

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-  
КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ  
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ  
ТЕХНОЛОГІЙ  
КАФЕДРА ІНФОРМАЦІЙНИХ СИСТЕМ ТА ТЕХНОЛОГІЙ**

**КВАЛІФІКАЦІЙНА РОБОТА**

на тему:

«IoT-платформа для централізованого моніторингу та керування  
приладами розумного будинку»

на здобуття освітнього ступеня магістр  
зі спеціальності 126 Інформаційні системи та технології  
(код, найменування спеціальності)  
освітньо-професійної програми Інформаційні системи та технології  
(назва)

*Кваліфікаційна робота містить результати власних досліджень.  
Використання ідей, результатів і текстів інших авторів мають посилання  
на відповідне джерело*

Ростислав ВОВЧАНСЬКИЙ

\_\_\_\_\_  
(підпис)

\_\_\_\_\_  
Ім'я, ПРІЗВИЩЕ здобувача

Виконав: здобувач вищої освіти гр. ІСДМ- 62

Ростислав ВОВЧАНСЬКИЙ

\_\_\_\_\_  
Ім'я, ПРІЗВИЩЕ

Керівник: Д.т.н., професор Каміла СТОРЧАК

науковий ступінь,  
вчене звання

\_\_\_\_\_  
Ім'я, ПРІЗВИЩЕ

Рецензент: \_\_\_\_\_

науковий ступінь,  
вчене звання

\_\_\_\_\_  
Ім'я, ПРІЗВИЩЕ

**Київ 2025**

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ  
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ**

**Навчально-науковий інститут Інформаційних технологій**

Кафедра Інформаційних систем та технологій

Ступінь вищої освіти Магістр

Спеціальність 126 Інформаційні системи та технології

Освітньо-професійна програма Інформаційні системи та технології

**ЗАТВЕРДЖУЮ**

Завідувач кафедру ІПЗАС

\_\_\_\_\_ Каміла СТОРЧАК

« \_\_\_\_ » \_\_\_\_\_ 2025 р.

**ЗАВДАННЯ  
НА КВАЛІФІКАЦІЙНУ РОБОТУ  
Вовчанському Ростиславу Миколайовичу**

---

*(прізвище, ім'я, по батькові здобувача)*

1. Тема кваліфікаційної роботи: ІоТ-платформа для централізованого моніторингу та керування приладами розумного будинку

керівник кваліфікаційної роботи Каміла СТОРЧАК д.т.н, професор

*(Ім'я, ПРІЗВИЩЕ, науковий ступінь, вчене звання)*

затверджені наказом Державного університету інформаційно-комунікаційних технологій від «15» жовтня 2025 р. № 36

2. Строк подання кваліфікаційної роботи «26» грудня 2025 р.

3. Вихідні дані до кваліфікаційної роботи:

1. Технологія Bluetooth Low Energy;
2. Стандарт BLE;
3. Функції «розумного будинку».

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

1. Концепція розумного будинку та дослідження її реалізації.

2. Опис технології Bluetooth Low Energy.
3. Програмна реалізація інформаційної системи.
5. Ілюстративний матеріал: *презентація*
6. Дата видачі завдання: «15» жовтня 2025 р.

### КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз концепції «розумного будинку» та вимог до системи	15.10 - 28.10.2025	виконано
2	Дослідження технології Bluetooth Low Energy	29.10 - 15.11.2025	виконано
3	Проектування архітектури інформаційної системи	16.11 - 25.11.2025	виконано
4	Розробка та реалізація програмного забезпечення системи	26.11 - 08.12.2025	виконано
5	Розробка клієнт-серверної частини програмного комплексу	9.12 - 13.12.2025	виконано
6	Розробка презентаційних матеріалів	14.12 - 20.12.2025	виконано
7	Попередній захист роботи	24.12.25	виконано

Здобувач(ка) вищої освіти

\_\_\_\_\_ (підпис)

Ростислав ВОВЧАНСЬКИЙ

(Ім'я, ПРІЗВИЩЕ)

Керівник кваліфікаційної роботи

\_\_\_\_\_ (підпис)

Каміла СТОРЧАК

(Ім'я, ПРІЗВИЩЕ)





## ЗМІСТ

<b>ВСТУП.....</b>	<b>6</b>
<b>1 КОНЦЕПЦІЯ «РОЗУМНОГО БУДИНКУ» ТА АНАЛІЗ ПІДХОДІВ ДО ЇЇ ВПРОВАДЖЕННЯ .....</b>	<b>7</b>
Головні вимоги до реалізації системи розумний будинок .....	7
Функціональні вимоги до елементів розумного будинку .....	14
Аналіз можливостей, які надає система розумний будинок.....	16
<b>2 ОПИС ТЕХНОЛОГІЇ BLUETOOTH LOW ENERGY .....</b>	<b>21</b>
Особливості технології BlueTooth Low Energy .....	21
Режими роботи та структура даних .....	25
Частотний розподіл.....	36
Питання безпеки у протоколі BLE .....	38
<b>3 ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ .....</b>	<b>44</b>
Вибір технології та інструментарію.....	44
Розробка програмного забезпечення системи.....	47
Програмна реалізація алгоритму на основі нечіткої логіки .....	51
Розробка програмного комплексу клієнт–серверної частини .....	54
Сценарій роботи BLE сервера .....	58
Реалізація бази даних .....	59
<b>ВИСНОВКИ .....</b>	<b>63</b>
<b>ПЕРЕЛІК ПОСИЛАНЬ .....</b>	<b>64</b>

## ВСТУП

У сучасному розумінні, поняття розумний будинок охоплює систему передових технологічних пристроїв, спрямованих на забезпечення комфорту та ефективності в % щорічно, а витрати на утримання системи лишаються мінімальними. Перевагою житлових або робочих приміщеннях. Від початкового концепту, такі системи були розроблені з метою забезпечення безпеки та раціонального використання ресурсів в приміщенні. Енергоефективний будинок – це об'єкт, де енергетичні процеси оптимізовані для максимально ефективного використання ресурсів. Сучасні системи розумного будинку допомагають у раціональному управлінні фінансами. Незважаючи на високу вартість обладнання, установки та налаштування, вони демонструють значні економічні переваги: експлуатаційні витрати можуть зменшуватися на 15–40 таких систем є їх низьке споживання енергії, особливо у порівнянні з традиційними пристроями. Наприклад, у період очікування система майже не витрачає електроенергію. Навіть така проста функція, як автоматичне вимкнення світла при відсутності людей в приміщенні, може суттєво впливати на зменшення енергоспоживання. Освітлення, за статистикою, є одним з основних джерел споживання електроенергії в будинку, тож зменшення споживання енергії у системах освітлення може призвести до значної економії. Одним з факторів популярності розумних будинків є поява систем, побудованих на бездротових технологіях. Якщо раніше переважали провідні платформи, такі як KNX, Smart-bus та X10, то в останні роки системи на базі бездротових технологій, таких як Bluetooth, WiFi, ZigBee та Z-wave, набувають все більшого застосування. Метою магістерської роботи розробка є розробка інформаційної системи на основі технології Bluetooth Low Energy для оптимізації електроенергії в будинку.

# 1 КОНЦЕПЦІЯ «РОЗУМНОГО БУДИНКУ» ТА АНАЛІЗ ПІДХОДІВ ДО ЇЇ ВПРОВАДЖЕННЯ

## Головні вимоги до реалізації системи розумний будинок

Житло, що автоматично враховує бажання мешканців, забезпечує їхню безпеку і ефективно використовує енергоресурси, – це основна ідея розумного будинку, яка, хоча колись здавалася фантастичною, тепер вже є реальністю для багатьох людей у всьому світі. Розумне освітлення, автоматизоване паркування, керування системою за допомогою SMS та інші функції вже входять у повсякденне життя, дозволяючи відчутти практичну користь від використання передових технологій. Домашня автоматизація, або розумний будинок, – це система домашніх пристроїв, які можуть виконувати дії та розв'язувати різні завдання без участі людини. Домашня автоматизація розглядається як важлива частина Інтернету речей (IoT), і включає в себе доступні через Інтернет домашні пристрої, які можуть спілкуватися між собою та з іншими пристроями через мережу Інтернет.

Хоча концепція розумного будинку з'явилася ще у середині минулого століття, висока вартість реалізації подібних проектів тримала їх від широкого поширення. Однак з розвитком електроніки ситуація змінилася, і в даний час такі системи вже не вважаються нічим надзвичайним.

Перелік найбільш поширених систем розумного будинку:

- системи багатоканального озвучення це єдина система управління, яка дозволяє керувати аудіо– та відеообладнанням з будь-якої точки будинку, забезпечуючи зручне управління через сенсорний дисплей або смартфон;

- система відеомоніторингу яка дозволяє переглядати в реальному часі або в записі зображення з відеокамер на персональних пристроях, які можуть бути переносними або вбудованими в стіну та телевізори;
- система інтеркому, домофону яка сповіщає про прихід гостя та дозволяє відкривати двері через мобільний додаток або смс-повідомлення;
- системи безпеки яка у разі вторгнення в будинок система автоматично повідомляє про це власників та відповідні служби, а також може активувати сигналізацію та освітлення для затримання злочинців;
- система оповіщення про екстрені ситуації яка відслідковує витoki газу, пожежні та інші небезпеки, і автоматично викликає відповідні служби;
- система клімат-контролю яка за допомогою датчиків система визначає температуру та вологість повітря в будинку та регулює кондиціонування, забезпечуючи комфортні умови для проживання;
- система освітлення яка дозволяє змінювати освітлення в будинку залежно від потреб, економлячи електроенергію та створюючи затишну атмосферу.

Ці опції дозволяють створити розумний будинок, який відповідає потребам та уподобанням його мешканців. Єдиного стандарту для концепції розумний будинок немає, так само як і установленого списку вимог. Тому кожен може побудувати свій розумний будинок саме так, як йому заманеться. Технічною основою розумних будинків є автоматизована система управління будівлею, призначена для автоматизації процесів та операцій, які реалізуються в сучасних будівлях. У літературі часто зустрічається термін «Автоматична система управління будівлею» для опису систем автоматизації інженерних систем будівлі, таких як вентиляція, опалення, кондиціонування повітря, водопостачання, каналізація, електропостачання та освітлення. Великі і складні будівлі можуть мати десятки таких інженерних систем [1].

На сьогоднішній день багато відомих фірм пропонують готові рішення для реалізації концепції розумний будинок. Основні пристрої Synco living включають центральний квартирний модуль QAX910, який збирає дані з датчиків температури в кімнатах і керує опаленням (максимум 12 приміщень з 16 точками даних кожне). Він дозволяє задавати установки, забезпечує управління ГВС та взаємодію пристроїв системи (середа передачі TP1 і бездротова комунікація через KNX RF), а також виконує роль призначеного для користувача інтерфейсу (графічний дисплей з підсвічуванням). На модуль виводяться свідчення про споживання енергоносіїв, отримані від лічильників. Крім того, на лицьовій частині QAX910 розташовані 4 пари клавiш, які користувач може налаштувати самостійно для виконання різних додаткових функцій (управління освітленням, сценарії, перемикання різних режимів роботи). Можливо управління з розкладом з трьома режимами роботи: економія, передкомфорт і комфорт. Контролери контурів опалення RR V912 і RVV918. Вони управляють приводами (до 2 і 8 контурів відповідно), отримуючи команди від центрального модуля по KNX RF:

- безпроводний модуль QAW910 який призначений для вимірювання температури в приміщенні, завдання уставки і вибору режиму роботи. Не потребує зовнішньої напруги живлення і працює від батареї;
- кімнатний датчик QAA910 який вимірює і передає значення температури в приміщенні на центральний кімнатний модуль; ● інтерфейс збору даних WRI982.

Є елементом системи контролю споживання енергоносіїв та призначений для підключення різних лічильників (до 2 з імпульсними виходами і до 3 з комунікацією по стандартному протоколу M-bus). Інформацію про споживання передає на центральний квартирний модуль по радіоканалу. Завдяки циклічній відправці інформації він дозволяє користувачеві контролювати витрати на

тепло, холод, воду, газ і електрику. Під час підключення до вебсерверної системи OZW772 інформація про споживання ресурсів може переглядатися дистанційно за допомогою стандартного веббраузера. Концепція «розумного будинку» вже тривалий час перебуває на етапі активного розвитку: значна кількість виробників працює в цьому напрямі, однак універсального та загальноприйнятого стандарту досі не сформовано. Одним із перших кроків до інтеграції різномірних пристроїв стала діяльність компанії Nest. На початку 2014 року корпорація Google інвестувала близько 3,2 млрд доларів США у придбання компанії Nest, яка спеціалізується на розробці інтелектуальних пристроїв для житлових приміщень. [2].

У межах програми Works with Nest було оголошено про забезпечення сумісності термостатів і датчиків диму з пристроями інших виробників. Серед партнерів програми вже присутні August Smart Lock, Automatic Driving Assistant, Insteon, LG Appliances, Lutron, Ooma Telo, Philips Hue, Unikey Kevo Smart Lock, Withings Aura Sleep System, а також низка інших компаній, кількість яких поступово зростає.

Одним із найпростіших, проте водночас наочних прикладів такої інтеграції є взаємодія датчика диму Nest із розумними лампами Philips Hue. У разі спрацювання датчика освітлення починає блимати або змінює колір, що дозволяє користувачеві миттєво зреагувати на небезпеку, навіть перебуваючи в іншій частині будинку. Побутова техніка також може функціонувати в межах єдиної екосистеми Nest: аналізуючи показники термостата, розумні вентилятори Big Ass Fans автоматично підтримують стабільний температурний режим без втручання користувача. Холодильники LG та пральні машини Whirlpool переходять у режим енергозбереження, а розумний замок August Smart Lock автоматично зачиняє двері після виходу мешканців з оселі. Крім того, користувач отримує можливість віддаленого керування системою

«розумного будинку» за допомогою додатків Lutron та Insteon, які підтримують інтеграцію з Nest.

За допомогою сервісу Smart Home користувач може управляти всією технікою в будинку, використовуючи лише один додаток, інтегровану платформу і сервер, які об'єднують в домашню мережу всі пристрої: від холодильників і пральних машин до телевізорів, камер, смартфонів і дрібних пристроїв як, наприклад, Samsung GALAXY Gear.

Три основні функції Samsung Smart Home: управління пристроями (Device Control), спостереження за будинком (Home View) і користувача smart-сервіс (Smart Customer Service) дадуть можливість підключатися до пристроїв в будь-який час і в будь-якому місці. Сервіс «Samsung Smart Home» зображений на рис. 1.1.

Функція керування пристроями надає можливість змінювати параметри роботи побутової техніки незалежно від місця перебування користувача — як у межах житла, так і за його межами. За допомогою застосунку Smart Home на екрані мобільного пристрою можна дистанційно вмикати або вимикати кондиціонер чи освітлення, здійснюючи керування через єдину мережу, до якої під'єднано всю домашню техніку.



Рисунок 1.1 – Сервіс «Smart Home»

(<https://www.datasciencecentral.com/smart-home-as-a-service-how-smart-homes-can-deliver-1/>)

Також у системі передбачено можливість голосового керування. Користувачеві достатньо вимовити відповідну команду на мобільному пристрої-контролері з установленим застосунком. Наприклад, фраза «виходжу з дому», сказана в мікрофон Samsung Galaxy Gear, автоматично вимикає освітлення в оселі, а команда «на добраніч», подана через пульт дистанційного керування (хоч це може здаватися незвичним), призводить до автоматичного вимкнення телевізора [3].

Функція спостереження забезпечує виведення на екран смартфона актуального зображення з камер, інтегрованих у побутову техніку. Окрім цього, smart-сервіс надсилає нагадування про необхідність перевірки стану обладнання та сприяє своєчасному обслуговуванню техніки.

У процесі створення сервісу Smart Home розробники Samsung сфокусувалися, в першу чергу, на smart-телевізорах, смартфонах і побутовій техніки. Надалі число пристроїв (як «рідних», так і випущених іншими

виробниками), які будуть підтримувати роботу з сервісом значно збільшиться. Для забезпечення можливості підключення техніки, випущеної сторонніми компаніями, фахівці Samsung розробили Smart Home software protocol (SHP).

В майбутньому сервіс Smart Home дозволить стежити за витратою енергії, безпекою, здоров'ям користувачів і екологічністю – всі ці можливості будуть доступні у міру появи нових додатків, розроблених спільно з компаніями–партнерами для цього нового сегмента ринку з високим потенціалом до зростання та розвитку.

Apple HomeKit – це програмна платформа, яка об'єднує безліч пристроїв в одну бездротову мережу. Платформа Apple HomeKit зображена на рис. 1.2.

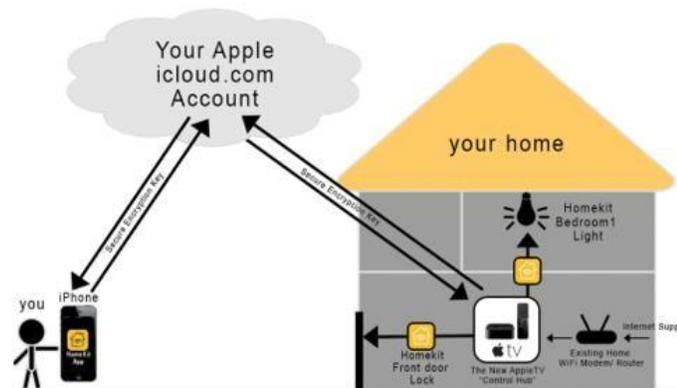


Рисунок 1.2 – Платформа Apple HomeKit (<https://solensy.com.ua/4-kroky/>)

Основною метою є створення системи «розумного будинку», що об'єднує велику кількість датчиків, інтелектуальних освітлювальних приладів і розеток, центральним елементом якої в ідеальному випадку виступає Apple TV четвертого покоління. Керування всією інфраструктурою здійснюється за допомогою пристроїв iPhone або iPad через спеціальний застосунок «Дім».

Без використання технології HomeKit користувач не має можливості контролювати освітлення з єдиного інтерфейсу та налаштовувати спільну

роботу пристроїв за розкладом. Наприклад, без цієї платформи неможливо автоматично вимикати світло й одночасно зачиняти штори щодня о визначеній годині — такі дії доводиться задавати вручну в окремих програмах.

Усі пристрої, що підтримують HomeKit, автоматично інтегруються з голосовим помічником Siri, і для їх подальшого використання достатньо одноразового налаштування через відповідний застосунок. У такій архітектурі Siri виконує роль універсального інтерфейсу для доступу до різних сервісів, тоді як окремі додатки залишаються необхідними для повноцінного керування розумними пристроями. Крім того, платформа HomeKit забезпечує шифрування даних, що передаються між пристроями та гаджетами Apple, завдяки чому підвищується рівень захисту інформації та унеможливорюється несанкціонований доступ до системи автоматизації [4].

### **Функціональні вимоги до елементів розумного будинку**

Для роботи програмної системи розумний будинок необхідно враховувати ряд функціональних вимог до його складових елементів, зокрема датчиків та контролера. Для цілей автоматизації та контролю різних аспектів життя в будинку використовуються різноманітні датчики та пристрої, які забезпечують збір інформації та можливість керування різними системами. До них відносяться датчики наявності та інтерфейсу. Датчики, які реагують на рух або присутність осіб у приміщенні, а також наявність інтерфейсу для зв'язку з центральним контролером через радіо– чи комп'ютерну мережі.

Контролер розумного будинку є ключовим елементом системи, який забезпечує керування всіма споживачами, приладами та системами. Він управляється датчиками температури, повітря та світла для контролю систем освітлення, обігріву та кондиціонування. Крім автономного режиму, з

контролером можна зв'язатися через спеціальний інтерфейс (комп'ютерну мережу, мобільний оператор або радіомережу), і керувати вручну приладами.

В залежності від архітектури системи управління використовуються різні типи контролерів:

- централізовані – єдині високопродуктивні центральні контролери;
- децентралізовані – декілька більш простих регіональних контролерів, кожен з яких відповідає за певну зону або групу пристроїв.

Також в залежності від комплектації може бути вбудований GSM модуль для віддаленого управління через мобільний телефон, або Wi-Fi передавач для управління системою з будь-якої точки будинку. Для зручності користування може бути встановлений графічний сенсорний або кнопковий інтерфейс, наприклад, РК-екран.

Регіональний контролер та дискретний модуль введення–виведення відіграють ключову роль у системі управління розумний будинок. Регіональний контролер є низькопотужним електронним блоком управління, який реалізує технологію розумний будинок. Він може налаштовуватися на виконання різних завдань за розкладом або відповідно до сигналів від датчиків. Дискретний модуль введення–виведення відповідає за управління елементами системи згідно отриманих сигналів. Ці компоненти взаємодіють між собою, забезпечуючи ефективно та зручне управління системою розумний будинок, та дозволяють забезпечити комфорт та безпеку в будинку.

Модулі розширення для системи розумний будинок розширюють функціональні можливості контролера. Ці пристрої можуть бути підключені до контролера через різні інтерфейси, такі як Ethernet, USB або спеціальні роз'єми. При виборі модулів важливо враховувати їх сумісність з контролером, особливо якщо вони від різних виробників. Модулі розширення можуть розширювати

функціональність системи, дозволяючи, наприклад, керувати звуковими засобами чи підключати різні типи пристроїв до системи розумний будинок.

Також варто відзначити розширення функціоналу за рахунок модулів–перехідників, які дозволяють досягти сумісності різних стандартів пристроїв. Спеціальні модулі управління призначені для контролю різних специфічних пристроїв, таких як двигуни, механізми відкривання/закривання дверей, клапани вентиляції тощо, а також для елементів комутації електричного кола.

Наприклад, контролер розумного будинку зазвичай не має можливості керувати силовими захисними автоматами (пробками, автоматичними вимикачами). Проте, при підключенні модуля управління автоматами і керованих автоматами, функція їх включення/виключення за допомогою контролера стає доступною. Для підключення до системи розумного будинку регуляторів освітлення також можуть бути необхідні спеціальні модулі розширення.

### **Аналіз можливостей, які надає система розумний будинок**

У разі реалізації ретельно прорахованого проекту, система розумний будинок дозволить не тільки істотно заощадити на оплаті комунальних послуг, але і подарує власнику максимальний затишок, надійний захист від небажаного проникнення, пожеж, протікання газу і води.

Система розумний будинок забезпечує злагоджену роботу всіх приладів і запобігає конфліктам і неузгодженості в роботі побутового обладнання та комплектуючих. Автоматизація домашніх процесів апріорі виключає ситуації нераціонального споживання ресурсів і електрики. Так, наприклад, система здатна самостійно відключити роботи кондиціонерів при зниженні температури і включення опалювальних приладів.

Практично всі представлені на сучасному ринку системи мають режим енергозбереження. У момент відсутності мешканців у приміщенні система вимикає всі прилади, які були запрограмовані на відключення. розумний будинок самостійно відстежує наближення господаря до будинку за допомогою GPS і включає необхідні для життєзабезпечення пристрої. Саме такий підхід і дозволяє добитися значної економії. Система розумний будинок зображена на рис. 1.3.



Рисунок 1.3 – Система розумний будинок

<https://naurok.com.ua/prezentaciya-do-uroku-yads-na-temu-rozumniy-dim-maybutnogo-336918.html>

Отже, розумний будинок здатний сьогодні:

- значно скоротити витрату електроенергії;
- контролювати використання води та газу;
- самостійно регулювати систему теплопостачання.

Автоматизоване опалення передбачає контроль за температурним режимом приміщень і регулювання графіка роботи нагрівачів, а також інших побутових приладів, які споживають енергоресурси. При цьому не варто хвилюватися про те, що на вулиці помітно потепліло, а в будинку щосили працюють обігрівачі. Система самостійно аналізує показники температури за вікнами, після чого віддає команду для корекції температурного режиму в будинку. Згідно зі статистикою, повна автоматизація вищевказаних процесів дозволяє заощадити близько 5 кВт · год в день. При перерахунку на вітчизняну валюту отримуємо пристойну суму – 1500–1800 грн на рік. Це абсолютно реальна цифра для двокімнатної квартири площею 50–60 квадратних метрів. Для величезного будинку або просторої квартири, природно, економія буде набагато істотніше [5].

Підтримуючий режим опалення – один з найефективніших способів пережити опалювальний сезон без дірки в кишені. Адже лєвова частка енергоресурсів при обігріві житла йде на періодичне підвищення температури до прийнятної. Чим просторіше квартира і більше будинок, тим, відповідно, вище енерговитрати. Крім того, багато що залежить і від товщини стін, склопакетів і динаміки зміни температури на вулиці.

При підтримуючому режимі опалення датчик температури сканує свідчення кожні 10–15 хвилин. Після досягнення встановленої позначки, опалювальна система вимикається автоматично. При зниженні температури в приміщенні більш ніж на 5 градусів, система моментально активується і приступає до обігріву по вже описаному сценарію.

Протягом календарного року за допомогою автоматизації тільки одного освітлення в типовій одне або двох кімнатній квартирі (50–60 квадратних метрів) можна заощадити до 1000 грн.

Для освітлення великого приватного будинку загальною площею в 150–180 метрів квадратних буде потрібно приблизно 100 світлодіодних ламп

потужністю 10 Вт (загальна потужність 1 кВт). При грамотному відключенні всіх ламп і раціональному розподілі ресурсів можна скоротити час роботи освітлювальних приладів до 4–5 годин в день.

Можливість регулювати яскравість освітлення в усьому будинку або квартирі – відмінний спосіб піти від зайвого енергоспоживання. Система розумний будинок буде включати тільки необхідні на даний момент світильники і лампи, попутно регулюючи їх потужність.

Для того, щоб освітлення працювало тільки в експлуатованих в даний момент зонах, в систему розумного будинку необхідно включити датчики руху. На щастя, такі пристрої коштують недорого. Якщо поставити датчики рівня освітленості, то система буде реагувати на зміни інтенсивності світла за вікном. Крім того, під час налаштування режиму «сон», інтелектуальна система подбає про відключення живлення всіх використовуваних електроприладів. Таким чином, всі підключені електроприлади контролюються розумним будинком, що дозволяє забути про працюючі без потреби телевізори або електроконвектори.

Для перемикання світлових схем як у всьому будинку, так і в межах одного приміщення, поряд з настінними пультами використовуються бездротові сенсорні панелі. Сценарії освітлення закладаються в пам'ять пристрою, після чого викликаються легким дотиком до сенсорних «кнопок». На практиці це дає багато, наприклад, одним дотиком можна освітлити шлях зі спальні у ванну, що не задіявши при цьому верхнього освітлення, а включивши тільки потрібну ділянку «світлової доріжки» з неяскравим світлом, який не буде сліпити очі після сну.

В системах освітлення розумного будинку знайшли широке застосування фотоелементи і датчики руху. Ці засоби відповідно дозволяють автоматично регулювати інтенсивність освітлення в залежності від зовнішньої освітленості і вмикати і вимикати світло в тих приміщеннях, де немає активності.

Також економити на електроенергії дозволяють фотореле для вуличного освітлення. Завдяки їм освітлювальні прилади вмикаються і вимикаються автоматично в залежності від сонячного світла.

Інша популярна функція розумного будинку – інтеграція всієї побутової техніки в єдину систему. У сучасному будинку можна без праці нарахувати зо два десятки різних гаджетів: мікрохвильовка, посудомийна і пральна машини, домашній кінотеатр, кондиціонер, душова кабіна ... У кожного з цих приладів – своя інструкція, свій пульт управління, свої особливості роботи.

Розумний будинок об'єднує всі системи, що існують в будинку, і забезпечує їх злагоджену і ефективну роботу, керувати якою можна за допомогою єдиного пульта, з домашнього комп'ютера або мобільного телефону. Наприклад, при включенні кондиціонера закриє вікна і вимкне теплі підлоги, контролюючи при цьому температуру і вологість повітря. Поки господарів немає вдома, температура повітря знижується, припиняється робота радіаторів, автоматично вимикаються світло і побутова техніка. А за півгодини до вашого повернення додому всі системи включаються і створюють для вас теплий, затишний, домашній клімат. Постійно аналізуючи обстановку, розумний будинок розпізнає конкретні ситуації і процеси, що відбуваються в будівлі, і реагує на них таким чином, щоб зберегти необхідний людині рівень комфорту.

Головна перевага розумного будинку полягає в тому, що він контролює всі системи в комплексі і не допускає неефективної, неекономної і зайвої роботи побутової техніки [6].

## 2 ОПИС ТЕХНОЛОГІЇ BLUETOOTH LOW ENERGY

### Особливості технології Bluetooth Low Energy

Передача та прийом даних потребують енергії, тому підвищують швидкість передачі даних, щоб в одиницю часу можна було передати більше інформації. У BLE швидкість передачі інформації становить 1 Mbit/c. Проте не лише швидкість передачі даних важлива. Головною особливістю BLE є можливість переходу пристроїв зв'язку в синхронний режим роботи. Іншими словами, пристрої перебувають у сплячому стані 99% часу, а потім на дуже короткий час прокидаються, обмінюються інформацією і знову засинають. Проте перед тим, як увійти в цей режим, необхідно пройти процедуру синхронізації, для чого використовується режим «advertising». Перш ніж розглядати особливості самої технології, варто згадати технічну частину, а саме маячки Bluetooth Low Energy. Маячки Bluetooth Low Energy зображені на рис. 2.1 [9].



Рисунок 2.1 – Маячки Bluetooth Low Energy

(<https://idcard.com.ua/bikon-hid-beeks-ble-beacon-mini-mini-tamper/>)

Термін «маячки» відноситься до мініатюрних радіопередавачів на основі технології Bluetooth Low Energy, які періодично відправляють дані в ефір. Маячки у бездротовій технології – це тип мініатюрних батарейних пристроїв для передачі інформації невеликого обсягу. Інформація може стосуватися різних параметрів навколишнього середовища (температура, тиск, вологість і т.і.) або стану об'єкта (місце розташування, прискорення, вібрація), а також повідомлень про ціни та товари в магазині. Передані дані, як правило, є статичними, але можуть також бути динамічними й змінюватися з часом. Маячки, що використовують Bluetooth Low Energy, зазвичай призначені для роботи протягом багатьох років на одній батареї розміром з монету. Логічно BLE складається з трьох основних рівнів:

- рівня додатків, призначеного для програм користувача, що використовують можливості BLE для реалізації конкретного призначення;
- рівня хоста, верхнього рівня стека протоколів BLE;
- рівня контролера, нижніх рівнів стека протоколів BLE.

Додатково до специфікації надається стандартний протокол обміну даними між хостом і контролером – Host Controller Interface (HCI), щоб забезпечити взаємодію між контролерами і хостами різних виробників [7].

Ці рівні можуть бути реалізовані в одній інтегральній мікросхемі або можуть бути рознесені в різні мікросхеми, які взаємодіють за допомогою будь-якого послідовного протоколу (UART, USB, SPI або інші). Тому всі пристрої можуть бути одного з трьох видів:

- системи на кристалі, де в одній мікросхемі працюватимуть і хост і контролер;
- приймача і мікроконтролера, які взаємодіють через HCI;

- системи на кристалі і окремого мікроконтролера з призначенням для користувача додатком, які взаємодіють з нестандартного протоколу.

Структура типової системи на кристалі зображена на рис. 2.2.

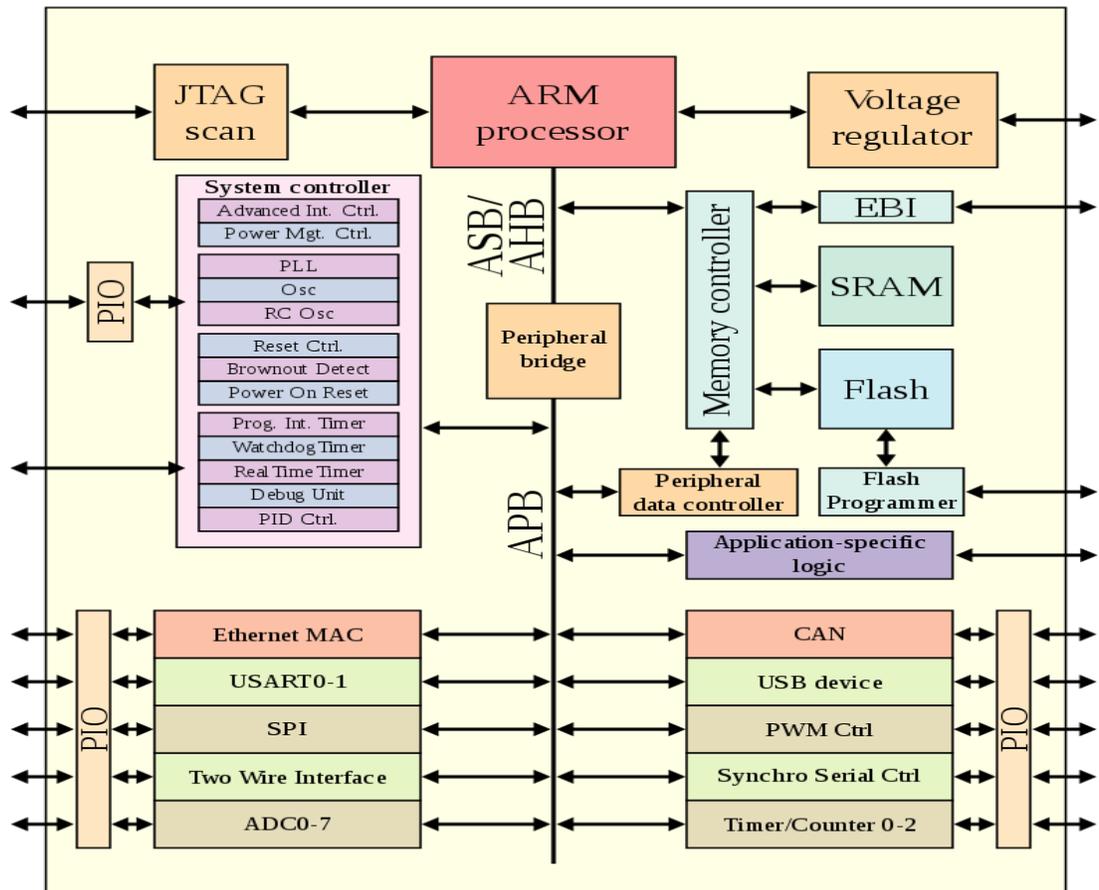


Рисунок 2.2 – Структура системи на кристалі побудованої на основі ARM-мікропроцесора

([https://uk.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0\\_%D0%BD%D0%B0\\_%D0%BA%D1%80%D0%B8%D1%81%D1%82%D0%B0%D0%BB%D1%96](https://uk.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D0%BD%D0%B0_%D0%BA%D1%80%D0%B8%D1%81%D1%82%D0%B0%D0%BB%D1%96))

Прості пристрої можуть використовувати систему на кристалі, а смартфони і планшети зазвичай використовують взаємодію за допомогою HCI,

третій тип використовується, коли до вже готового пристрою додають функції BLE.

Стандарт BLE не є універсальним способом організації бездротової передачі даних і очевидно, що звичайний Bluetooth, Wi-Fi і інші протоколи, займають свої ніші на ринку. Для BLE основна ніша – це пристрої з низькою пропускнуою здатністю і наднизьким енергоспоживанням.

Пропускна здатність BLE встановлена в специфікації, і становить 1 Mbps. Це теоретично верхня межа пропускнуої здатності, але реальні значення набагато нижче через службові дані, обмеження радіо і програмні обмеження.

Для оцінки реальної швидкості можна розглянути приклад: центральний пристрій (ведучий) встановлює з'єднання з периферійним (веденим) пристроєм. Поки з'єднання активне, специфікація визначає інтервал між двома послідовними подіями з'єднання (обміном пакетами) в межах від 7,5 мс до 4 с. Кількість пакетів, переданих в кожному інтервалі, залежить від реалізації, і зазвичай становить не більше шести. Кожен пакет може містити 20 байтів інформації користувача. Тому, встановивши мінімальний інтервал в 7,5 мс, можна отримати 133 події з'єднання та 120 байтів в кожній події. Отже, реальна швидкість передачі даних складе 125 кбіт/с.

Це значно нижче, ніж теоретичний максимум BLE, але конкретні пристрої можуть додати ще більше обмежень. Смартфон або планшет може бути зайнятий обміном даними з іншими пристроями, а також конкретна реалізація стека неминуче має обмеження, тому центральний пристрій майже ніколи не може обробляти дані з максимальною швидкістю. Тому на практиці, в кращому випадку максимальна пропускна здатність може досягати 5–10 кбіт в секунду.

У світі, де швидше означає краще, 10 кбіт/с може виглядати дуже скромним результатом, але це зайвий раз підкреслює головну мету BLE – низьке енергоспоживання. Навіть швидкість передачі даних в 10 кбіт/с буде швидко

витрачати батарею, тому в специфікації BLE вжито всі необхідні заходи, щоб скоротити енергоспоживання.

Найпростіший спосіб знизити споживання енергії – вимкнути передавач. Тому в BLE передавач вимикається, як тільки це можливо і на максимально тривалий час. Передача даних відбувається групами пакетів на певній частоті: невеликі порції даних, що передаються через тривалі проміжки часу, що задаються користувачем в межах від 4,7 мс до 4 с.

Дійсна дальність радіозв'язку залежить від безлічі факторів (оточення, антени, орієнтації в просторі), але BLE призначений для з'єднань на коротких відстанях. Потужність передавача зазвичай налаштовується в межах від «мінус» 30 до 0 dBm, велика потужність вимагає більше енергії від батареї, знижуючи тривалість роботи.

На відкритій місцевості можна досягти дальності зв'язку до 50 метрів, але типовими значеннями є 10–20 метрів, оскільки основна ідея BLE це економія енергії [8].

### **Режими роботи та структура даних**

З погляду процесів передавання та обробки даних, що використовують BLE, діляться на центральні і периферійні, а обмін даними може відбуватися за допомогою встановлення з'єднання або без нього. Периферійний пристрій постійно розсилає на заданих частотах ширококомвні пакети, які отримують всі центральні пристрої в зоні радіовидимості. При розсилці ширококомвних пакетів периферійний пристрій не отримує ніякої інформації у відповідь. Такий пакет складається з 31 байта, і якщо цього не достатньо, то периферійний пристрій може обробляти запит певного виду, у відповідь на який буде відправляти інший пакет з 31 байта. Такий вид взаємодії зображений на рис.

2.3, де вгорі периферійний пристрій, який надсилає широкомовні пакети, а внизу запит додаткового пакета від центрального пристрою. За цим запитом може слідувати запит на встановлення з'єднання.

Такий спосіб використання BLE ідеально підходить для малоспоживаючих датчиків, пристроїв індикації, але у випадках, коли необхідно передавати більше 62 байт, або необхідний двобічний обмін даними, використовується режим роботи BLE з установкою з'єднань. Цей режим показаний на рис. 2.4, після отримання широкомовних пакетів, центральний пристрій може відправити периферійному запит на встановлення з'єднання, під цим терміном розуміється тільки періодичний обмін даними в певні моменти часу.

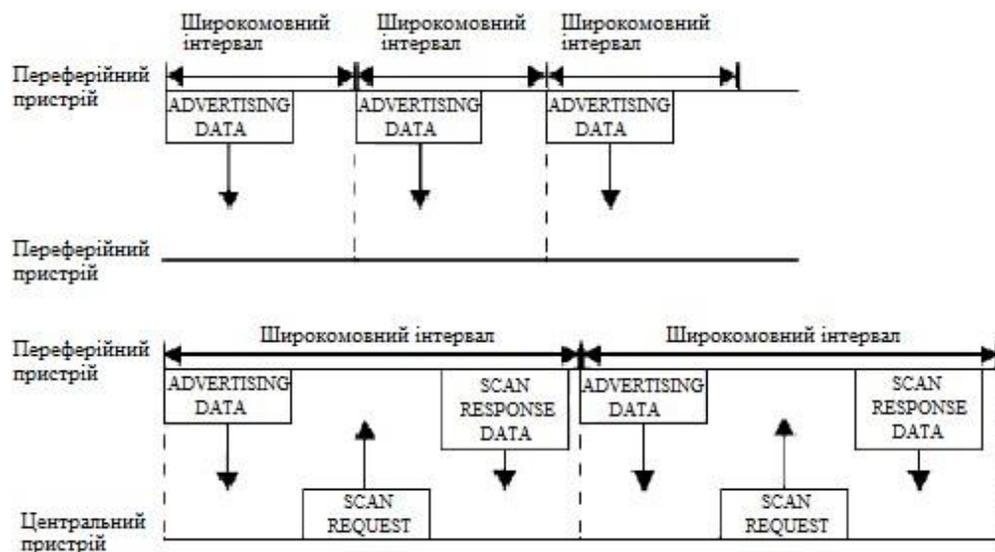


Рисунок 2.3 – Взаємодія BLE пристроїв без встановлення з'єднання

([https://www.researchgate.net/figure/Passive-vs-Active-Scanning-44\\_fig6\\_320957688](https://www.researchgate.net/figure/Passive-vs-Active-Scanning-44_fig6_320957688))

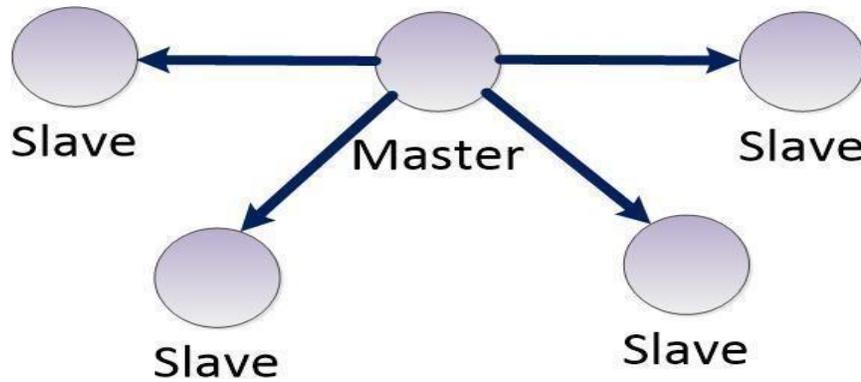


Рисунок 2.4 – Взаємодія BLE пристроїв зі встановленням з'єднання  
<https://itmaster.biz.ua/directory/standarts/ble.html>

Пакети даних, що передаються від пристрою Bluetooth Low Energy, мають формат, визначений базовою специфікацією Bluetooth, та складаються з кількох структурних компонентів, які зображені на рис. 2.5.



Рисунок 2.5 – Пакет даних Bluetooth Low Energy  
<https://wiki.cusu.edu.ua/index.php/ARP>

Преамбула є однобайтовим значенням, що застосовується для синхронізації приймача під час обробки сигналу. Для широкомовних пакетів її значення є сталим і дорівнює 0xAA. Адреса доступу для таких пакетів також фіксована та має значення 0x8E89BED6. Корисне навантаження пакета включає

заголовок і безпосередньо інформаційну частину. Заголовок містить дані про тип пакета, тоді як тип PDU визначає функціональне призначення пристрою. У ширококомовних застосуваннях використовуються три різні типи PDU, перелік яких наведено в таблиці 2.1

(<https://journals.indexcopernicus.com/api/file/viewByFileId/808516>).

Біт TxAdd використовується для визначення типу адреси передавача, що міститься в корисному навантаженні: публічна адреса відповідає значенню TxAdd = 0, тоді як випадкова — TxAdd = 1. Біт RxAdd у ширококомовних пакетах не задіяний і зарезервований для інших типів передавання.

Завершальною частиною переданого пакета є циклічний надлишковий код (CRC). CRC являє собою механізм виявлення помилок, який застосовується для контролю цілісності даних і захисту пакета від спотворень, що можуть виникати внаслідок завад у радіоефірі. Використання цього коду забезпечує надійну перевіряєність помилок та збереження коректності всіх пакетів, переданих бездротовим каналом.

Таблиця 2.1 – Типи PDU для ширококомовної передачі даних у BLE

Тип PDU	Назва пакета	Опис
0	ADV_IND	Подія направлено оголошення з можливістю встановлення з'єднання
10	ADV_NONCONN_IND	Подія ненаправлено оголошення без підтримки з'єднання
110	ADV_SCAN_IND	Подія ненаправлено оголошення з підтримкою сканування

Корисне навантаження пакета містить адресу оголошувача разом із даними, визначеними користувачем, які передаються в оголошувальному пакеті

(user-defined advertised data), що показано на рис. 2.6. Зазначені поля використовуються для передавання широкомовних адрес маячків та пов'язаної з ними інформації [9].

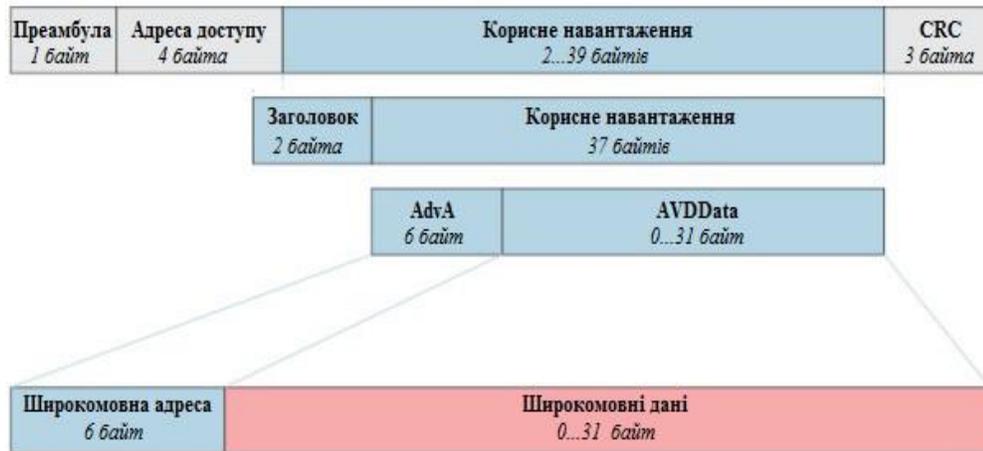


Рисунок 2.6 – Широкомовні дані Bluetooth Low Energy

(<https://studfile.net/preview/9109997/page:21/>)

Широкомовна адреса може мати публічний або випадковий тип. Публічна адреса відповідно до стандарту IEEE 802–2001 використовується як унікальний ідентифікатор організації (OUI), який надається через центр реєстрації IEEE. Компанія Texas Instruments забезпечує присвоєння IEEE-адрес для всіх своїх пристроїв. У технології Bluetooth Smart випадкові адреси можуть генеруватися безпосередньо маячком і поділяються на три основні типи, що наведені на рис. 2.7: статичні, нерозв'язні приватні та розв'язні приватні (resolvable private). Статична адреса залишається незмінною протягом усього часу роботи пристрою та може бути змінена лише після його перезавантаження.



## Рисунок 2.7 – Типи адрес маячка BLE

Приватні адреси можуть змінюватися з плином часу, при цьому розв'язна приватна адреса дає змогу відновити справжню адресу пристрою. Нерозв'язна приватна адреса також є змінною, що відрізняє її від статичної, яка залишається сталою протягом сеансу роботи. Використання випадкових адрес є одним із механізмів забезпечення конфіденційності, оскільки воно унеможливорює відстеження пристрою сторонніми особами. Процедури формування випадкових адрес регламентуються базовою специфікацією Bluetooth.

Дані, що передаються, можуть бути структуровані відповідно до спеціалізованих форматів, визначених організацією Bluetooth SIG; приклади таких форматів наведено в таблиці 2.2.

(<https://journals.indexcopernicus.com/api/file/viewByFileId/808516>).

Таблиця 2.2 – Види оголошувальних даних

Тип даних AD	Значення	Опис
Прапори	0x01	Можливості виявлення пристрою
Сервіс UUID	0x02...0x07	Служби GATT-пристрої
Локальне ім'я	0x08...0x09	Ім'я пристрою
Рівень потужності TX	0x0A	Вихідна потужність пристрою
Дані, специфічні для виробника	0xFF	Визначаються користувачем
Тип даних AD	Значення	Опис
(Manufacturer-Specific Data)		

Далі розглянемо прапори та дані, що є специфічними для конкретного виробника. Перші три байти широкомовних даних описують функціональні можливості пристрою, що є вимогою базової специфікації. Формат цих байтів наведено в таблиці 2.3.

(<https://journals.indexcopernicus.com/api/file/viewByFileId/808516>).

Таблиця 2.3 – Прапори типів даних установки з'єднання BLE

Байт	Біт	Прапор/Значення	Опис
0	–	0x02	Довжина поля даних
1	–	0x01	Прапор типу AD (GAP)
2	0	Обмежений режим виявлення LE	Тривалість оголошення — 180 с
–	1	Загальний режим виявлення LE	Необмежений час оголошення
–	2	Підтримка BR/EDR відсутня	–
–	3	Одночасна підтримка LE та BR/EDR (контролер)	–
–	4	Одночасна підтримка LE та BR/EDR (головний пристрій)	–
–	5...7		Зарезервовані

Прапори режимів виявлення являють собою замасковані біти, можливі значення яких наведено в таблиці 2.3. У випадку, коли жоден із бітових прапорів не встановлено, поле з відповідним типом даних може бути відсутнім.

Дані, що є специфічними для виробника, використовуються для передавання додаткової інформації. Під час їх формування застосовується

спеціальний прапорець, який вказує на наявність таких даних у пакеті. Перші два байта самих даних повинні бути кодом ідентифкатора компанії [9].

Формат пакета Bluetooth Low Energy дозволяє пристрою передавати 25 байтів даних, специфічних для виробника, якщо оголошення має тип ненаправленого оголошення (ADV\_IND) або сканування ненаправленого оголошення (ADV\_SCAN\_IND), при цьому потрібен прапор режиму виявлення. Для непоєднуваного ненаправленого оголошення (ADV\_NONCONN\_IND) максимальна довжина даних в полі «Manufacturer-Specific Data» може досягати 28 байт. Ці дані визначаються виробником і можуть містити будь-яку інформацію.

Широкомовні дані також можуть передаватися у стандартизованому форматі. На практиці застосовувалися два основні стандарти: iBeacon, розроблений компанією Apple, та AltBeacon від Radius Networks. Стандарт iBeacon захищений ліцензією MFi та забезпечує сумісність з усіма пристроями на базі iOS.

Для зниження енергоспоживання маячки більшу частину часу перебувають у режимі очікування та активуються лише на короткі інтервали для передавання інформації. Часовий проміжок між такими сеансами передавання називається інтервалом оголошення. Для непоєднаних маячків мінімальне значення цього інтервалу становить 100 мс, тоді як для маячків, що підтримують встановлення з'єднання, — не менше 20 мс. До основного інтервалу додається псевдовипадкова затримка в межах 0–10 мс, що дозволяє кільком маячкам коректно функціонувати одночасно навіть у разі синхронного початку передавання.

Вибір інтервалу оголошення здійснюється з урахуванням компромісу між рівнем енергоспоживання та часом очікування отримання даних. Збільшення інтервалу сприяє зменшенню споживаної потужності, однак водночас подовжує

період, протягом якого спостерігач очікує на приймання широкомовного пакета.

Як правило, спостерігач виконує сканування з коефіцієнтом заповнення менше 100 %, що дає змогу знизити енергоспоживання або забезпечити час для роботи інших бездротових протоколів. Типовим прикладом є смартфони, у яких Bluetooth і Wi-Fi реалізовані на одному апаратному модулі. У випадку, коли навушники під'єднані через класичний Bluetooth, а з'єднання з точкою доступу підтримується через Wi-Fi, сканування Bluetooth Low Energy зазвичай відбувається лише протягом коротких активних інтервалів. Типи оголошувальних даних та формат даних, специфічних для виробника, наведено в таблиці 2.4.

(<https://journals.indexcopernicus.com/api/file/viewByFileId/808516>).

Таблиця 2.4 – Типи даних оголошення та формат даних в BLE

Байт	Значення	Опис
0	0x03...0x1F	Довжина поля оголошувальних даних
1	0xFF	Прапор даних, специфічних для виробника
2	0x0D	Старший байт ідентифікатора компанії
3	0x00	Молодший байт ідентифікатора компанії (наприклад, 0x000D — Texas Instruments)
4...31	—	Дані, визначені користувачем (додаткова інформація)

Спостерігач може здійснювати сканування в пасивному або активному режимі. У разі використання активного режиму, за умови його підтримки маячком, надсилається команда **Scan Request**, у відповідь на яку пристрій формує повідомлення **Scan Response**. Запит сканування являє собою порожній пакет, що не містить корисних даних, тоді як відповідь зазвичай включає

статичну інформацію, наприклад назву або модель пристрою. Формат і зміст відповіді повністю визначаються самим маячком і можуть містити дані з різних датчиків або іншу корисну інформацію. Під час пасивного сканування спостерігач не надсилає запитів сканування.

Живлення маячків може здійснюватися кількома способами. Виділяють три основні варіанти, наведені на рис. 2.8:

- підключення до постійного джерела живлення (USB, електрична мережа тощо);
- використання автономних елементів живлення (CR2032, AAA, літієві батареї тощо);
- застосування технологій збору енергії, зокрема сонячної, кінетичної та інших (energy harvesting).



Рисунок 2.8 – Варіанти електроживлення

<https://www.ti.com/lit/an/swra475a/swra475a.pdf?ts=1715886516565>

Зазвичай першочергово обирають батареї з ресурсом роботи, достатнім для більшості застосувань, які водночас дають змогу створювати компактні та недорогі пристрої. У деяких випадках можливе використання акумуляторних елементів живлення, зокрема в поєднанні з технологіями бездротової зарядки. Правильний вибір типу батареї має суттєве значення, оскільки окремі джерела

живлення можуть демонструвати низьку ефективність у режимах високого струмового навантаження. Ємність батареї визначається з урахуванням частоти передавання даних, а також необхідності їх подальшої обробки, такої як зчитування показників датчиків або виконання алгоритмів аналізу отриманої інформації. Взаємодія з датчиками зазвичай передбачає послідовне передавання даних через інтерфейси UART, SPI або I2C, що потребує додаткових енергетичних витрат і в окремих випадках може перевищувати споживання енергії під час радіообміну за протоколом Bluetooth Low Energy.

Якщо пристрій живиться від джерела постійного струму, то передбачається, що споживана потужність не є критичним параметром. А якщо цей параметр все-таки критичний, використовуються ті ж вимоги до проекта, що і для працюючого від акумулятора пристрою.

Збір енергії реалізується частіше для бездротових проектів з низьким енергоспоживанням, і маячок може бути оснащений джерелом збору енергії. Найбільш поширеними джерелами енергії є механічний тиск і сонячне випромінювання. Крім того, для живлення маячка може використовуватися навіть штучне освітлення в приміщенні.

Відстань. Теоретично відстань для заданої радіочастоти залежить від багатьох факторів, наприклад, таких як:

- чутливість радіоприймача;
- вихідна потужність радіопередавача;
- навколишнє середовище і перешкоди;
- характеристики антени.

Маячки, як правило, працюють на відстанях від сантиметрів до кілька сотень метрів. Максимальна вихідна потужність, яка визначається базовою специфікацією, дорівнює 10 дБм, при цьому може покриватися відстань до кількох сотень метрів, якщо усі вищезгадані фактори враховані.

## Частотний розподіл

На фізичному рівні BLE використовує частоту 2.4 ГГц ISM діапазону. Для з'єднання різних пристроїв доступний діапазон частот розділений на 40 каналів від 2,4000 ГГц до 2,4865 ГГц, з цих каналів 37 використовуються для передачі даних і останні три канали використовуються для службових цілей – встановлення з'єднання і розсилки ширококомовних пакетів. Розподіл частот по каналах приведено на рис. 2.9.

Стандарт використовує псевдо випадкову перебудову робочої частоти (FHSS – frequency hopping spread spectrum), через яку, при кожній події передачі даних відбувається стрибок між каналами



Рисунок 2.9 – Розподіл частот по каналах на фізично му рівні BLE

<https://modwiggler.com/forum/viewtopic.php?t=221356>

Значення hop визначається в процесі установки з'єднання і тому по-різному для кожного нового встановленого з'єднання. Цей спосіб мінімізує вплив інтерференції на частотах 2,4 ГГц, особливо між Wi-Fi і класичним

Bluetooth, що дозволяє пристроям BLE працювати поблизу потужних передавачів Wi-Fi.

Для модуляції сигналу використовується частотна модуляція з фільтром Гаусса (GFSK), така ж, як у класичній технології Bluetooth і багатьох інших протоколах ISM діапазону. Швидкість модуляції BLE фіксована на 1 Mbit/s, що є верхньою фізичною межею швидкості передачі даних за цією технологією.

На транспортному рівні апаратно реалізуються формування пакетів, перевірка контрольних сум і шифрування, а програмно здійснюється управління підключенням до інших пристроїв. Варто відзначити, що BLE властива асиметрія в енергоспоживанні: центральний пристрій споживає набагато більше енергії, ніж периферійний пристрій. Транспортний рівень є єдиним, що пред'являє вимоги жорсткого реального часу. В якості адреси використовується 48-бітний номер, унікально ідентифікує пристрій, він може бути, як заданий на заводі, так і запрограмований або розрахований в ході експлуатації пристрою. Кожен пристрій може сформувати список адрес, з якими буде працювати, при цьому пакети від всіх інших пристроїв будуть проігноровані.

Як зазначалося вище, всі периферійні пристрої постійно розсилають ширококомовні пакети, період їх розсилки може змінюватися в межах від 20 мс до 10,24 с. При цьому вони можуть розсилатися на будь-який з трьох доступних для цього частот, а приймач може бути не синхронізований. Очевидно, що чим коротше інтервал, тим більша ймовірність їх прийому і швидкість передачі даних, але вище і енергоспоживання.

Якщо використовується режим з установкою з'єднання, то після прийому широкомовного пакета, центральний пристрій відправить запит на установку з'єднання. Основними параметрами в цьому запиті є інтервал між послідовними обмінами пакетами і затримка з'єднання, тобто кількість пакетів, які можуть бути втрачені без обриву з'єднання. Ці параметри можуть

змінюватися як центральним пристроєм, так і периферійним в уже встановлене з'єднання [11].

## **Питання безпеки у протоколі BLE**

За безпеку в стеку протоколів BLE відповідає окремий протокол. У ньому визначаються дві ролі: ініціатор і відповідач, ініціатор завжди є центральним, а відповідач периферійним пристроєм.

Безпека в BLE будується на здійсненні захищеного з'єднання, через яке відбувається обмін даними. Побудова подібного з'єднання виконується в два етапи: сполучення і зв'язування. Під сполученням розуміється обмін незашифрованими службовими даними, за яким слід зашифрований обмін на етапі зв'язування. Процедура сполучення включає обмін пакетами для генерації тимчасового ключа шифрування (STK) на обох сторонах з'єднання. Останній крок процедури (незалежно від того, чи послідує потім зв'язування чи ні) це шифрування лінії попередньо згенерованих STK. Протягом обміну пакетами на етапі сполучення, пристрої домовляються про використання одного з наступних способів генерації STK.

Відображення ключа (passkey display) має на увазі, що одна зі сторін з'єднання відображає випадково генерований шестизначний код і інша сторона запитує введення цього ключа (або в деяких випадках обидві сторони допускають введення ключа, якщо немає можливості відобразити). Це дає захист від атаки людина в середині і використовується, коли можливо.

Додатковий канал зв'язку (out of band), коли використовується цей метод, для вироблення ключа застосовують будь-якої інший спосіб зв'язку, наприклад, такий як бездротова технологія NFC. Цей метод також дозволяє захиститися від атаки людина в середині. З усіх перерахованих способів максимальну

безпеку з'єднання зможе забезпечити тільки додатковий двобічний канал зв'язку для обміну ключами.

Для шифрування даних BLE використовує алгоритм AES-CCM (128 біт), це забезпечує достатній рівень захисту. Для шифрування використовується ключ зв'язування (LTK), що зберігається для кожного підключеного пристрою, з яким було виконано сполучення. При повторних запитах буде використаний ключ LTK. Можна виділити три стани, в яких може перебувати BLE-пристрій:

- Сполучення (перше з'єднання);
- Встановлення (повторні з'єднання);
- Обмін даними.

На етапі сполучення можливі як пасивна (коли злоумисник не втручається в обмін даними), так і активна (коли злоумисник втручається) передачу даних. За допомогою простого прослуховування ініціалізації з'єднання злоумисник зможе отримати всю необхідну інформацію для розшифровки подальшого обміну даними. Для здійснення даної атаки злоумисникові досить знаходитися в межах радіовидимості безпосередньо в момент встановлення з'єднання. Секретний ключ розраховується на основі випадкової величини, адреси пристрою і PIN-коду, який вводиться користувачем або задається виробником пристрою. Саме PIN-код є єдиною величиною, використовуваною при розрахунку секретного ключа, яка невідома злоумисникові. Він являє собою шестизначне десяткове число, яке можливо визначити за допомогою перебору. Це є серйозною уразливістю. Як активну атаку на даному етапі можна розглядати атаку людина-в-середині, якщо сполучення здійснюється способом простої роботи.

Якщо процес сполучення був упушений зловмисником, то подальший злом представляється скрутним, тому що стійкість шифру AES–CCM (128 біт) досить висока. У цьому випадку може бути застосована атака на пересопряженні змушує пристрій почати процедуру сполучення заново.

Якщо зловмисником ні підслухано процес сполучення і йому не відомий LTK, то з'єднання видається цілком стійким до алгоритмічних атак. Але на даному етапі можливі наступні проблеми: атака на відмову в обслуговуванні, глушіння сигналу на частоті 2.4 ГГц і ланцюжок ретрансляторів.

Так як можуть бути перехоплені ширококомовні пакети, то зловмисниками можуть бути зроблені пристрої–двійники, які будуть заважати роботі справжнього пристрою. Також зловмисник може здійснити глушіння сигналу на частоті 2.4 ГГц, оскільки для ширококомовних пакетів використовуються тільки три канали, здійснити це буде не складно. Ланцюжок ретрансляторів дозволить організувати між двома, розташованими на відстані один від одного, справжніми пристроями ланцюжок з повторювачів пакетів, пристрої зможуть встановити з'єднання. При цьому злому з'єднання не відбудеться, але справжні пристрої виявляться на явно близькій відстані один від одного, що може бути небезпечним стосовно автомобільної системи безпеки.

Для фільтрації пристроїв за адресами BLE надає можливість організації списку довірених пристроїв, тобто пристроїв пакети від яких не будуть відфільтровані. Також надається можливість зберегти дані про пристрій, що динамічно змінюють свою адресу для запобігання відстеження зловмисником, для цього необхідний окремий ключ (IRK), обмін якими відбувається на етапі сполучення.

Таким чином, протокол BLE не забезпечує необхідний рівень захисту інформації та потребує доопрацювання на рівні додатку. Основна вразливість – це процес формування секретного ключа на етапі сполучення. З трьох

вбудованих в протокол способів необхідний рівень захисту інформації забезпечує тільки додатковий канал зв'язку. Тому необхідно удосконалити процедуру обміну ключами.

Коли клієнт хоче прочитати або записати значення атрибута на сервері, він відправляє запит на читання або запис, що містить його дескрип тор. Сервер відповість або значенням атрибута, або підтвердженням. У разі операції читання, це повідомить клієнтові, що можна прийняти значення і визначити що це за дані по UUID. У разі ж операції записи, клієнт очікує отримання даних відповідно до атрибута і сервер може скасувати операцію, якщо це не так.

У відкритому діапазоні частот 2,4 ГГц, який використовується Bluetooth Low Energy, працюють також багато інших бездротових протоколів, такі як Wi-Fi і ZigBee. У цій смузі частот можливі і потенційні перешкоди від домашніх приладів, такі як мікрохвильові печі. Подібна активність в радіоефірі може заважати роботі пристроїв Bluetooth Low Energy.

Широкомовна передача відбувається послідовно, зображених на рис. 2.10.

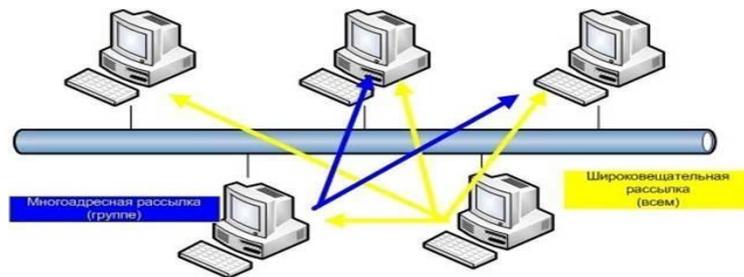


Рисунок 2.10 – Широкомовна передача оголошень

([https://elib.lntu.edu.ua/sites/default/files/elib\\_upload/12/teoretic/lec5.htm](https://elib.lntu.edu.ua/sites/default/files/elib_upload/12/teoretic/lec5.htm))

Канали 37, 38 і 39 були обрані для того, щоб не вступати у взаємодію з трьома зазвичай використовуваними каналами Wi-Fi – 1, 6 і 11. Полоса частот і канали Bluetooth Low Energy зображені на рис. 2.11.

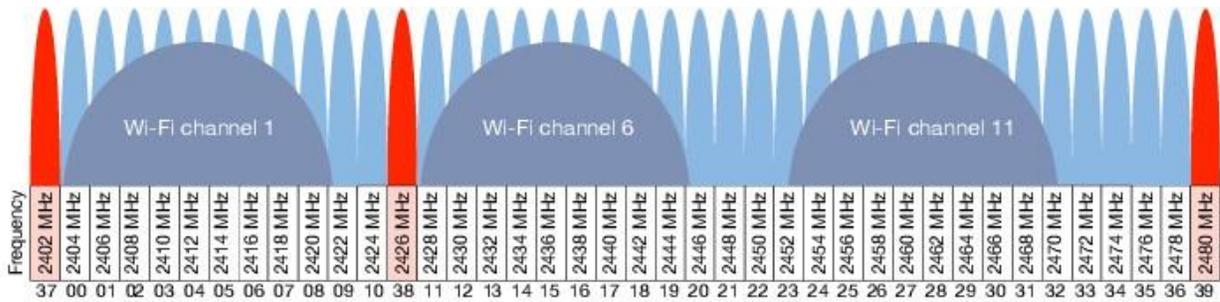


Рисунок 2.11 – Полоса частот і канали Bluetooth Low Energy

([https://www.semanticscholar.org/paper/Empirical-analysis-and-modeling-of-Bluetooth-\(BLE\)-Nikoukar-Abboud/b1181e101f1ddd883776deb0d9c09daf6495002f](https://www.semanticscholar.org/paper/Empirical-analysis-and-modeling-of-Bluetooth-(BLE)-Nikoukar-Abboud/b1181e101f1ddd883776deb0d9c09daf6495002f))

Втім, технологія Wi-Fi характеризується значно вищою вихідною потужністю — до 23 дБм, тоді як для Bluetooth Low Energy максимально допустиме значення становить 10 дБм. Це призводить до того, що у випадку розміщення маячка на незначній відстані від джерела Wi-Fi прийняті від нього дані можуть зазнавати спотворень. Причиною цього є паразитне випромінювання Wi-Fi-модуля на побічних каналах, яке накладається на сигнал маячка, що проілюстровано на рис. 2.13. На наведеному зображенні спостерігаються три піки вихідної потужності маячка на широкомовних каналах 37, 38 і 39 у порівнянні з максимальним рівнем потужності сигналу Wi-Fi під час передавання даних зі швидкістю 24 Мбіт/с.

Незважаючи на те, що оголошувальні канали Bluetooth Low Energy спеціально розташовані в діапазоні 2,4 ГГц з метою мінімізації взаємного впливу з найбільш поширеними каналами Wi-Fi, на рис. 2.12 показано, що проблеми сумісності з окремими приватними каналами Wi-Fi все ж можуть виникати. Водночас слід зауважити, що у наведеному експерименті частотний

спектр вимірювався для маячка, розміщеного безпосередньо на корпусі Wi-Fi-пристрою.

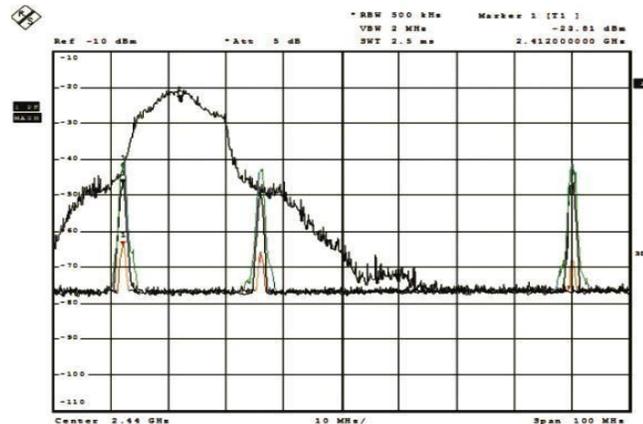


Рисунок 2.12 – Спектр Wi-Fi в порівнянні з мовними каналами маячка

([https://www.researchgate.net/figure/Spectrum-for-Wi-Fi-Channel-1-versus-BLE-Beacon-Advertisements-Channels-31\\_fig5\\_343031212](https://www.researchgate.net/figure/Spectrum-for-Wi-Fi-Channel-1-versus-BLE-Beacon-Advertisements-Channels-31_fig5_343031212))

Залежно від програмної конфігурації маячка до питань сумісності висуваються різні вимоги. Навіть невелика кількість одночасно активних маячків може створювати взаємні завади, що призводить до втрати пакетів даних. Як уже зазначалося, у загальному випадку технологія Wi-Fi характеризується вищою передавальною потужністю та займає ширшу смугу в ISM-діапазоні 2,4 ГГц порівняно з пристроями Bluetooth Low Energy.

## **3 ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ**

### **Вибір технології та інструментарію**

Для реалізації поставленої задачі було обрано такі інструментальні засоби:

- середовище розробки MPLAB X IDE;
- програматор PICkit 3.

MPLAB X IDE є безкоштовним інтегрованим середовищем розробки, призначеним для створення вбудованих програмних рішень на базі мікроконтролерів сімейств PIC і dsPIC компанії Microchip. Дане середовище розроблене на основі платформи NetBeans та забезпечує повний цикл розробки програмного забезпечення, включаючи керування проєктами, редагування програмного коду, налагодження та програмування мікроконтролерів. MPLAB X підтримує роботу з 8-бітовими мікроконтролерами PIC і AVR, 16-бітовими PIC24 та dsPIC, а також 32-бітовими мікроконтролерами SAM (ARM) і PIC32 (MIPS).

Середовище MPLAB X призначене для використання з сертифікованими апаратними засобами налагодження та програмування, зокрема MPLAB ICD 3 і MPLAB REAL ICE, що забезпечують взаємодію з мікроконтролерами PIC через персональний комп'ютер. Окрім цього, MPLAB X підтримує програмні рішення, створені з використанням програматора PICkit. Додатковими можливостями середовища є автоматизоване генерування програмного коду за допомогою MPLAB Code Configurator, а також використання плагінів MPLAB Harmony Configurator.

Під час розробки вбудованих систем застосовуються крос-асемблери або крос-компілятори, які виконуються на персональному комп'ютері, проте

формують машинний код для виконання на іншому мікропроцесорі або мікроконтролері. Процес компіляції програмного коду наведено на рис. 3.1.

PICkit 3 є апаратним відладчиком і програматором, структурну схему якого зображено на рис. 3.2. Даний пристрій належить до категорії недорогих і простих у використанні засобів внутрішньосхемного налагодження та програмування. Він керується з персонального комп'ютера під управлінням операційної системи Windows у поєднанні з середовищем розробки MPLAB X IDE.

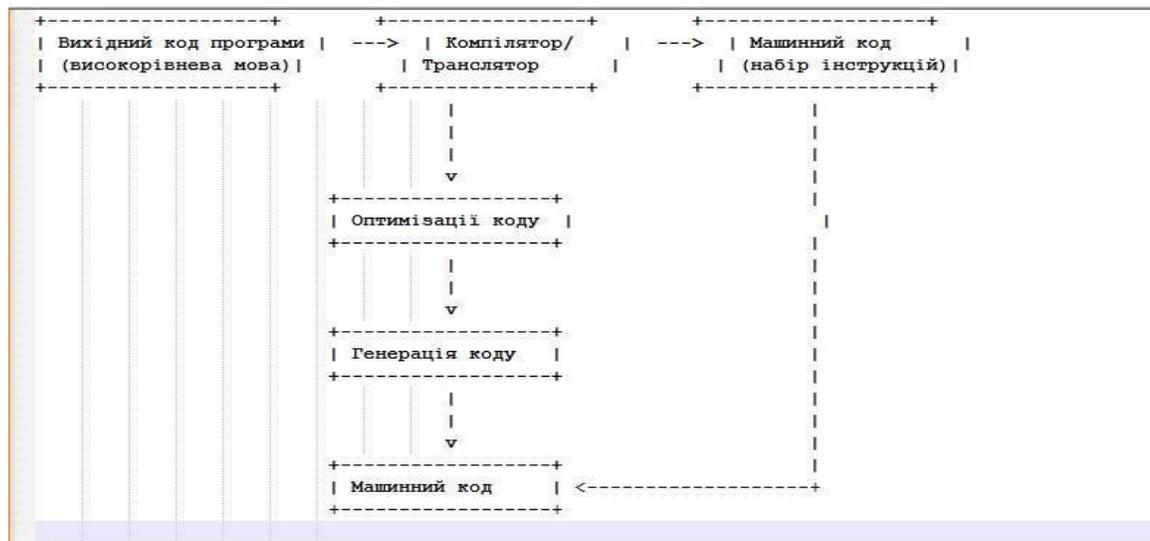


Рисунок 3.1 – Компіляція коду джерела у машинні інструкції

PICkit 3 In-Circuit Debugger/Programmer є налагоджувальною системою, що використовується для розроблення апаратного та програмного забезпечення на базі мікроконтролерів Microchip серій PIC (MCU) та цифрових сигнальних контролерів dsPIC (DSC). Функціонування даного пристрою ґрунтується на технології внутрішньосхемного послідовного програмування In-Circuit Serial Programming (ICSP), а також на вдосконаленому двопровідному міжпровідному послідовному програмуванні, що застосовується для 2-провідних послідовних інтерфейсів. Основні особливості PICkit 3:

- підтримка повної швидкості USB за допомогою стандартних драйверів операційної системи Windows;
- виконання в режимі реального часу;
- високошвидкісні процесори;
- вбудований монітор перенапруги та короткого замикання;
- робота при низькій напрузі до 5 В (діапазон 1,8–5 В);
- діагностичні світлодіоди (індикація живлення, активності та стану);
- зчитування та запис програми і даних у пам'ять мікроконтролера;
- стирання всіх типів пам'яті (EEPROM, ідентифікатор, конфігурація та програма) з верифікацією;
- периферійна затримка на точці розриву.

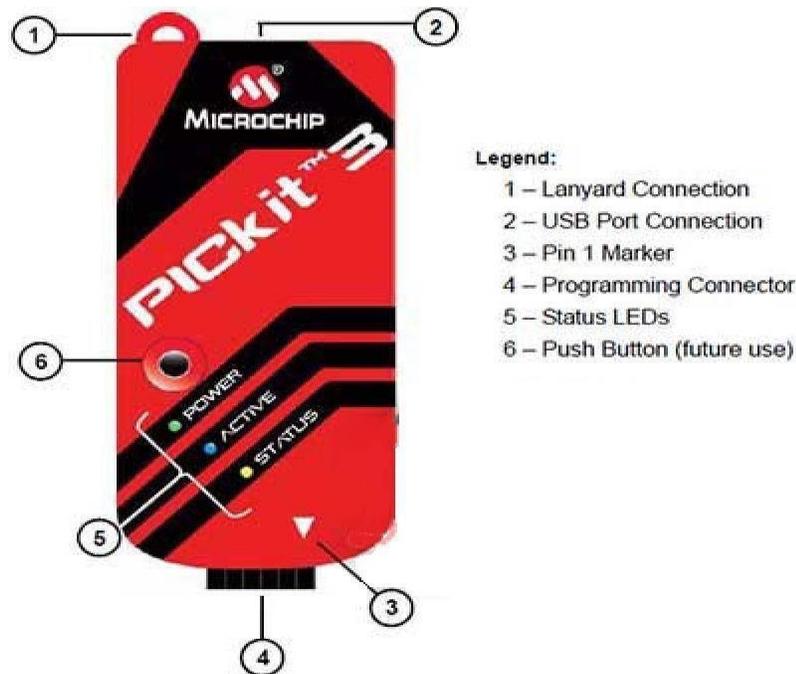


Рисунок 3.2 – Схема відладчика/програматора PICkit3

([https://www.researchgate.net/figure/shown-the-pickit-3-programmer\\_fig2\\_324766509](https://www.researchgate.net/figure/shown-the-pickit-3-programmer_fig2_324766509))

## Розробка програмного забезпечення системи

Запропонований алгоритм на основі нечіткої логіки (fuzzy-based) відстежує та контролює електроприлади в розумному будинку, плануючи для них оптимальний час роботи. Щоб забезпечити рентабельну експлуатацію, алгоритм управління гарантує баланс між попитом та пропозицією, враховуючи поточну ціну електроенергії та стан системи зберігання енергії.

Блок-схема алгоритму НЕМ представлена на рис. 3.3.

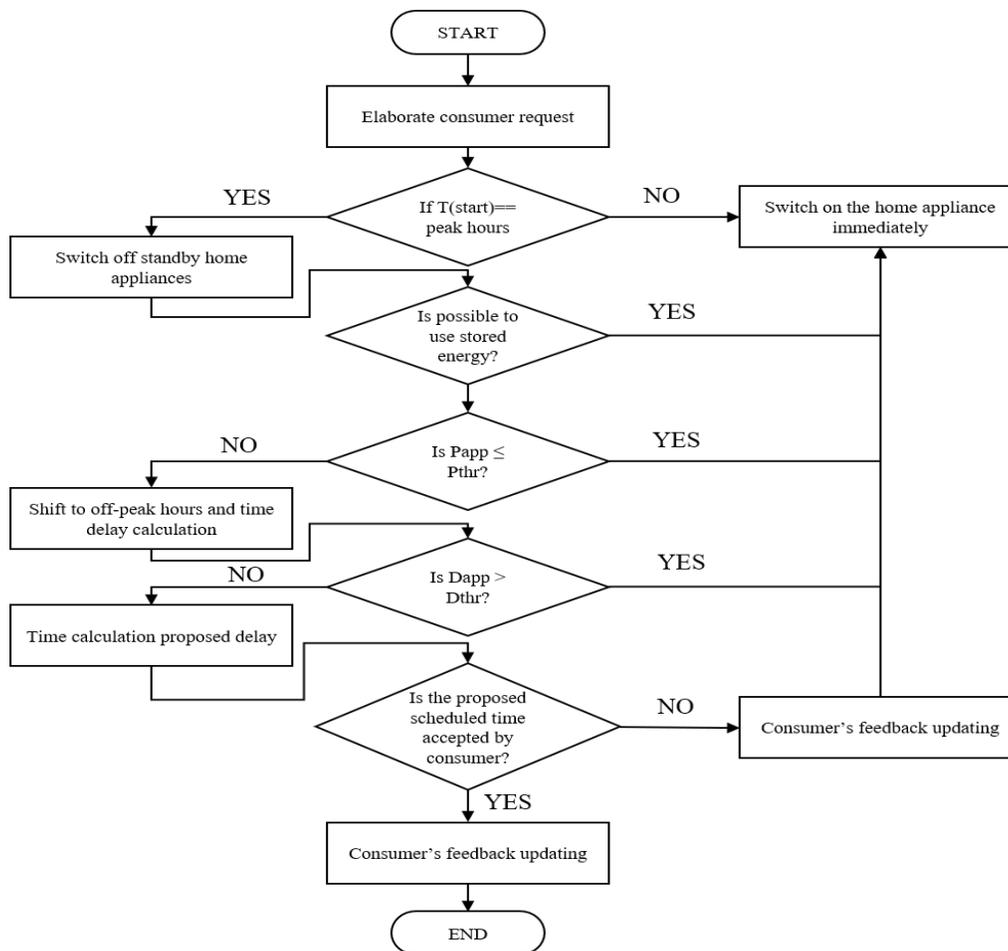


Рисунок 3.3 – Блок-схема алгоритму управління енергією у розумному будинку

(<https://www.sciencedirect.com/science/article/pii/S2214629620303571>)

Початкова точка визначається моментом часу, коли споживач вмикає прилад ( $T_{start}$ ). У цей момент надсилається відповідний сигнал до ЕРС, щоб повідомити про операцію. Далі ЕРС взаємодіє зі смарт-лічильником та системою зберігання енергії, щоб отримати інформацію про ціни за часом використання (TOU). Це дозволяє ЕРС знати актуальні ціни на споживання енергії в реальному часі і визначити, чи поточний час є піковим. На наступному етапі система керування енергоспоживанням (EPS) перевіряє, чи не припадає запланований час запуску приладу  $T_{start}$  на години пікового навантаження. Якщо запуск не потрапляє в піковий період, модуль керування енергією (EMU) дозволяє пристрою розпочати роботу без затримки. У протилежному випадку алгоритм переходить до подальшого етапу. З урахуванням того, що навіть прилади в режимі очікування споживають електроенергію, EPS здійснює перевірку таких пристроїв і за необхідності вмикає їх.

Далі EPS взаємодіє з локальною системою накопичення енергії з метою визначення обсягу доступної або збереженої енергії. Якщо наявних ресурсів достатньо, прилад запускається негайно без додаткової затримки. У разі нестачі енергії алгоритм переходить до етапу оцінювання потужності. Зважаючи на те, що в запропонованій схемі задано порогове значення потужності ( $P_{thr}$ ), EMU визначає потужність ( $P_{app}$ )  $i$ -го приладу та порівнює її з пороговим значенням для кожного запиту на запуск. Якщо виконується умова  $P_{app} \leq P_{thr}$  пристрій запускається негайно; у протилежному випадку алгоритм переходить до наступного кроку.

У такій ситуації робота приладу переноситься з пікового періоду на непіковий. Внаслідок цього в цикл роботи пристрою вводиться затримка ( $D_{app}$ ), яка визначається як різниця між часом запуску, рекомендованим EMU, та початковим часом запиту. Реалізація цієї затримки здійснюється із застосуванням методів перемикавання навантаження.

Слід зазначити, що величина затримки є обернено пропорційною рівню комфорту користувачів: надмірні затримки можуть істотно знижувати їхню задоволеність. Тому в запропонованій схемі враховується порогове значення затримки ( $D_{thr}$ , яке встановлюється індивідуально для кожного електроприладу).

На наступному етапі алгоритм порівнює затримки. Якщо  $D_{app}$  більше  $D_{thr}$ , прилад можна запустити негайно. В іншому випадку алгоритм переходить до наступного кроку. Це призводить до зміщення роботи приладів на години з порівняно низькими цінами на електроенергію.

Після цього модуль керування енергією (EMU) визначає величину затримки робочого циклу приладу та передає відповідне значення пристрою для встановлення часу його запуску. Запланований час, що відповідає затримці, обчислюється як різниця між часом початку роботи, рекомендованим EMU, та часом запуску, запитаним приладом. Надалі отримане значення часу разом зі зворотним зв'язком від користувача застосовується для коригування та оновлення запланованого графіка роботи приладу [4].

На наступному кроці споживач може прийняти або відхилити графік, запропонований EMU. Якщо споживачеві необхідно скористатися приладом негайно, він може відхилити графік, запропонований EMU, та запустити прилад, навіть якщо це буде в пікові години. В обох випадках модуль керування енергією (EMU) передає інформацію про прийняте користувачем рішення до системи зберігання енергії та розумного лічильника. Очевидно, що затримка, прийнята споживачем, буде нижчою за максимальний поріг затримки для цього пристрою.

Загальне рішення, представлене в цій роботі, базується на роботі контролера нечіткої логіки (Fuzzy Logic Controller), зображеного на рис. 3.4.

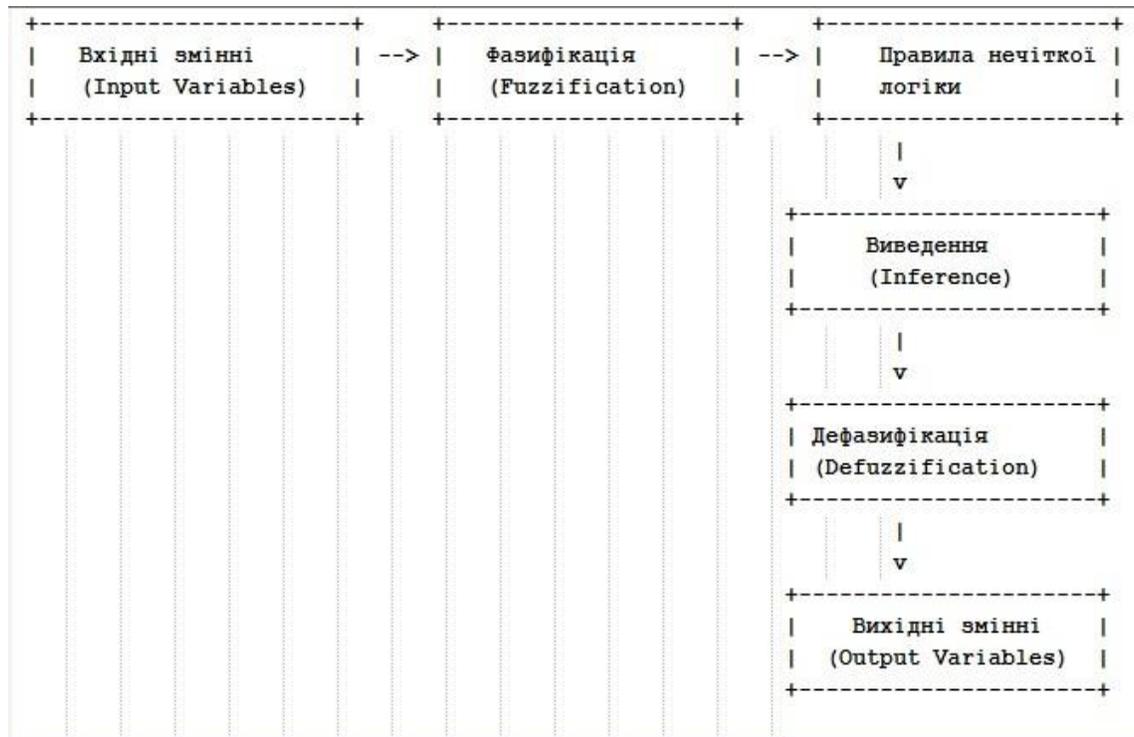


Рисунок 3.4 – Логічна схема роботи FLC

Незважаючи на те, що подання є спрощеним і узагальненим, воно наочно демонструє вхідні параметри FLC та результати, що формуються на виході. Запропонована система характеризується мінімальним обсягом обміну даними, оскільки інформація, яка передається до контролера, обмежується розрахованим значенням запланованої затримки та показниками зворотного зв'язку споживача для конкретного пристрою. Зазначені вхідні дані застосовуються для коригування запланованої затримки, яка з урахуванням наданого користувачем зворотного зв'язку повертається як вихідне значення FLC.

Це знижує обчислювальні витрати у порівнянні з іншими підходами, які використовують більш складні FLC з декількома контролерами, що працюють паралельно (наприклад, контролери у мережевому кластері) або з іншими методами, що враховують багато параметрів на вході до контролера.

Властивості запропонованого FLC не тільки дозволяють реалізувати це рішення на недорогому апаратному забезпеченні, але також сприяють скороченню часу обробки даних та зменшенню затримок для подальших логічних операцій у контролері.

### Програмна реалізація алгоритму на основі нечіткої логіки

Програмний продукт, призначений для функціонування на апаратній платформі, складається з кількох файлів і пакетів, кожен з яких виконує визначену функцію. У цьому підрозділі наведено опис розроблених бібліотек та основних методів програмної реалізації.

Бібліотека «definitions.h» є функціональною програмною одиницею, яка відповідає за ініціалізацію портів та змінних, а також визначення розміру буферів, як показано на рис. 3.5.

Взаємодія та обмін даними втілені у файлі «bluetooth.c». На рис. 3.6 представлена функція, що здійснює ініціалізацію BLE модуля з вказаним набором команд.

```
#ifndef DEFINITIONS_H
#define DEFINITIONS_H

// Загальні налаштування
#define MAX_DEVICES 10
#define BUFFER_SIZE 64

// Оголошення змінних
extern int deviceCount;
extern int deviceIDs[MAX_DEVICES];
extern float devicePower[MAX_DEVICES];
extern float deviceTemperature[MAX_DEVICES];
```

Рисунок 3.5 – Лістинг ініціалізації змінних у «definitions.h»

```

#include "definitions.h"

void initBLEModule() {
    setBLEBaudRate(9600);
    setBLEPower(3);
    setBLEMode(AUTO_MODE);
}

```

Рисунок 3.6 – Лістинг функції ініціалізації Bluetooth модуля

У рис. 3.7 зображена схема взаємодії між мікроконтролером та модулем RN4020 через UART інтерфейс.

На рис. 3.8 наведено логіку отримання повідомлень від модуля RN4020. Запуск програмного забезпечення здійснюється у головній функції main, яка розміщена у файлі main.c. Основна логіка роботи програми реалізована у вигляді нескінченного циклу, в межах якого виконується постійна перевірка вхідних повідомлень, що надходять до модуля, а також динамічне керування навантаженням. Фрагменти реалізації відповідної логіки наведено на рис. 3.9 та 3.10.

```

void communicateWithRN4020() {
    // Ініціалізація UART для взаємодії з RN4020
    initializeUART();

    // Відправка команди до RN4020
    sendCommandToRN4020("AT+COMMAND");

    // Отримання відповіді від RN4020
    char response = receiveResponseFromRN4020();

    // Обробка відповіді
    processResponse(response);
}

```

Рисунок 3.7 – Лістинг функції комунікації мікроконтролера та модуля  
RN4020

```
// Чекаємо, поки доступні дані для прийому
while (!UART_ReadyToReceive());

// Отримуємо символи даних з буфера UART
char receivedData = UART_Receive();

// Обробляємо отримані дані
processData(receivedData);
```

Рисунок 3.8 – Логіка отримання повідомлень від модуля

```
typedef enum { DISCONNECTED, CONNECTED } BluetoothState;

typedef struct {
    int Command;
} BluetoothPacket;

void sendADCValues() {
}

int BT_ReceivePacket(BluetoothPacket* packet) {
    return 1;
}
```

Рисунок 3.9 – Оголошення основних структур даних

```
int main() {
    BluetoothState bluetoothState = CONNECTED;
    BluetoothPacket incomingMessage;

    while (1) {
        if (bluetoothState == CONNECTED) {
            sendADCValues();
            int gotPacket = BT_ReceivePacket(&incomingMessage);
            if (gotPacket) {
                if (incomingMessage.Command == SWITCH) {
                    TRIAC_1 = (TRIAC_1 + 1) % 2;
                }
            }
        }
    }
}
```

Рисунок 3.10 – Лістинг фрагменту функції main()

### Розробка програмного комплексу клієнт–серверної частини

Система управління та моніторингу підключених пристроїв мережі має серверну архітектуру і включає в себе наступне:

- BLE сервер керує пристроями SmartSwitch у мережі за допомогою протоколу Bluetooth Low Energy;
- Web–сервер, який збирає дані від BLE сервера і, використовуючи алгоритм оптимізації споживання електроенергії, надсилає відповідні керуючі сигнали. Він також надає зовнішнє API для ручного управління та діагностики пристроїв користувачем.

На рис. 3.11 наведено детальну схему внутрішньої програмної архітектури системи.



Рисунок 3.11 – DFD діаграма системи оптимізації енергоспоживання

Далі, на рис. 3.12, подано деталізовану схему внутрішньої програмної архітектури системи оптимізації енергоспоживання. У цій системі комунікація відбувається за допомогою протоколу WebSocket. Інтерфейс користувача реалізований у вигляді веб-додатку, де керування та діагностика проводяться через веб-браузер.

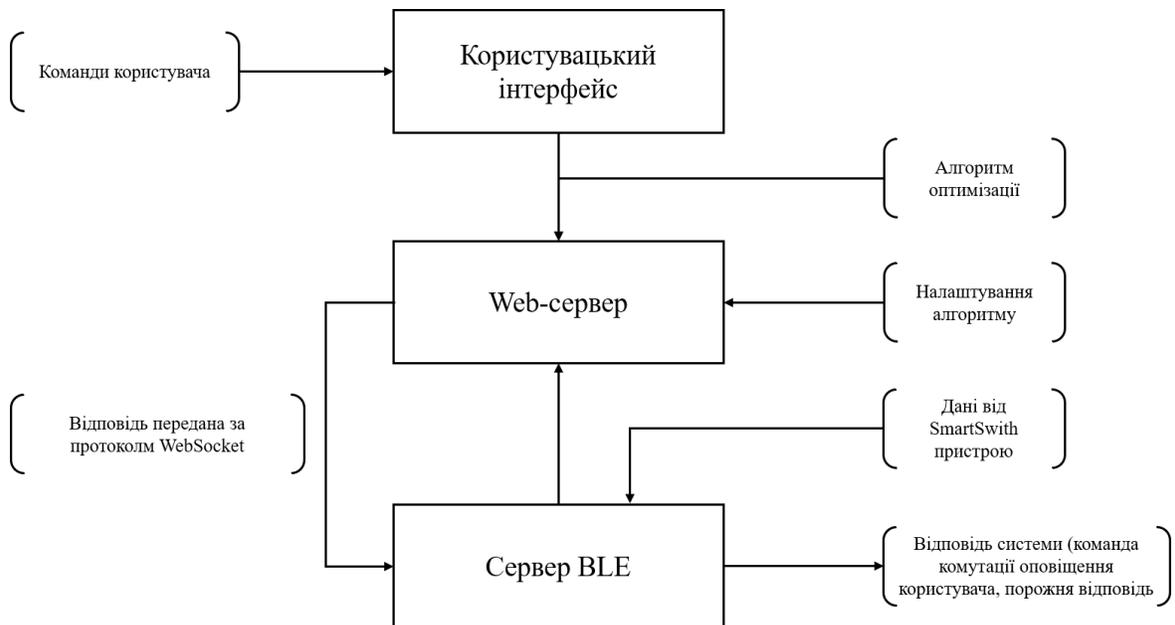


Рисунок 3.12 – Схема архітектури внутрішньої програмної взаємодії

Для реалізації зазначеної архітектури внутрішнього програмного комплексу було обрано такі технологічні засоби:

- noble – бібліотека для взаємодії з BLE пристроями;
- expressJS – фреймворк для веб-додатків на Node.js, розповсюджується під MIT ліцензією та призначений для створення веб-додатків та API;
- socketIO – JavaScript бібліотека для веб-додатків у реальному часі, що дозволяє здійснювати двосторонній зв'язок між веб-клієнтами та серверами;
- pg-promise – бібліотека, що надає інтерфейс для взаємодії з СУБД PostgreSQL;
- redis – проект in-memory бази даних, використовується як кеш для даного проекту;
- vueJS – модуль JavaScript для створення інтерфейсів користувача та односторінкових додатків;
- postgresSQL – безкоштовна система керування реляційними базами даних з відкритим вихідним кодом.

Для визначення активних BLE пристроїв у мережі використано стандартні методи бібліотеки Noble. Функція сканування представлена на рис. 3.13. Підключення та виявлення нових пристроїв здійснюються за допомогою колбеку discover, у межах якого реалізується вся необхідна взаємодія з пристроєм під час встановлення з'єднання та синхронізації даних. Відповідний фрагмент реалізації наведено на рис. 3.14 у лістингу цього колбеку.

```

function scanActiveBLEDevices() {
  noble.on('stateChange', function(state) {
    if (state === 'poweredOn') {
      noble.startScanning([], true, function(error) {
        if (error) {
          console.error('Error starting scanning:', error);
        }
      });
    } else {
      noble.stopScanning();
    }
  });
}

```

Рисунок 3.13 – Лістинг функції, що виконує сканування активних BLE-пристроїв

```

noble.on('discover', function(Peripheral) {
  console.log('Found device:', Peripheral.uuid, 'RSSI:', Peripheral.rssi);

  // При знаходженні нового пристрою підключаємося до нього
  Peripheral.connect(function(error) {
    if (error) {
      console.error('Error connecting to peripheral:', error);
      return;
    }
  });
}

```

Рисунок 3.14 – Лістинг колбек-функції підключення нового BLE-пристрою

Наступним етапом є реалізація логіки відключення активного BLE-пристрою, фрагмент якої наведено у лістингу програми на рис. 3.15. Після цього було реалізовано механізм синхронізації BLE-сервісів, що проілюстровано на рис. 3.16.

```

// Отримуємо дані з пристрою, потім відключаємося
Peripheral.once('disconnect', () => {
  console.log('Disconnected from device:', Peripheral.uuid);
});

```

Рисунок 3.15 – Лістинг функції відключення активного пристрою від мережі

```
peripheral.discoverServices([], (error, services) => {
  if (error) {
    console.error('Error discovering services:', error);
    peripheral.disconnect();
    return;
  }
}
```

Рисунок 3.16 – Лістинг функції синхронізації BLE сервісів

Процеси зчитування даних та підписки на їх оновлення реалізовані у відповідних методах `read` та `on`, приклади використання яких наведено на рис. 3.17.

```
service.discoverCharacteristics([], (error, characteristics) => {
  if (error) {
    console.error('Error discovering characteristics:', error);
    peripheral.disconnect();
    return;
  }
}
```

Рисунок 3.17 – Лістинг функцій зчитування та підписки на нові дані

## Сценарій роботи BLE сервера

Робота BLE-сервера здійснюється за таким сценарієм:

- встановлюється нове з'єднання за протоколом WebSocket із вебсервером;
- виконується перевірка та зчитування ідентифікаційних даних пристрою з кешу Redis, де вони зберігаються у серіалізованому форматі JSON. У цих даних поля `uuid` є унікальним ідентифікатором пристрою (Universally Unique Identifier), а поле `id` є ідентифікатором пристрою у базі даних PostgreSQL. Якщо пристрій присутній у мережі за даним ідентифікатором, то встановлюється з'єднання з ним;

- проводиться активація пристрою, передається відповідний сигнал у 16-ковому коді 0x01 методом write;
- після встановлення з'єднання з пристроєм відбувається синхронізація метаданих пристрою. Поля device\_id є ідентифікатором у базі даних, а current, temp та voltage – відповідні значення мікроконтролера.

На основі отриманих даних алгоритм або користувач приймає рішення щодо режиму споживання електроенергії, після чого формується та надсилається відповідне повідомлення. У цьому повідомленні параметр switch визначає рішення про комутацію BLE-пристрою: значення true відповідає переведенню пристрою в активний стан, тоді як false — у пасивний режим.

### Реалізація бази даних

Схема БД складається з двох таблиць:

- devices, де зберігаються поточні дані щодо пристрою;
- logs, яка відповідає за збереження журналу станів пристрою.

ER діаграма залежностей сутностей представлена на рис. 3.18.

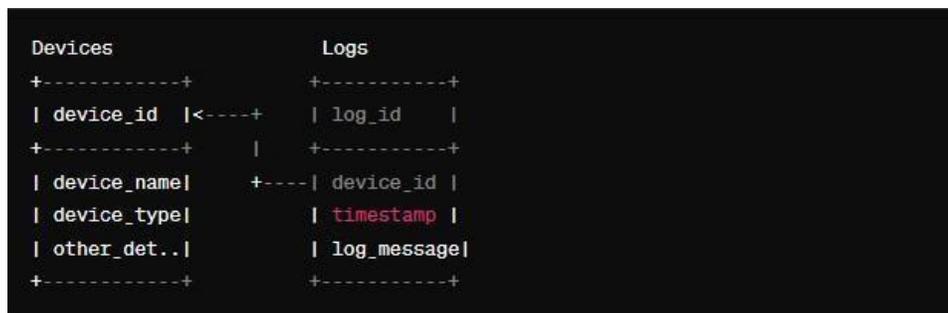


Рисунок 3.18 – ER діаграма залежностей сутностей у БД

Реалізація користувацького інтерфейсу передбачає налаштування системи у вигляді конфігураційної «мапи», що має структуру типу «ключ–значення». Сформована конфігурація кешується в системі Redis та зберігається протягом роботи застосунку, однак видаляється після перезавантаження системи.

До переліку основних конфігураційних параметрів належать:

- булевий параметр `AUTO_MODE`, який визначає режим функціонування системи, де значення `true` відповідає автоматичному керуванню пристроями, а `false` — ручному режиму;
- параметр типу `POWER` задає максимально допустиме значення потужності BLE-пристрою у ватах;
- параметр типу `TEMPERATURE` визначає гранично допустиму температуру BLE-пристрою у градусах Цельсія, при перевищенні якої пристрій автоматично вимикається.

Налаштування параметрів здійснюється за допомогою користувацького інтерфейсу, розробленого з використанням JavaScript-бібліотеки `Vue.js`. Приклад реалізації такого інтерфейсу наведено на рис. 3.19.



Рисунок 3.19 – Користувацький інтерфейс керування конфігураціями системи

Після підключення нового пристрою до системи всі наявні пристрої відображаються на головній сторінці застосунку, що продемонстровано на рис. 3.20.

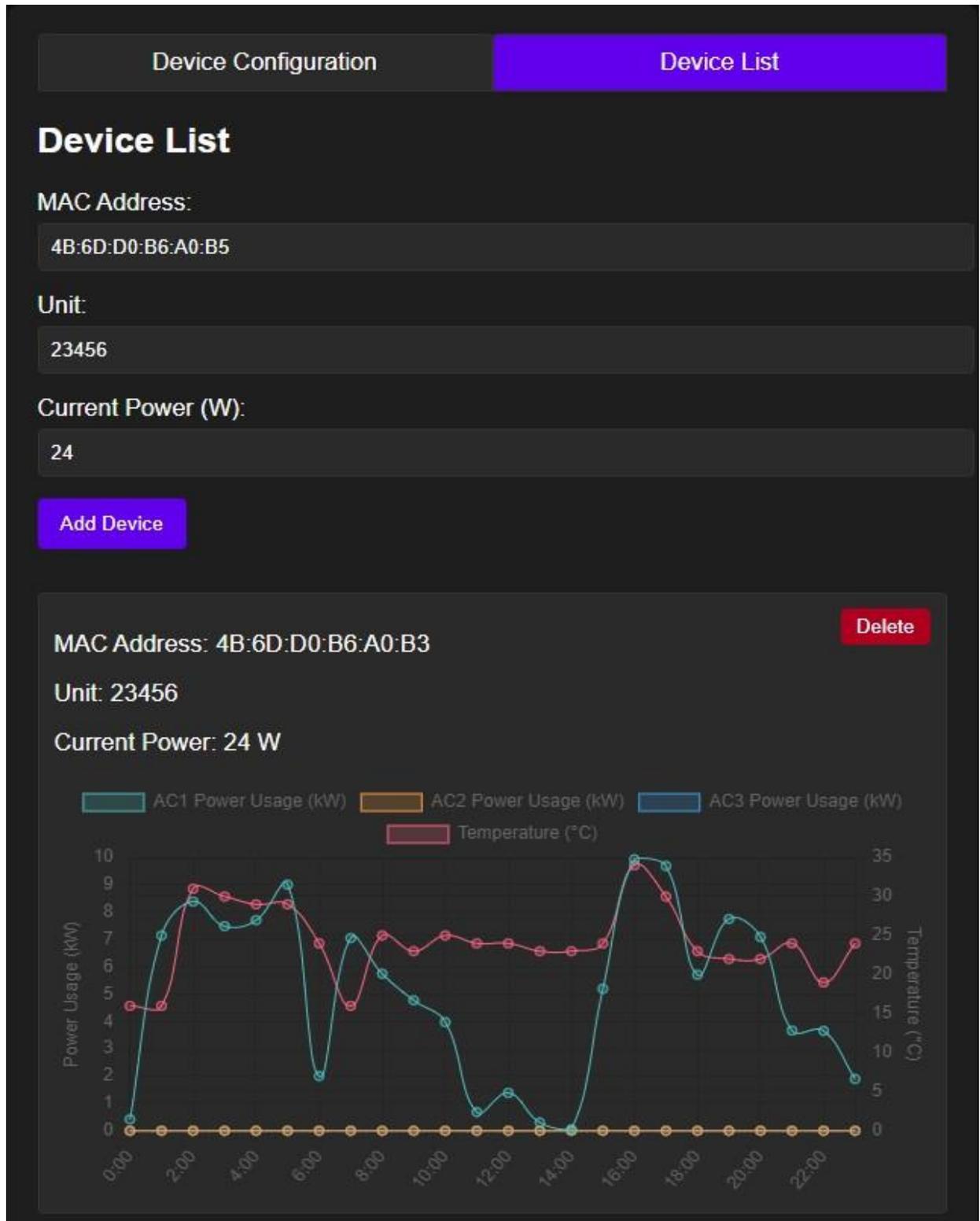


Рисунок 3.20 – Користувачький інтерфейс головної сторінки керування пристроями системи

Окрім цього, реалізовано графічний інтерфейс для моніторингу системних метрик упродовж заданого проміжку часу. Даний інтерфейс надає можливість оцінювати ефективність роботи окремого пристрою мережі та визначати його пікові навантаження, як показано на рис. 3.21.



Рисунок 3.21 – UI поточного статусу пристрою

## ВИСНОВКИ

У межах магістерської роботи було проведено комплексне дослідження концепції систем «розумного будинку», зокрема проаналізовано їх функціональні можливості, а також вимоги до апаратного та програмного забезпечення, необхідного для побудови таких систем. Результати дослідження підтвердили, що сучасні рішення у сфері автоматизації житлових і комерційних приміщень здатні суттєво підвищити рівень комфорту, безпеки та енергоефективності, а також забезпечити гнучке керування інженерними системами.

Основною метою роботи була розробка та аналіз інформаційної системи на основі технології Bluetooth Low Energy, спрямованої на підвищення енергоефективності систем розумного будинку. Для досягнення поставленої мети було запропоновано архітектуру системи, що забезпечує збір та обробку даних про стан електроприладів і дозволяє автоматизовано приймати рішення щодо їх увімкнення або вимкнення з урахуванням поточного навантаження внутрішньої електромережі. Реалізований підхід дає змогу здійснювати балансування навантаження, зменшувати пікові значення споживаної потужності, попереджати можливі аварійні ситуації та мінімізувати витрати на електроенергію.

У ході роботи було обґрунтовано вибір апаратної платформи на базі мікросхем сімейства Microchip, виконано їх порівняльний аналіз та визначено найбільш доцільне рішення для реалізації розроблюваної системи.

## ПЕРЕЛІК ПОСИЛАНЬ

1. Концепція розумного будинку: технології, можливості, перспективи [Електронний ресурс] – Режим доступу до ресурсу: <https://zounb.zp.ua/wp-content/uploads/2021/07/Rozumniy-budinok-pokazhchik-6.04.21-s-oblozhkoj>\ О.В. Іванова. – Техніка.
2. Розумний будинок: переваги, ризики, виклики: Електротехнічні системи та комплекси [Електронний ресурс]. – Режим доступу до ресурсу: [https://buidemo.com/ua/news/smart\\_systems/what-is-a-smart-home/](https://buidemo.com/ua/news/smart_systems/what-is-a-smart-home/) / Сидоренко І.П
3. Інтеграція технологій у розумних будинках: аналіз сучасного стану [Електронний ресурс] – Режим доступу до ресурсу: <https://zmist.pl.ua/publications/rozumnyj-budynok-ideyi-dlya-avtomatyzacziyi-vashoyi-oseli/> / В.М. Петренко.
4. Дослідження впровадження концепції розумного будинку в реальному житті [Електронний ресурс] – Режим доступу до ресурсу: [https://www.researchgate.net/publication/369990845\\_Dosvid\\_realizacii\\_koncepcii\\_rozumnogo\\_mista\\_v\\_krainah\\_so\\_rozvivautsa/](https://www.researchgate.net/publication/369990845_Dosvid_realizacii_koncepcii_rozumnogo_mista_v_krainah_so_rozvivautsa/) / Ковальчук О.С.
5. Технології Synco living для реалізації концепції розумного будинку. Поглиблене вивчення функцій Synco living в системах автоматизації розумного будинку [Електронний ресурс]. – Режим доступу до ресурсу: <https://ek.ua/ua/post/1990/618-что-такое-umnyy-dom-funkcii-vidy-sostavlyayuschie-i-ekosistemy/> / Д.В. Харченко [www.siemens.com/synco-living](http://www.siemens.com/synco-living).
6. Поглиблене вивчення функцій Synco living в системах автоматизації розумного будинку [Електронний ресурс]. – Режим доступу до ресурсу: <https://ek.ua/ua/post/1990/618-что-такое-umnyy-dom-funkcii-vidy-sostavlyayuschie-i-ekosistemy/> / Д.В. Харченко

7. Технологія розумного будинку: як AI створює простір, комфортний для життя [Електронний ресурс]. – Режим доступу до ресурсу: <https://www.everest.ua/tehnologiya-rozumnogo-budynku-yak-ai-stvoryuye-prostirkomfortnyj-dlya-zhyttya/> Основи програмної інженерії: навчальний посібник / Є. О. Зайцев – К.: КНТЕУ, 2017. – 423 с
8. Що таке розумний будинок? Все що потрібно знати про систему Розумний Дім [Електронний ресурс]. – Режим доступу до ресурсу: <https://bron.ua/article/schotake-rozumnij-budinok-vse-scho-potrбно-znati-pro-sistemu-rozumnij-dm/5/> Бородкіна, І.Л. Інженерія програмного забезпечення: навч. посіб. / І.Л.Бородкіна, Г.О.Бородкін. – Київ : ЦУЛ, 2019. – 204 с
9. Кучеров Д.П. Інженерія програмного забезпечення : навч. посіб. / Д.П.Кучеров, Є.Б.Артамонов. – Київ : НАУ, 2017. –386с
10. Табунщик Г.В, Кудерметов Р.К., Брагіна Т.І. Інженерія якості програмного забезпечення [Електронний ресурс]:навчальний посібник. – Запоріжжя: ЗНТУ, 2013. – 180.
11. ДСТУ 3008–2015 «Державний стандарт України. Документація. Звіти в сфері науки і техніки. Структура і правила оформлення»
12. Табунщик Г.В. Проектування та моделювання програмного забезпечення сучасних інформаційних систем / Г. В. Табунщик, Т.І. Каплієнко, О.А. Петрова – Запоріжжя : Дике Поле, 2016. – 250 с.
13. Sommerville I. Software Engineering— 9th ed. / Ian Sommerville. – Addison– Wesley, 2011. – 773 p.
14. Інженерія програмного забезпечення: рекомендаційний список літератури\ уклад. Н. С. Лузіна. – Чернігів : Наукова бібліотека НУ «Чернігівська політехніка», 2021. – 10 с

IoT-платформа для централізованого моніторингу та керування приладами розумного будинку



Київ 2025

Створення інформаційної системи за допомогою технології Bluetooth Low Energy для оптимізації електроенергії.



КОНЦЕПЦІЯ РОЗУМНОГО БУДИНКУ	ОПИС ТЕХНОЛОГІЇ BLUETOOTH LOW ENERGY	ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ
<ul style="list-style-type: none"><li>Головні вимоги до реалізації</li><li>Функціональні вимоги</li><li>Аналіз можливостей</li></ul>	<ul style="list-style-type: none"><li>Особливості технології BLE</li><li>Режими роботи та структура даних</li><li>Частотний розподіл</li><li>Питання безпеки у протоколі BLE</li></ul>	<ul style="list-style-type: none"><li>Вибір технології та інструментарію</li><li>Розробка програмного забезпечення системи</li><li>Сценарій роботи BLE сервера</li><li>Реалізація бази даних</li></ul>

# Поширені системи розумного будинку



Системи багатоканального озвучення



Система відеомоніторингу



Системи безпеки



Система клімат-контролю



Система освітлення

## Функціональні вимоги до елементів розумного будинку



Датчик наявності



Датчик інтерфейсу



централізовані - єдині високопродуктивні центральні контролери;



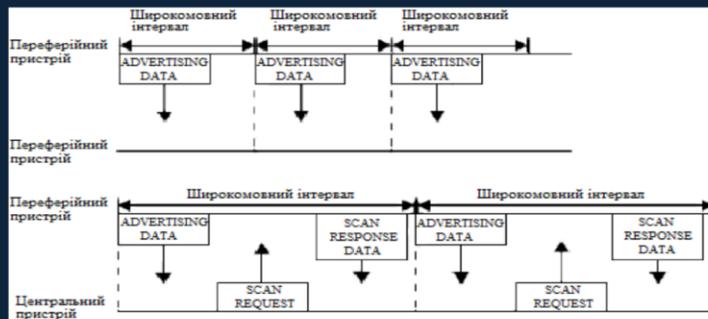
децентралізовані - декілька більш простих регіональних контролерів, кожен з яких відповідає за певну зону або групу пристроїв



## Особливості технології BlueTooth Low Energy

Логічно BLE складається з трьох основних рівнів:

- рівня додатків, призначеного для програм користувача, що використовують можливості BLE для реалізації конкретного призначення;
- рівня контроллера, нижніх рівнів стека протоколів BLE;
- рівня хоста, верхнього рівня стека протоколів BLE.



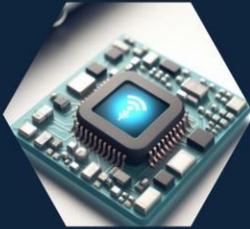
З точки зору обміну даними пристрої, що використовують BLE, діляться на центральні і периферійні, а обмін даними може відбуватися за допомогою встановлення з'єднання або без нього. Периферійний пристрій постійно розсилає на заданих частотах широкомовні пакети, які отримують всі центральні пристрої в зоні радіовидимості. При розсилці широкомовних пакетів периферійний пристрій не отримує ніякої інформації у відповідь.

Такий пакет складається з 31 байта, і якщо цього не достатньо, то периферійний пристрій може обробляти запит певного виду, у відповідь на який буде відправляти інший пакет з 31 байта. Такий вид взаємодії зображений на рисунку, де вгорі периферійний пристрій, який надсилає широкомовні пакети, а внизу запит додаткового пакета від центрального пристрою. За цим запитом може слідувати запит на встановлення з'єднання.

На фізичному рівні BLE використовує частоту 2.4 ГГц ISM діапазону. Для з'єднання різних пристроїв доступний діапазон частот розділений на 40 каналів від 2,4000 ГГц до 2,4865 ГГц, з цих каналів 37 використовуються для передачі даних і останні три канали використовуються для службових цілей - встановлення з'єднання і розсилки широкомовних пакетів.



В якості адреси використовується 48-бітний номер, унікально ідентифікує пристрій, він може бути, як заданий на заводі, так і запрограмований або розрахований в ході експлуатації пристрою. Кожен пристрій може сформувати список адрес, з якими буде працювати, при цьому пакети від всіх інших пристроїв будуть проігноровані.



Канали фізичного рівня	
37	2402 MHz
0	2404 MHz
1	2406 MHz
2	2408 MHz
3	2410 MHz
4	2412 MHz
5	2414 MHz
6	2416 MHz
7	2418 MHz
8	2420 MHz
9	2422 MHz
10	2424 MHz
38	2426 MHz
11	2428 MHz
12	2430 MHz
13	2432 MHz
14	2434 MHz
15	2436 MHz
16	2438 MHz
17	2440 MHz
18	2442 MHz
19	2444 MHz
20	2446 MHz
21	2448 MHz
22	2450 MHz
23	2452 MHz
24	2454 MHz
25	2456 MHz
26	2458 MHz
27	2460 MHz
28	2462 MHz
29	2464 MHz
30	2466 MHz
31	2468 MHz
32	2470 MHz
33	2472 MHz
34	2474 MHz
35	2476 MHz
36	2478 MHz
39	2480 MHz

Частотний діапазон

Пакети з даними  
Широкомовні пакети

## Програмна реалізація інформаційної системи

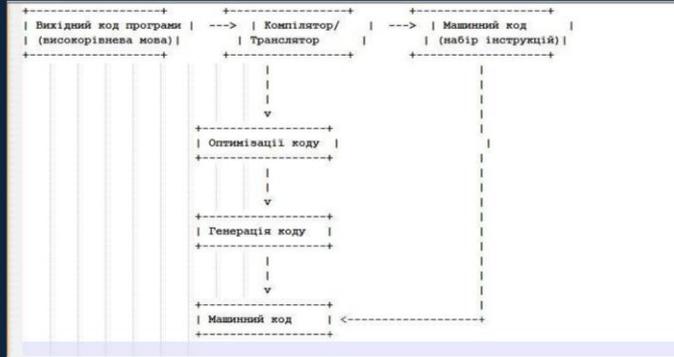
Для реалізації поставленої задачі були вибрані наступні інструменти:

- середовище розробки MPLAB X IDE;
- програматор PICkit3.

Це безкоштовне інтегроване середовище розробки, створене для розробки вбудованих додатків на мікроконтролерах PIC та dsPIC від компанії Microchip. Це середовище розроблене на платформі NetBeans. MPLAB та MPLAB X підтримують управління проектами, редагування коду, налагодження та програмування мікроконтролерів: 8-бітових PIC та AVR від Microchip, 16-бітових PIC24 та dsPIC, а також 32-бітових SAM (ARM) та PIC32 (MIPS).



Для розробки вбудованих систем використовуються «крос-асемблер» або «крос-компілятор», які працюють на комп'ютері, але генерують код для виконання на іншому мікропроцесорі або мікроконтролері.



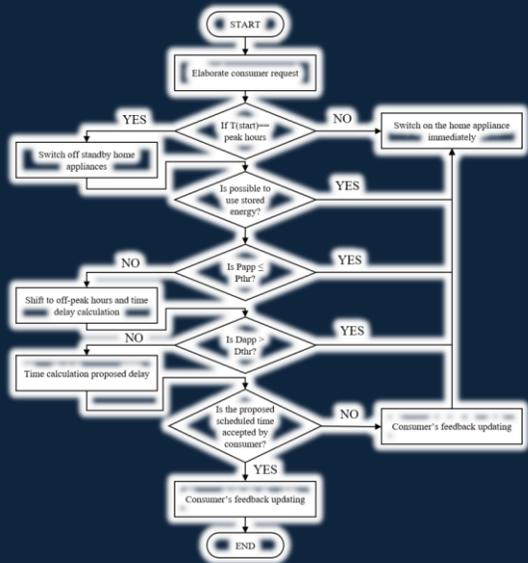
Приклад компіляції



Даний відладчик/програмер PICKIT - це простий, недорогий відладчик в ланцюзі, який керується ПК, що працює на платформі Windows з середовищем розробки MPLAB.

Розбір блок-схеми алгоритму управління енергією у розумному будинку:

- Початкова точка визначається моментом часу, коли споживач вмикає прилад (Tstart).
- У цей момент надсилається відповідний сигнал до ЕРС, щоб повідомити про операцію.
- На наступному кроці ЕРС перевіряє, чи не перевищує час запуску приладу (Tstart) години пікового споживання. Якщо ні - дозволяє приладу запускатися негайно. Якщо ж Tstart знаходиться в пікові години, алгоритм переходить до наступного кроку. Враховуючи, що навіть пристрої в режимі очікування споживають енергію, ЕРС перевіряє всі такі пристрої та вмикає їх.
- Далі ЕРС зв'язується з локальною системою зберігання енергії, щоб дізнатися про доступну або накопичену енергію. Якщо в системі зберігання є достатньо енергії, пристрій запускається негайно без затримки.
- У цьому випадку робота приладу зміщується з пікової години на не пікову. В результаті затримка (Darr), що дорівнює різниці між запланованим часом, запропонованим EMU, і часом початку запиту, впроваджується в цикл роботи приладу завдяки техніці перемикання навантаження.
- На наступному кроці споживач може прийняти або відхилити графік, запропонований EMU.



Реалізація користувацького інтерфейсу передбачає конфігурацію системи у вигляді «мапи», яка представляє собою структуру «ключ-значення». Ця конфігурація кешується у Redis і зберігається протягом роботи додатку, але видаляється при перезавантаженні системи.

Список відповідних конфігурацій:

- AUTO\_MODE: - це конфігурація, яка визначає режим роботи при якому система буде керувати пристроями автоматично при значенні true та мануально при значенні false відповідно;
- POWER: - це конфігурація максимально допустимої потужності BLE пристрою у Вт;
- TEMPERATURE: - це конфігурація максимально допустимої температури BLE пристрою у градусах Цельсія. При перевищенні заданого бар'єру, пристрій автоматично вимикається.

Конфігурування пристрою здійснюється за допомогою користувацького інтерфейсу, розробленого з використанням JavaScript бібліотеки Vue.js.

Також реалізовано графічний інтерфейс для відстеження метрик системи протягом певного проміжку часу. Цей інтерфейс дозволяє розрахувати ефективність роботи конкретного пристрою мережі та його пікові навантаження.

