearl
 SSID: SP_SSID
 Password: password93

Reading interval (s): 1 (slider from 1 to 60)
 Storing interval (s): 10 (slider from 10 to 600)

Carbon dioxide sensor

Confirm zero calibration

Span point (ppm): 1000 (slider from 1000 to 5000) **Confirm span calibration**

Detection range:
☒ 2000 ppm
☐ 5000 ppm
☒ Auto calibration

Human presence detector

Detection distance: 8 (slider from 1 to 8)
 Sensitivity: 80 (slider from 0 to 100)

Unoccupied state delay:
☒ 10 min
☐ 30 min
☐ 1 hour
☐ 2 hours

Reset settings

Рисунок 3.10 – Зовнішній вигляд вкладки «Settings»

Блок налаштувань датчика вуглекислого газу надає можливість програмного калібрування за нульовою точкою, тобто концентрацією в 400 ppm, та за обраною точкою, яку треба обрати в інтервалі між 1000 та 5000 ppm; а також переведення в режим автоматичного калібрування, яке передбачає, що датчик буде підлаштовуватись під поточний рівень концентрації вуглекислого газу для більш точного вимірювання. Також у даному блоці можна обрати верхню межу

вимірювання із значень у 2000 та 5000 ppm, що впливатиме на точність вимірювання.

У блоці взаємодії із налаштуваннями датчика присутності людини можна змінювати дистанцію розпізнавання у діапазоні від 1 до 8 ділянок по 0,75 м, налаштовувати чутливість датчику у відсотках та обирати величину затримки перед реєструванням людської відсутності.

Задля можливості зручного використання веб-інтерфейсу на будь-якому пристрої, під час його розробки був дотриманий принцип адаптивності. Це означає, що вміст сторінки змінює свій вигляд, підлаштовуючись під поточну ширину дисплею та обраний масштаб. Зовнішній вигляд вищенаведених вкладок на дисплеї мобільного телефону наведено на рисунках 3.11-3.12.

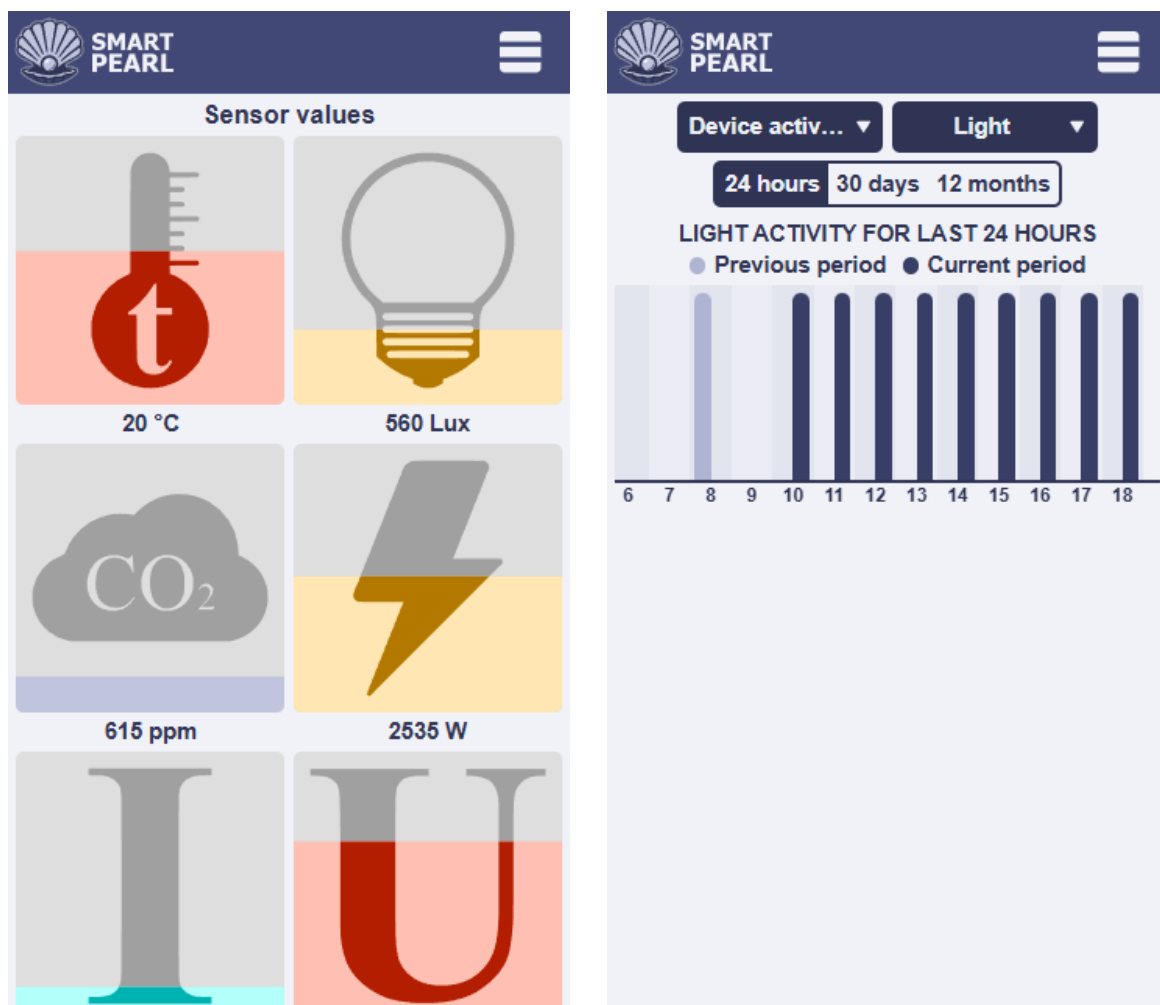


Рисунок 3.11 – Зовнішній вигляд вкладок «Home» та «Analytics» відповідно на дисплеї із роздільною здатністю 360×640 пікселів



Рисунок 3.12 – Зовнішній вигляд вкладок «Scenarios» та «Settings» відповідно на дисплеї із роздільною здатністю 360×640 пікселів

ВИСНОВКИ ДО РОЗДІЛУ 3

У даному розділі було детально розглянуто алгоритм роботи системи моніторингу та управління енергоспоживанням в приміщенні: порядок спрацьовування та призначення функцій, записаних у прошивку мікроконтролера, а саме команд, що виконуються на початку роботи системи, та тих, що викликаються у головному циклі з певним інтервалом; а також показано повний діапазон можливостей віддаленої взаємодії з системою за допомогою веб-інтерфейсу користувача.

На основі перевірки роботи прототипу системи можна зробити наступний висновок. Мета роботи була досягнута, а дана система відповідає заданим вимогам. Вона спроможна: відстежувати у реальному часі показники якості внутрішнього середовища (температуру, освітленість, концентрацію вуглекислого газу), параметри електромережі (активну потужність, силу струму, напругу, частоту та коефіцієнт потужності) та присутність людей; проводити калібрування датчиків у випадку їх заміни; збирати та зберігати дані у локальному сховищі; автономно керувати підключеними до системи пристроями на основі визначених користувачем сценаріїв; здійснювати статистичну обробку даних та представляти результати у зручному форматі для прийняття рішень щодо оптимізації енергоспоживання. Створений інтерфейс користувача є зручним, інтуїтивно зрозумілим та завдяки своїй адаптивності підходить для використання на дисплеї будь якого розміру. Веб-підхід в свою чергу гарантує, що інтерфейс може бути використаний на будь-якій платформі.

Таким чином, було створено ефективну систему моніторингу та управління енергоспоживанням в приміщенні з інтуїтивно зрозумілим кросплатформним інтерфейсом користувача. Система є зручною у використанні, має високу точність вимірювання, є гнучкою, адже дозволяє оновлення та заміну апаратних компонентів, а також доступна завдяки своїй низькій вартості впровадження.

ВИСНОВКИ

Для створення ефективної системи моніторингу та управління енергоспоживанням у приміщенні було проведено комплексне дослідження, яке включало послідовне виконання кількох послідовних етапів.

Спочатку була розглянута проблема енергоефективності будівель та визначено поняття «розумного енергоспоживання». Потім був здійснений аналіз існуючих пристроїв та систем інтернету речей, спрямованих на підвищення енергоефективності будівель. Це дозволило виявити коло проблем, що потребують вирішення. Прості розумні пристрої є доступними, але націлені лише на один напрямок енергоспоживання і не реалізують моніторинг мікроклімату та енерговитрат. Енергомонітори виконують лише функцію контролю енергоспоживання. Комплексні системи розумного будинку хоч і мають широкий функціонал, однак їх придбання, встановлення та експлуатація коштують занадто дорого. Крім того, частина їх функціоналу часто виявляється надлишковою, а складний інтерфейс вимагає від користувача технічної обізнаності.

Виявлені недоліки існуючих засобів інтернету речей у сфері розумного енергоспоживання показали необхідність створення системи моніторингу та управління енергоспоживанням в приміщенні, яка б була доступною широкому колу споживачів і, водночас, мала широкі функціональні можливості.

На наступному етапі були визначені основні вимоги до системи, а саме: масштабованість та гнучкість архітектури, здатність до інтеграції та сумісність, достатня потужність апаратного забезпечення, можливість автономної роботи, надійність, точність та оперативність моніторингу, безпека, зручність використання, економічність.

Проведений аналітичний огляд існуючих підходів до формування архітектури систем інтернету речей показав, що для системи моніторингу та управління енергоспоживанням у приміщенні оптимальним вибором є 4-рівнева архітектура, адже вона поєднує такі якості, як простоту розгортання й

використання, низьку вартість, достатній рівень безпеки та можливість подальшого розширення та розвитку.

Далі було розроблено архітектуру системи, її структурну та функціональну схеми. Проведений аналіз апаратних і програмних рішень дозволив обрати мікроконтролер, вимірювальні модулі, протокол передачі даних та систему управління базою даних. На основі цього було створено робочий прототип на макетній платі.

Задля створення програмного забезпечення (прошивки мікроконтролера та інтерфейсу користувача) були оглянуті можливі варіанти засобів розробки та мов програмування. Обґрунтованим вибором для написання серверної частини програмного забезпечення стали фреймворк PlatformIO та мова C++, а для клієнтської частини був обраний веб-підхід із чистим JavaScript. Створене програмне забезпечення було завантажено у мікроконтролер та відлагоджено.

Проведене тестування підтвердило працездатність системи та її спроможність виконувати всі необхідні функції: відстеження у реальному часі показників якості внутрішнього середовища (температури, освітленості, концентрації вуглекислого газу), параметрів електромережі (активної потужності, сили струму, напруги, частоти, коефіцієнта потужності) та присутності людей; калібрування датчиків у випадку їх заміни або додавання нових; збирання та зберігання даних у локальному сховищі; автономне керування підключеними до системи пристроями на основі визначених користувачем сценаріїв; статистична обробка даних та представлення результатів у зручному форматі. Створений інтерфейс користувача є зручним, інтуїтивно зрозумілим та завдяки своїй адаптивності підходить для використання на дисплеї будь якого розміру. Веб-підхід в свою чергу гарантує, що інтерфейс може бути використаний на будь-якій платформі.

Наукова новизна роботи полягає у новій структурній організації системи моніторингу та управління енергоспоживанням в приміщенні, яка перевершує існуючі рішення тим, що поєднує такі переваги 4-рівневої архітектури, як низька вартість, простота розгортання та використання, з розширеними можливостями, які

включають не лише моніторинг енергоспоживання, але й відслідковування параметрів якості внутрішнього середовища, зберігання великого обсягу статистичних даних без використання зовнішніх серверів, можливість заміни та калібрування датчиків, аналітичний функціонал та можливість налаштування сценаріїв використання електроприладів.

Практична цінність отриманих результатів роботи дозволяє рекомендувати створену систему для:

- використання у приміщеннях різного призначення (медичні та навчальні установи, готельний бізнес, офісні та житлові приміщення тощо) з метою забезпечення оптимального балансу між комфортом, витратами та енергоспоживанням;
- оцінки ефективності роботи енергетичного обладнання (виявлення неефективних пристроїв, пікових навантажень та аномалій у споживанні енергії, частки «паразитних» навантажень тощо);
- збору даних за великий проміжок часу, які можуть бути використані для прогнозування енергоспоживання з урахуванням звичок та сезонності;
- використання у якості складової системи «розумного» будинку;
- розробки модифікацій системи з урахуванням специфічних потреб різних галузей;
- використання у якості основи для подальших досліджень у галузі технологій розумного енергоспоживання.

Таким чином, у результаті виконаних досліджень було створено багатофункціональну IoT-систему моніторингу та управління енергоспоживанням в приміщенні, що є доступною широкому колу споживачів та відповідає всім висунутим до неї вимогам.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Бенч Н. «Зелені» будинки: як підвищити енергоефективність споруд в Україні. Що змінилося в ДБН і на кого в Європі слід рівнятися. *Mind*. 20 липня 2020. URL: <https://mind.ua/openmind/20213368-zeleni-budinki> (дата звернення 03.09.2025).
2. Белоусова К. В ЄС домовилися про нові правила з підвищення енергоефективності будівель. *Екополітика*. 08 грудня 2023. URL: <https://ecopolitic.com.ua/ua/news/v-ies-domovilisya-pro-novi-pravila-z-pidvishhennya-energoefektivnosti-budivel/> (дата звернення 03.09.2025).
3. ДБН В.2.6-31:2021. Теплова ізоляція та енергоефективність будівель. [Чинний від 2022-10-01]. Київ : Мінрегіон України, 2022. 27 с. URL: https://dreamdim.ua/wp-content/uploads/2022/08/DBN-V_2_6-31-2021.pdf (дата звернення 03.09.2025).
4. Жураковський, Б. Ю., Зенів І. О. Технології інтернету речей : навчальний посібник для здобувачів ступеня бакалавра за освітньою програмою «Інформаційне забезпечення робототехнічних систем». Київ : КПП ім. Ігоря Сікорського, 2021. 271 с. URL: https://ela.kpi.ua/bitstream/internet_rechey.pdf (дата звернення: 28.04.2025).
5. Кицкай Л. І. Енергоефективність в Україні: аналіз, проблеми та шляхи підвищення. *Інноваційна економіка. Всеукраїнський науково-виробничий журнал*, 2013. № 3 (41), С. 32-37. URL: http://www.irbis-nbuv.gov.ua/cgi-bin/inek_2013_3_8 (дата звернення 03.09.2025).
6. Колесник Є. Мікроклімат приміщень: нормування. Європейський підхід. *Міжнародний портал з енергозбереження*. URL: <https://patriot-nrg.com/uk/content-yeuropeyskyy-pidhid> (дата звернення 09.09.2025).
7. Немчинський С. Що таке СУБД і для чого вони потрібні. URL: <https://foxminded.ua/systema-upravlinnia-danykh/> (дата звернення: 13.04.2025).
8. Про енергетичну ефективність будівель : Закон України від 22 червня 2017 р. № 2118-VIII. URL: <https://zakon.rada.gov.ua/laws/show/2118-19#Text> (дата звернення 07.09.2025).
9. Про затвердження Методики визначення економічно доцільного рівня

енергетичної ефективності будівель : Наказ Міністерства регіонального розвитку, будівництва та житлово-комунального господарства України від 11 липня 2018 р. № 170. URL: <https://zakon.rada.gov.ua/laws/show/z0823-18#Text> (дата звернення 07.09.2025).

10. Про затвердження Методики визначення енергетичної ефективності будівель : Наказ Міністерства регіонального розвитку, будівництва та житлово-комунального господарства України від 11 липня 2018 р. № 169. URL: <https://zakon.rada.gov.ua/laws/show/z0822-18#Text> (дата звернення 07.09.2025).

11. Про затвердження мінімальних вимог до енергетичної ефективності будівель : Наказ Міністерства розвитку громад та територій України від 27 жовтня 2020 р. № 260. URL: <https://zakon.rada.gov.ua/laws/show/z1257-20#Text> (дата звернення 07.09.2025).

12. Що таке ZigBee в IoT? Повний путівник у 2024 році. *Dusun IoT*, Лютий 19, 2024. URL: <https://www.dusuniot.com/uk/blog/the-definitive-guide-for-zigbee-gateway-for-solution-vendors/> (дата звернення: 1.04.2025).

13. Що таке енергоефективність і як її впроваджувати. *Центр екологічних ініціатив «Екодія»*. 15 березня 2023. URL: <https://ecoaction.org.ua/shcho-take-ee.html> (дата звернення 03.09.2025).

14. Яцків В. В., Гавриляк Н. В., Стасюк І. І. Аналіз мережевих шлюзів інтернет речей. *Комп'ютерні інформаційні технології CIT'2019* : матеріали школи-семінару молодих вчених і студентів, 29 листопада 2019. Тернопіль : ТНЕУ, 2019. С. 30-31. URL: <https://api.dspace.wunu.edu.ua/api/core/bitstreams/dc1c8870-f5fb-44e8-a5ea-53a88364d700/content> (дата звернення: 5.05.2025).

15. Arduino IDE. Програмування Ардуїно. URL: <https://doc.arduino.ua/> (дата звернення: 22.04.2025).

16. Bluetooth Low Energy (Ble): A Comprehensive Overview. *CQR company*, April 19, 2023. URL: <https://cqr.company/wiki/protocols/bluetooth-low-energy-ble-a-comprehensive-overview/> (дата звернення: 1.04.2025).

17. Cloud, Fog and Edge Computing – What's the Difference? *Winsystems Inc. Company*, December 4, 2017. URL: <https://www.winsystems.com/cloud-fog-and-edge->

[computing-whats-the-difference/](#) (дата звернення: 7.04.2025).

18. Commercial Buildings Energy Consumption Survey. Consumption and Expenditures Highlights. U.S. *Energy Information Administration, Office of Energy Statistics, U.S. Department of Energy*. December 2022. 38 p. URL: <https://www.eia.gov/consumption/commercial/data/2018/pdf/CBECS%202018%20CE%20Release%202%20Flipbook.pdf> (дата звернення: 21.03.2025).

19. DBMS Tutorial – Database Management System. Sanchhaya Education Private Limited. URL: <https://www.geeksforgeeks.org/dbms/?ref=lbp> (дата звернення: 13.04.2025).

20. Desigo CC – Better. Of course. *Siemens*. URL: <https://www.siemens.com/global/en/products/buildings/desigo-building-automation/building-management/desigo-cc.html> (дата звернення 03.09.2025).

21. EcoStruxure Building Operation. *Schneider Electric*. URL: <https://www.se.com/us/en/work/products/product-launch/building-management-system/> (дата звернення 03.09.2025).

22. Ejaz W., Anpalagan A. Internet of things for smart cities. Technologies, Big Data and Security. Cham, Switzerland : Springer International Publishing, 2019. 66 p. URL: <https://doi.org/10.1007/978-3-319-95037-2> (дата звернення: 12.10.2025).

23. Ellis C., Watson J. The best smart lights 2025: light up your life with smart bulbs and lamps. *TechRadar: The Technology Experts*. May 13, 2025. URL: <https://www.techradar.com/news/best-smart-lighting> (дата звернення 03.09.2025).

24. Energy Efficiency as a Resource. *American Council for an Energy-Efficient Economy (ACEEE)*. URL: <https://www.aceee.org/topic/energy-efficiency-as-a-resource> (дата звернення: 21.03.2025).

25. Energy Efficiency Strategies and Upgrades. *American Council for an Energy-Efficient Economy (ACEEE)*. URL: <https://www.aceee.org/topic/energy-efficiency-strategies-and-upgrades> (дата звернення: 21.03.2025).

26. Energy Monitor EmonPi2: User Guide. *OpenEnergyMonitor*. URL: <https://docs.openenergymonitor.org/emonpi2/index.html> (дата звернення 03.09.2025).

27. ESP Development Boards. *ESPBoards – The Hub for ESP32 Boards*,

Troubleshooting, Tools & Tutorials URL: <https://www.espboards.dev/> (дата звернення: 12.02.2025).

28. ESP32. Electronic Components Datasheet Search. URL: <https://www.alldatasheet.com/view.jsp?Searchword=Esp32%20datasheet> (дата звернення: 23.04.2025).

29. ESP8266 Technical Reference. *Espressif Systems (Shanghai) Co., Ltd*, 2017. 117 p. URL: <https://pdf1.alldatasheet.com/datasheet-pdf/view-ESPRESSIF-ESP8266.html> (дата звернення: 29.04.2025).

30. ESP-IDF Programming Guide. *Espressif Systems (Shanghai) Co., Ltd*. URL: <https://docs.espressif.com/projects/esp-idf/stable> (дата звернення: 21.04.2025).

31. Evans D. The Internet of Things: How the Next Evolution of the Internet Is Changing Everything. *Cisco Internet Business Solutions Group*, 2011. 11 p. URL: https://www.cisco.com/docs/innov/IoT_IBSG_0411FINAL.pdf (дата звернення: 28.09.2025).

32. Flanagan D. JavaScript: The Definitive Guide: 6th-Edition. O'Reilly Media, Inc., 2011. 1098 p. URL: <https://pepa.holla.cz/wp-content/uploads/2016/08/JavaScript-The-Definitive-Guide-6th-Edition.pdf> (дата звернення: 15.04.2025).

33. Future Internet: The Internet of Things Architecture, Possible Applications and Key Challenges / Khan R., Khan S. U., Zaheer R., Khan S. 10th International Conference on Frontiers of Information Technology (FIT). *Institute of Electrical and Electronics Engineers Inc. Queen's University Belfast*, December 17-19, 2012. P. 257-260. URL: <https://pureadmin.qub.ac.uk/PID2566391.pdf> (дата звернення: 15.04.2025).

34. GL55 Series Photoresistor Manual. URL: <https://pdf1.alldatasheet.com-/datasheet-pdf/view/1131893/ETC2/GL5528.html> (дата звернення: 21.04.2025).

35. Hersent O., Boswarthick D., Elloumi O. The Internet of Things: Key Applications and Protocols. John Wiley & Sons Ltd, 2012. 353 p. URL: <https://nasrinword.wordpress.com/wp-content/uploads/2018/05/olivier-hersent-david-boswarthick-omar-elloumi-e28095the-internet-of-things-e28093-key-applications-and-protocols-ref-5.pdf> (дата звернення: 5.10.2025).

36. HLK-LD2410 Human presence sensing module manual. *Shenzhen Hilink*

Electronics Co., Ltd., 2022. 16 p. URL: <https://seengreat.com/upload/file/220629.pdf> (дата звернення: 21.04.2025).

37. How the Sense Energy Monitor Works. URL: <https://sense.com/consumer-blog/how-the-sense-home-energy-monitor-works/> (дата звернення 03.09.2025).

38. Howell J. An Introduction to IoT Architecture. *101 Blockchains*, April 3, 2023. URL: <https://101blockchains.com/iot-architecture/> (дата звернення: 8.04.2025).

39. Intelligent Efficiency in Commercial and Industrial Buildings. *American Council for an Energy-Efficient Economy (ACEEE)*, May 23, 2018. URL: <https://www.aceee.org/toolkit/2018/05/intelligent-efficiency-commercial-and-industrial-buildings> (дата звернення: 21.03.2025).

40. Intelligent Efficiency. *American Council for an Energy-Efficient Economy (ACEEE)*. URL: <https://www.aceee.org/topic/intelligent-efficiency> (дата звернення: 21.03.2025).

41. Intelligent Infrared CO₂ Module (Model: MH-Z19B). User's Manual. Version 1.0. Zhengzhou Winsen Electronics Technology Co., Ltd. 2016. 9 p. URL: https://www.winsen-sensor.com/d/files/infrared-gas-sensor/mh-z19b-co2-ver1_0.pdf (дата звернення 08.09.2025).

42. IoT Architecture – Detailed Explanation. *Scaler Academy*, June 3, 2022. URL: <https://www.interviewbit.com/blog/iot-architecture/> (дата звернення: 5.04.2025).

43. Kolban N. Kolban's Book on ESP32. September 2018. 1228 p. URL: https://roberthart56.github.io/SCFAB/SC_lab/Electronics/Microcontrollers/ESP32/kolban-ESP32.pdf (дата звернення: 28.03.2025).

44. Kubba S. Indoor Environmental Quality. *National Library of Medicine: National Center for Biotechnology Information*, 2017. URL: https://pmc.ncbi.nlm.nih.gov/articles/PMC7150156/?utm_source=chatgpt.com (дата звернення: 3.08.2025).

45. Kurniawan A. Internet of Things Projects with ESP32. Packt Publishing, 2018. 232 p. <https://else.fcim.utm.md/pluginfile.php/mod/content/1/esp32programming.pdf> (дата звернення: 22.04.2025).

46. Lea P. Internet of Things for Architects : Architecting IoT Solutions by

Implementing Sensors, Communication Infrastructure, Edge Computing, Analytics, and Security. Packt Publishing, The Limited, 2018. 524 p. URL: <https://www.oreilly.com/library/view/internet-of-things/9781788470599/> (дата звернення: 3.08.2025).

47. LM393 Datasheet. *Texas Instruments Inc.*, 2020. 55 p. URL: <https://pdf1.alldatasheet.com/datasheet-pdf/view/1260433/TI/LM393.html> (дата звернення: 22.04.2025).

48. McEwen A., Cassimally H. Designing the Internet of Things. John Wiley & Sons Ltd, 2014. 338 p. URL: https://madsg.com/Designing_the_Internet_of_Things.pdf (дата звернення: 7.05.2025).

49. MicroPython documentation. URL: <https://docs.micropython/en/latest> (дата звернення: 21.04.2025).

50. MKR 1000 WiFi. Documentation. URL: <https://docs.arduino.cc-/hardware/mkr-1000-wifi/> (дата звернення: 30.04.2025).

51. Nadel S., Ungar L. Halfway There: Energy Efficiency Can Cut Energy Use and Greenhouse Gas Emissions in Half by 2050. *American Council for an Energy-Efficient Economy (ACEEE)*, September 17, 2019. 70 p. URL: <https://www.aceee.org/sites/default/files/publications/researchreports/u1907.pdf> (дата звернення: 21.03.2025).

52. Nicklaus A. Fog Computing: Solving the limitations of Cloud and Edge Computing. Computer Science Blog. *Hochschule der Medien, Stuttgart*, February 27, 2023. URL: <https://blog.mi.hdm-stuttgart.de/fog-computing-solving-the-limitations-of-cloud-and-edge-computing/> (дата звернення: 7.04.2025).

53. Novello C. Mastering STM32. Leanpub, 2018. 852 p. URL: <https://www.embedic.com/uploads/files/20201008/Mastering%20STM32.pdf> (дата звернення: 7.03.2025).

54. Overview of the Internet of things. Recommendation ITU-T Y.2060. International Telecommunication Union. 2012. 22 p. URL: <https://www.itu.int/rec/T-REC-Y.2060-201206-I> (дата звернення: 20.10.2025).

55. Perry Ch. Smart buildings save energy and improve occupant comfort.

American Council for an Energy-Efficient Economy (ACEEE), December 20, 2017. URL: <https://www.aceee.org/blog/2017/12/smart-buildings-save-energy-and> (дата звернення: 21.03.2025).

56. Perry Ch. Smart Buildings: A Deeper Dive into Market Segments. *American Council for an Energy-Efficient Economy (ACEEE)*, December 20, 2017. 82 p. URL: <https://www.aceee.org/sites/default/files/publications/researchreports/a1703.pdf> (дата звернення: 21.10.2025).

57. Perry Ch., King J. Smart Buildings: Using Smart Technology to Save Energy in Existing Buildings. *American Council for an Energy-Efficient Economy (ACEEE)*, February 27, 2017. 55 p. URL: <https://www.aceee.org/research-report/a1701> (дата звернення: 30.09.2025).

58. Philips WiZ LED A60 Review – Enlightening Connected Smart Light. *Hitech Century*. URL: <https://hitechcentury.com/philips-wiz-led-a60-review/> (дата звернення: 03.09.2025).

59. Photoresistor Light Sensor Module for Arduino. Quick Overview MH Sensor Series LM393. URL: <https://robotchip./obzor-modulya-osveshchennosti-lm393/> (дата звернення: 29.04.2025).

60. PZEM-004T V3.0 User Manual. URL: <https://github.com/vortigont/pzem-edl/blob/main/docs/PZEM-004T-V3.0-Datasheet-User-Manual.pdf> (дата звернення: 21.03.2025).

61. Pico-series Microcontrollers. Documentation. *Raspberry Pi*. URL: <https://www.raspberrypi.com/documentation/microcontrollers/pico-series.html#picow-technical-specification> (дата звернення: 28.02.2025).

62. Shelly Pro 3EM. URL: [https://kb.shelly.cloud/knowledge-base/shelly-pro-3em#ShellyPro3EM\(SPEM-003CEBEU120\)-Deviceidentification](https://kb.shelly.cloud/knowledge-base/shelly-pro-3em#ShellyPro3EM(SPEM-003CEBEU120)-Deviceidentification) (дата звернення: 03.09.2025).

63. SQLite Documentation. URL: <https://www.sqlite.org/docs.html> (дата звернення: 10.04.2025).

64. State of IoT – Spring 2023. Market Report: IoT analytics / Brügge F., Hasan M., Kulezak M., Lueth K. L. May 2023. 137 p. URL: <https://iot->

[analytics.com/product/state-of-iot-spring-2023/](https://iot-analytics.com/product/state-of-iot-spring-2023/) (дата звернення: 23.11.2025).

65. State of IoT – Spring 2025. Market Report: IoT analytics / Kadian R., Lueth K. L., Taparia A., Wegner P. February 2025. 89 p. URL: <https://iot-analytics.com/product/state-of-iot-spring-2025/> (дата звернення: 4.10.2025).

66. State of IoT – Summer 2024. Market Report: IoT analytics / Krishnan A., Taparia A., Paraskevopoulos D., Pasqua E. August 2024. 171 p. URL: <https://iot-analytics.com/product/state-of-iot-summer-2024/> (дата звернення: 4.02.2025).

67. STM32L4 Discovery kit IoT node, low-power wireless, BLE, NFC, SubGHz, Wi-Fi. Product overview. URL: <https://www.st.com/en/evaluation-tools/b-l475e-iot01a.html> (дата звернення: 28.04.2025).

68. Struk V. The Pillar of Smart Devices: IoT Architecture Introduction. *Relevant Software LLC*, June 4, 2025. URL: <https://relevant.software/blog/iot-architecture/> (дата звернення: 5.10.2025).

69. Zigbee проти Bluetooth чи Wi-Fi, який із них найкращий для розумного дому? *Dusun IoT*, Лютий 20, 2025. URL: <https://www.dusuniot.com/uk/zigbee-vs-bluetooth-vs-wi-fi/> (дата звернення: 2.06.2025).