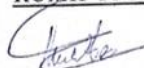


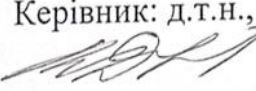
Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра автоматизації та інтелектуальних інформаційних технологій

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА
на тему:

Комплексна система децентралізованого керування опаленням


Виконав: студент 2-го курсу, групи
ЗАКІТ-22м
спеціальності 151 – Автоматизація та
комп'ютерно-інтегровані технології

 Артур ПИЛЯВЕЦЬ
Ім'я ПРИЗВИЩЕ

Керівник: д.т.н., професор кафедри КСУ
 Володимир ДУБОВОЙ
Ім'я ПРИЗВИЩЕ

« 4 » листопада 2023 р.

Опонент: к.т.н., доцент кафедри
комп'ютерних наук

 Руслан БЕЛЗЕЦЬКИЙ
Ім'я ПРИЗВИЩЕ

« 8 » листопада 2023 р.

Допущено до захисту

Зав. кафедри АІТ ВНТУ

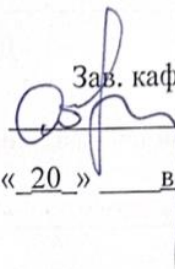
д.т.н., професор

 Олег БІСКАЛО
Ім'я ПРИЗВИЩЕ

« 10 » листопада 2023 р.

Вінниця ВНТУ – 2023 рік

ВІННИЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра автоматизації та інтелектуальних інформаційних технологій
Рівень вищої освіти II-й (магістерський)
Галузь знань – 15 Автоматизація та приладобудування
Спеціальність 151 Автоматизація та комп'ютерно-інтегровані технології
Освітня програма Інформаційні системи і інтернет речей


ЗАТВЕРДЖУЮ
Зав. кафедри АІТ, проф.,
Олег БІСІКАЛО
Ім'я ПРІЗВИЩЕ
« 20 » вересня 2023 р.

ЗАВДАННЯ НА МАГІСТЕРСЬКУ ДИПЛОМНУ РОБОТУ СТУДЕНТУ

Пилявцю Артуру Ігоровичу

(прізвище, ім'я, по батькові)

1. **Тема роботи:** Комплексна система децентралізованого керування опаленням,
керівник роботи Володимир ДУБОВОЙ – д.т.н., професор, професор кафедри КСУ,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затвержені наказом вищого навчального закладу від “18” вересня 2023 року № 247

2. Термін подання студентом роботи 5 грудня 2023 року

3. Вихідні дані до роботи: загальна інформація про розподіленні децентралізовані системи опалювання їх будова, алгоритми, аналоги, програмне забезпечення для мікроконтролеру та апаратна частина, мобільний застосунок, діапазон керування температурою 15-30 °С, керування нагрівачем потужністю до 2200W.

Джерела розробки:

1. В. М. Дубовой, М. С. Юхимчук. Децентралізоване координаційне керування розподіленими кібер-фізичними системами з неперервними об'єктами: монографія. Вінниця : ВНТУ, 2022. 230 с.
2. Yukhymchuk M., Dubovoi V., Kovtun V. «Decentralized Coordination of Temperature Control in Multiarea Premises». Complexity. 2022. ID 2588364, 18 pages. [3]
3. F. Bauman, H. Zhang, E. Arens et al., Advanced Integrated Systems Technology Development: Personal Comfort Systems and Radiant Slab Systems, UC Berkeley: Center for the Built Environment, Berkeley, CA, USA, 2015.

4. Зміст текстової частини:

1) аналіз предметної області;

2) аналіз, розробка та тестування апаратної частини;

3) аналіз технологій, опис, розробка та тестування програмного забезпечення для

мікроконтролеру;

4) аналіз, розробка та тестування функціональності мобільного застосунку.

5. Перелік ілюстративного матеріалу (з точним зазначенням обов'язкових креслень): актуальність, мета та практична цінність, розділ “Дослідження”, загальний аналіз систем опалення, розділ “Моделювання та розробка апаратної частини”, розробка принципової схеми, розробка та тестування зібраних варіантів, розділ “Програмне забезпечення”, принцип роботи, використання хвильового методу, розробка та інтеграція мобільного застосунку, висновки.

6. Консультанти розділів роботи

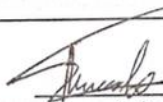
Розділ	Прізвище, ініціали	Підпис, дата	
		Завдання видав	Виконання прийняв
1	Дубовой В. М. розділ 1	21.09.2023	
2	Дубовой В. М. розділ 2	01.10.2023	
3	Дубовой В. М. розділ 3	10.10.2023	
4	Дубовой В. М. розділ 4	25.10.2023	

7. Дата видачі завдання "21" вересень 2023 року


КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів магістерської дипломної роботи	Строк виконання етапів роботи	При- мітка
1	Аналіз сучасних засобів для створення бездротової системи збору даних від сенсорів розумного будинку.	26.09.2023 р.	Вик.
2	Розробка функціональної та структурної схеми	01.10.2023 р.	Вик.
3	Розробка апаратної частини	10.10.2023 р.	Вик.
4	Розробка програмного забезпечення	15.11.2023 р.	Вик.
5	Тестування апаратної частини	20.11.2023 р.	Вик.
6	Оформлення пояснювальної записки і графічного матеріалу	01.12.2023 р.	Вик.
8	Попередній захист	07.12.2023 р.	Вик.
9	Остаточний захист	15.12.2023 р.	

Студент


 (підпис)
Артур ПИЛЯВЕЦЬ

Керівник роботи


 (підпис)
Володимир ДУБОВОЙ

АНОТАЦІЯ

УДК 681.513.54

Пилявець А. І. Комплексна система децентралізованого керування опаленням. Магістерська кваліфікаційна робота зі спеціальності 151 – Автоматизація та комп'ютерно-інтегровані технології. Інтелектуальні комп'ютерні системи. Вінниця: ВНТУ, 2023. 115 с.

На укр. мові. Бібліогр.: 88 назв; рис.: 46; табл. 1.

У магістерській кваліфікаційній роботі проведено дослідження в галузі комплексної системи опалення з акцентом на аналіз електронних приладів для управління. Вивчено різні підходи, алгоритми та пристрої для опалювальних систем, включаючи їхню робочу логіку та можливі застосування. Проведено порівняльний аналіз цих підходів.

Розроблена апаратна частина на основі контролера ESP32, яка взаємодіє з комплексною системою опалення та використовується для регулювання температури повітря. Для забезпечення взаємодії з апаратною частиною розроблено програмний компонент. Крім того, розроблено мобільний застосунок для пристроїв iOS, який використовується для зручного керування комплексною системою опалення.

Ключові слова: комплексна система опалення, ESP32, BLE, MESH, MSH, Swift, мобільний застосунок, iOS, хвильовий алгоритм.

ABSTRACT

Pyliavets A. I. Complex system of decentralized heating control. Master's thesis on specialty 151 - Automation and computer-integrated technologies. Intelligent computer systems. Vinnytsia: VNTU, 2023. 115 p.

In Ukrainian speech Bibliography: 88 titles; fig.: 46; table 1.

In the master's qualification work, a study was conducted in the field of a complex heating system with an emphasis on the analysis of electronic devices for control. Different approaches, algorithms and devices for heating systems are studied, including their working logic and possible applications. A comparative analysis of these approaches was carried out.

A hardware part based on the ESP32 controller has been developed, which interacts with the integrated heating system and is used to regulate the air temperature. A software component has been developed to ensure interaction with the hardware part. In addition, a mobile application for iOS devices has been developed, which is used for convenient control of the complex heating system.

Keywords: integrated heating system, ESP32, BLE, MESH, MSH, Swift, mobile application, iOS, Lee algorithm.

ЗМІСТ

ВСТУП.....	4
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	7
1.1 Причини для керування температури приміщення	7
1.2. Огляд сучасних систем опалення	8
1.2.1 Повітряні системи опалення	8
1.2.2 Водяні системи опалення	10
1.2.3 Теплові насоси.....	11
1.2.4 Електричні системи опалення.....	11
1.3 Системи керування опаленням	13
1.3.1 Загальний огляд систем керування опалення.....	13
1.3.2 Огляд конкурентів.....	15
1.4 Комплексна система керування опаленням.....	17
1.4.1 Комплексна система.....	19
1.5 Висновок до розділу 1.....	19
2 АНАЛІЗ, РОЗРОБКА ТА ТЕСТУВАННЯ АПАРАТНОЇ ЧАСТИНИ.....	21
2.1 Вибір компонентів апаратної частини	21
2.2 Розробка схеми та збір апаратної частини.....	25
2.3 Огляд та тестування зібраних варіантів.....	29
2.4 Висновок до розділу 2.....	33
3 АНАЛІЗ ТЕХНОЛОГІЙ, ОПИС, РОЗРОБКА ТА ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ МІКРОКОНТРОЛЕРУ	34
3.1 Розробка MESH модулю.....	36
3.2 Розробка BLE модулю	38
3.3 Розробка модулю контролю/підтримання заданої температури	40
3.4 Опис хвильового алгоритму для керування температурою у комплексній системі нагрівачів.....	43
3.5 Висновок до розділу 3.....	49
4 АНАЛІЗ, РОЗРОБКА ТА ТЕСТУВАННЯ ФУНКЦІОНАЛЬНОСТІ МОБІЛЬНОГО ЗАСТОСУНКУ	51
4.1 Аналіз функціональних частин застосунку	51

4.2 Підтримка технології BLE.....	51
4.3 Вибір та аналіз інструментів розробки – BLE – огляд ядра Bluetooth.....	54
4.3.1 Центральні, периферійні пристрої та їхня роль у зв'язку Bluetooth	55
4.4 Визначення середовища та мови програмування	61
4.5 Розробка мобільного застосунку	64
4.5.1 Визначення інтерфейсу користувача.....	65
4.5.2 Розробка інтерфейсу користувача	66
4.5.2 Розробка функцій мобільного застосунку	71
4.5 Визначення тестування мобільного застосунку.....	75
4.5.1 Тестування функціональності.....	76
4.5.2 Тестування продуктивності.....	77
4.6 Висновок до розділу 4.....	77
ВИСНОВКИ	79
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	81
Додаток А – Технічне завдання (обов'язковий)	91
Додаток Б – Ілюстративна частина (обов'язковий).....	94
Додаток В – Лістинг програмної частини основної функції налаштувань (обов'язковий).....	107
Додаток Г – Лістинг програмної частини основної циклової функції (обов'язковий).....	108
Додаток Ґ – Лістинг програмної частини інтерфейсу користувача у мобільному застосунку (обов'язковий)	110
Додаток Д – Лістинг програмної частини керування BLE у мобільному застосунку (обов'язковий)	111
Додаток Е – Протокол перевірки кваліфікаційної роботи (обов'язковий)	112

ВСТУП

Актуальність. В умовах сучасної ситуації, коли Україна переживає війну з Росією, особливо важливо звернути увагу на питання енергоощадності. Необхідність заощадження паливно-енергетичних ресурсів стає надзвичайно актуальною, особливо в умовах постійних обстрілів, які призводять до проблем з електропостачанням, особливо взимку.

Однією з ключових проблем для українського промислово-житлового комплексу є питання енергоощадності. У сучасних умовах обмеженості паливно-енергетичних ресурсів, зокрема через воєнний конфлікт, нагальною стає необхідність раціонального їх використання. Значну кількість енергії споживають системи опалення великих промислових приміщень та багатоквартирних житлових будинків, які, через їхні великі об'єми та значну висоту, є складними у плані теплозабезпечення.

Зазвичай такі приміщення опалюються традиційними методами, використовуючи повітряні або водяні опалювальні системи. Проте ці методи не є ефективними, оскільки вони призводять до надмірних витрат енергоносіїв через інерційність, високий градієнт температури по висоті приміщення та нагромадження теплого повітря вище робочої зони. [2]

У зв'язку з цим, для теплозабезпечення великих об'ємів приміщень стає доречним пошук альтернативних рішень. Найбільш ефективним і економічним способом опалення високих приміщень є використання систем з електричними обігрівачами. Вони мають перевагу у тому, що нагрівають лише ті зони, де потрібне обігрівання, та дозволяють цілеспрямовано опалювати окремі зони приміщення або робочі місця. При цьому такий вид обігрівачів швидко досягає необхідного рівня потужності і мають меншу інерційність у порівнянні з традиційними системами опалення.

Предмет дослідження: комплексна система терморегуляції для "розумного" керування температурою в приміщеннях, зокрема, система опалення.

Об'єкт дослідження: методи, засоби, і апаратна частина для створення мережі мікроконтролерів, призначених для автоматичного контролю параметрів температури у приміщеннях.

Метою магістерської кваліфікаційної роботи є підвищення надійності, зручності використання, зменшення навантаження на мережу та підвищення ефективності на основі опалювальних приладів – мобільних нагрівачів.

Завданнями роботи є:

- аналіз предметної області;
- розробка та тестування апаратної частини;
- аналіз технологій, опис, розробка та тестування програмного забезпечення для мікроконтролеру;
- розробка мобільного застосунку для управління контролером.

Новизна роботи полягає у створенні нової системи терморегуляції для "розумного" керування температурою в приміщеннях, зокрема, в розробці комплексної системи, яка використовує мережу мікроконтролерів для оптимального управління опалюванням великих об'ємів приміщень. Використання електричних обігрівачів і їхнє керування через мобільний додаток також є інноваційним підходом.

Практична цінність. Дослідження має практичну цінність в умовах сучасної ситуації в Україні, де енергоощадні рішення для опалення стають особливо важливими через обмеженість паливно-енергетичних ресурсів, спричинену військовим конфліктом. Розроблена система може сприяти зменшенню енерговитрат і оптимізації опалювання великих приміщень.

Апробація. Представлені в роботі результати апробовані в результаті участі в "LIII Науково-технічна конференція факультету інтелектуальних інформаційних технологій та автоматизації (2024)" та XIII Міжнародна

науково-практична конференція «Modern ways of development of science and the latest theories».

Публікації:

- В. М. Дубовой, А. І. Пилявець, “Комплексна система децентралізованого керування опаленням”, “LIII Науково-технічна конференція факультету інтелектуальних інформаційних технологій та автоматизації (2024)”, Вінниця : ВНТУ, 2023. [1]
- Пилявець Артур Ігорович, Дубовой Володимир Михайлович “КОМПЛЕКСНА СИСТЕМА ДЕЦЕНТРАЛІЗОВАНОГО КЕРУВАННЯ ОПАЛЕННЯМ”, “XIII Міжнародна науково-практична конференція «Modern ways of development of science and the latest theories» Мадрид, Іспанія”, 2023. [88]

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Причини для керування температури приміщення

Важливим аспектом в сфері виробництва є підвищення ефективності та використання енергозберігаючих технологій. Для досягнення високої продуктивності та безперебійної роботи системи необхідно розробляти системи, що базуються на потужних електричних нагрівачах. Автоматизоване регулювання цих систем дозволить зменшити економічні витрати і підвищити коефіцієнт ефективності використання теплоносіїв. Встановлення економічних режимів роботи електричних нагрівачів може бути забезпечено лише завдяки автоматичному керуванню їх основними параметрами. [50]

Аналіз поточного стану існуючих систем опалення свідчить про активний розвиток високоефективних, енергозберігаючих та економічних рішень. В Україні та за кордоном проведено значну кількість досліджень щодо використання електричних систем опалення [49].

Можливість спрощення процесу синтезу регуляторів та їх практичної реалізації можлива завдяки математичному опису регуляторів та моделей об'єктів у дискретному вигляді. Сучасні системи автоматичного керування параметрами електричних обігрівачів забезпечують оптимальне регулювання з високою швидкістю реакції регуляторів [2].

У випадку опалення великих приміщень різного призначення найчастіше застосовують традиційні системи, такі як водяні, парові та повітряні. Зазвичай, великий об'єм приміщень не має належної теплоізоляції, що призводить до значних тепловтрат. Розробка ефективної комплексної системи опалення для таких приміщень є складним завданням і економічно неефективною задачею, особливо враховуючи, що потрібно опалювати лише робочу зону [29].

На ринку датчиків для контролю оточуючого середовища набагато менше пропозицій, особливо якщо розглядати їх як окремий компонент системи.

Створення повноцінної системи "розумних" нагрівачів повинно бути легко інтегрованою в інші системи або працювати самостійно, особливо для невеликих приміщень [81].

1.2. Огляд сучасних систем опалення

Опалення – це процес передачі тепла від нагрівального елемента до навколишнього середовища. Системи опалення призначені для забезпечення комфортної температури в приміщеннях. [9]

Сучасні системи опалення можна класифікувати за такими ознаками:

- За типом теплоносія:
 - повітряні;
 - водяні;
 - електричні;
 - теплові насоси.
- За способом передачі тепла:
 - конвекційні;
 - комбіновані.
- За способом управління:
 - ручні;
 - механічні;
 - автоматизовані.

1.2.1 Повітряні системи опалення

Повітряні системи опалення використовують тепло повітря для нагрівання приміщення. Вони можуть бути централізованими або автономними.

Центральні повітряні системи опалення використовують тепло, вироблене в котлі або котельні, для нагрівання повітря в теплогенераторі. Нагріте повітря подається в приміщення по повітроводах.

Автономні повітряні системи опалення використовують тепло, вироблене електричним або газовим обігрівачем. Нагріте повітря подається в приміщення по повітроводах або розподіляється за допомогою вентиляторів.

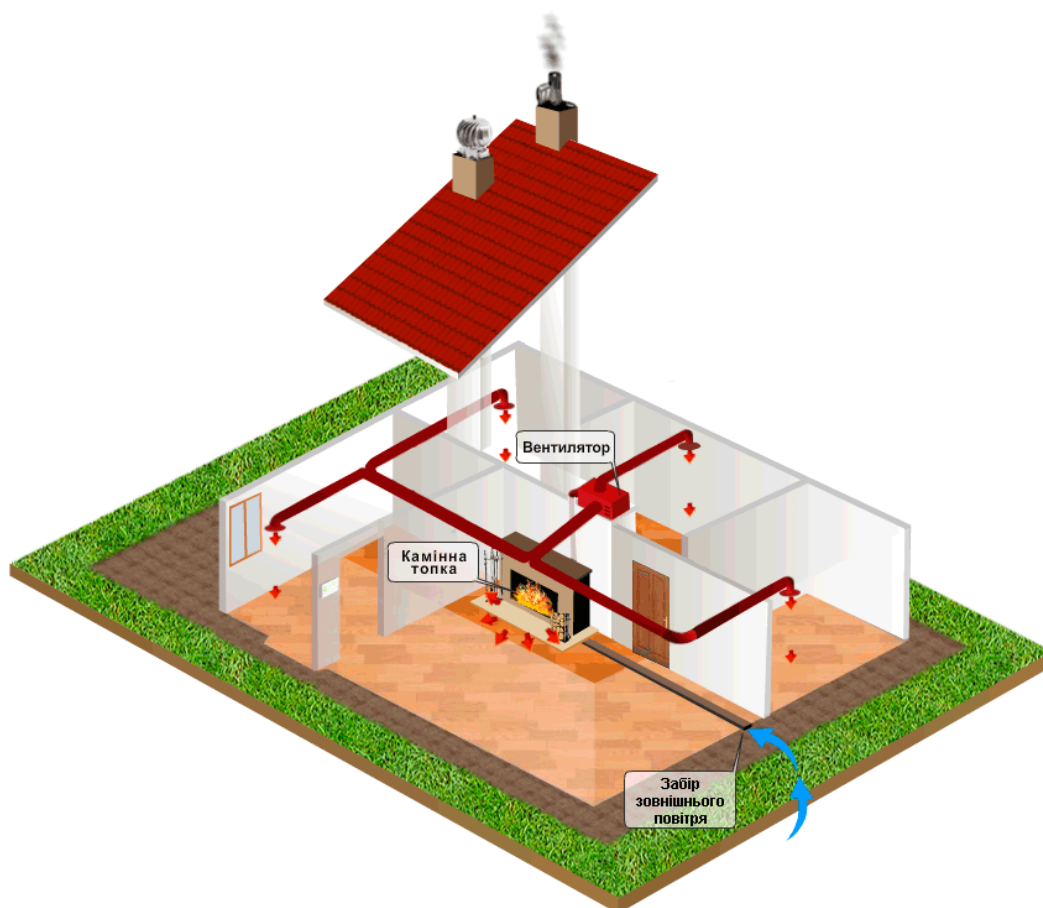


Рисунок 1.1 – Схема роботи повітряної системи опалення [62]

Переваги повітряних систем опалення:

- простота монтажу та обслуговування;
- економічність;
- гнучкість у регулюванні температури.

Недоліки повітряних систем опалення:

- нерівномірний розподіл тепла в приміщенні;
- шум;
- сухе повітря.

1.2.2 Водяні системи опалення

Водяні системи опалення використовують тепло води для нагрівання приміщення. Вони можуть бути централізованими або автономними.

Центральні водяні системи опалення використовують тепло, вироблене в котлі або котельні, для нагрівання води в теплогенераторі. Нагріта вода подається в приміщення по трубах.

Автономні водяні системи опалення використовують тепло, вироблене електричним або газовим обігрівачем. Нагріта вода подається в приміщення по трубах.

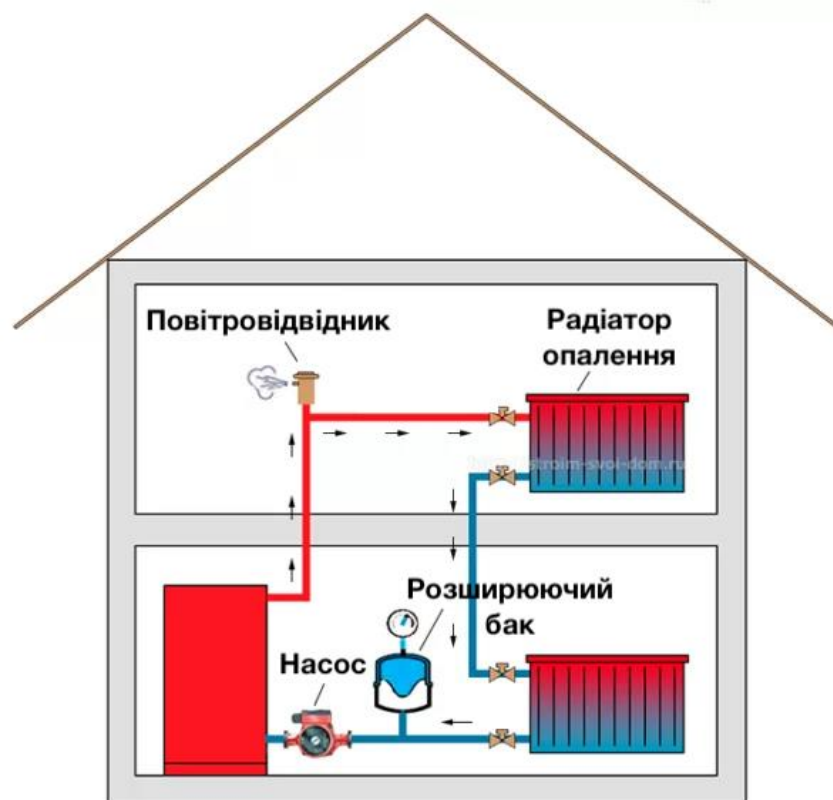


Рисунок 1.2 – Схема роботи водяної системи опалення [64]

Переваги водяних систем опалення:

- рівномірний розподіл тепла в приміщенні;
- безшумна робота;
- можливість використання різних джерел тепла.

Недоліки водяних систем опалення:

- складність монтажу та обслуговування;
- вартість обладнання.

1.2.3 Теплові насоси

Теплові насоси використовують тепло з навколишнього середовища для нагрівання приміщення. Вони можуть використовувати тепло повітря, землі або води. [65]

Теплові насоси можуть бути централізованими або автономними.

Переваги теплових насосів:

- високий коефіцієнт корисної дії (ККД);
- економічність;
- екологічність.

Недоліки теплових насосів:

- висока вартість обладнання;
- складність монтажу.

1.2.4 Електричні системи опалення

Електричні системи опалення використовують електричний струм для нагрівання приміщення. Вони можуть бути централізованими або автономними.

Централізовані електричні системи опалення використовують тепло, вироблене в електростанції, для нагрівання повітря або води в теплогенераторі. Нагріте повітря або вода подається в приміщення по трубах або повітроводах.

Автономні електричні системи опалення використовують електричні обігрівачі для нагрівання повітря або води. Нагріте повітря або вода подається в приміщення по трубах або повітроводах.

Дана магістерська робота описує роботу з електричними мобільними системами опалення та побудову розумних контролерів для керування системою децентралізованого опалення.

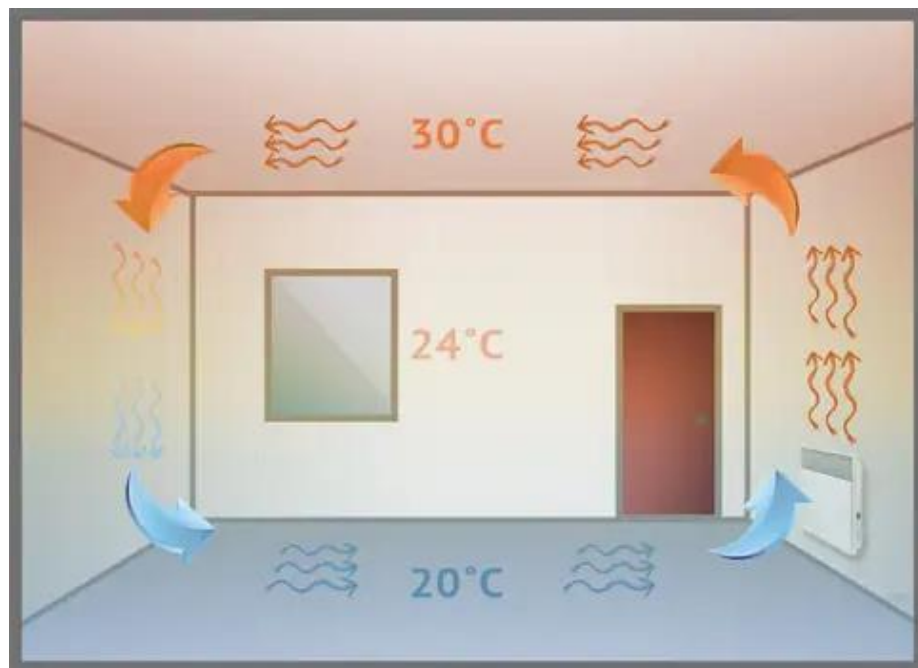


Рисунок 1.3 – Схема роботи конвекторної системи опалення [63]

Перевагами електричних систем опалення є:

- простота монтажу та обслуговування;
- гнучкість у регулюванні температури.

Недоліки електричних систем опалення:

- висока вартість енергії;
- залежність від електропостачання.

Існує декілька різновидів електричного опалення:

- котел із водяною системою опалення;
- конвектори;

- масляні батареї;
- керамічні електронагрівальні панелі;
- інфрачервоні обігрівачі;
- система теплої підлоги.

Свою увагу ми виділили на конвектори через їх простоту у використанні, легкість при зміні розташуванні та відносно великий ККД.

1.3 Системи керування опаленням

Системи керування опаленням в сучасному середовищі відіграють ключову роль у забезпеченні ефективного та економічного використання теплових ресурсів в будівлях. Цей розділ розглядає поняття, застосування та особливості систем керування опаленням, а також їх важливість у контексті дослідження "розумного" керування температурою в приміщеннях.

Загальною метою використання систем керування опаленням є створення ефективної та зручної інфраструктури для підтримки оптимальної температури в будь-який час.

1.3.1 Загальний огляд систем керування опалення

Системи керування опаленням представляють собою комплекс технологій та пристроїв, спрямованих на автоматизацію та оптимізацію процесів теплопостачання. Основною метою цих систем є забезпечення комфортної температури в приміщенні за мінімальних витрат енергії. [46, 47]

Системи керування опаленням визначаються такими характеристиками як:

- Ефективність використання енергії: системи керування опаленням дозволяють оптимізувати роботу опалювальних систем, забезпечуючи ефективне використання енергоресурсів;

- Економія: автоматичне регулювання температури дозволяє уникнути зайвого споживання енергії, що призводить до економії фінансових та фізичних ресурсів;
- Комфорт: системи керування опаленням забезпечують сталу температуру в приміщенні, забезпечуючи комфорт та зручність для користувачів.

Види та варіанти застосування:

1. Термостати: електронні термостати дозволяють регулювати температуру у приміщенні і автоматично вимикати опалення при досягненні заданого рівня;

2. Системи "розумного" керування: використання сучасних технологій та зв'язку з мобільними пристроями для дистанційного керування температурою;

3. Таймери опалення: програмовані таймери дозволяють встановлювати часи роботи опалювальних систем відповідно до графіку активності користувачів.

Більшість сучасних систем керування опаленням роблять акцент на легкості використання, надаючи користувачам інтуїтивно зрозумілі інтерфейси. Це включає в себе доступність основних функцій через один або кілька дотиків, забезпечуючи зручний та швидкий доступ до параметрів системи. Багато з них також підтримують дистанційне керування через мобільні додатки, що робить процес керування температурою ще більш зручним. Користувачі можуть легко налаштувати температурні режими, графіки та інші параметри за допомогою інтерфейсу, що не вимагає спеціальних технічних знань. [45]

Однак, коли йдеться про встановлення, ситуація може бути іншою. Інтеграція "розумних" систем керування опаленням може вимагати деяких технічних навичок та розуміння роботи опалювальних систем. Зазвичай, це включає в себе підключення до електромережі, розташування та програмування термостатів, а також інтеграцію з іншими "розумними" пристроями. Велика увага до деталей і деяка експертиза можуть бути необхідні для ефективної

інсталяції та налагодження систем, забезпечуючи їхню надійну та оптимальну роботу. [37]

1.3.2 Огляд конкурентів

В даному підрозділі розглянемо основних конкурентів у сфері систем керування опаленням, їхні переваги та недоліки. [9]

1. Google Nest Learning Thermostat:

- Плюси:
 - автоматичне навчання термостата за звичками користувача.
 - дистанційне керування через мобільний додаток;
 - енергоефективність за рахунок оптимізації роботи.
- Мінуси:
 - висока вартість;
 - обмежена сумісність із деякими опалювальними системами.

2. Ecobee SmartThermostat:

- Плюси:
 - додаткові датчики для зонавого керування температурою;
 - сумісність із багатьма системами опалення та кондиціонування.
- Мінуси:
 - недоступний до керування за відсутності інтернет зв'язку;
 - специфічний дизайн, який може не всім подобатися.

3. Honeywell Home T9 Smart Thermostat:

- Плюси:
 - автоматичне визначення присутності користувачів для оптимізації роботи системи.

- Мінуси:
 - обмежена інтеграція з деякими "розумними" системами в порівнянні з конкурентами;
 - висока ціна порівняно з іншими термостатами.

4. Tado:

- Плюси:
 - високий рівень енергоефективності;
 - зручне використання;
 - широкий спектр функцій, включаючи:
 - зональне керування;
 - автоматичне навчання;
 - прогноз погоди;
 - моніторинг енергоспоживання.
- Мінуси:
 - висока вартість;
 - необхідність використання шлюзу для зв'язку з Інтернетом.

5. Honeywell Lyric T:

- Плюси:
 - широкий спектр функцій, включаючи:
 - зональне керування;
 - автоматичне навчання;
 - прогноз погоди;
 - підтримка голосового керування.
- Мінуси:
 - необхідність використання шлюзу для зв'язку з Інтернетом;
 - можливі проблеми з сумісністю з деякими опалювальними системами.

6. Netatmo Smart Thermostat:

- Плюси:
 - унікальні функції, такі як:
 - моніторинг якості повітря;
 - керування роботою опалювальних елементів.
- Мінуси:
 - не підтримує зональне керування;
 - неінтегрований з іншими розумними системами.

Загальні тенденції:

- більшість сучасних систем керування пропонують дистанційне керування через мобільні додатки, що забезпечує зручність користування;
- зростаюча популярність "розумного" керування з використанням штучного інтелекту та алгоритмів машинного навчання для оптимізації роботи системи.

1.4 Комплексна система керування опаленням

Даний розділ присвячений загальний розгляду та аналізу комплексної системи децентралізованого керування опаленням, яка пропонує інноваційні підходи до оптимізації комфорту та енергоспоживання в приміщеннях.

Основні переваги та інновації:

1. Простота встановлення:

- система визначається своєю легкістю встановлення, що робить її доступною для широкого кола користувачів. Інтуїтивний процес встановлення дозволяє швидко налаштувати систему без особливих труднощів;

2. Робота контролерів у мережі:

- унікальна риса полягає в роботі контролерів у децентралізованій мережі. Кожен контролер може взаємодіяти безпосередньо з іншими, утворюючи систему, для досягнення збалансованого розподілу тепла в різних зонах. Це забезпечує єдиний тепловий комфорт у всьому приміщенні. [43]

Децентралізоване керування та можливості:

1. Можливість керування з телефону:

- велика перевага системи - можливість дистанційного керування опаленням через мобільний додаток. Користувач може зручно налаштувати температурний режим з будь-якого місця та в будь-який час;

2. Різноманітні варіанти встановлення:

- система дозволяє вибрати оптимальний варіант встановлення, враховуючи архітектурні особливості приміщення. Це може бути централізована або розподілена система в залежності від потреб користувача;

3. Енергоефективність та комфорт:

- реалізація системи дозволяє досягти енергоефективності та максимального комфорту, завдяки інтелектуальному керуванню та адаптації до змін зовнішніх умов.

Загальний огляд цього розділу підкреслює важливість та ефективність комплексних систем децентралізованого керування опаленням в сучасному контексті, де зручність, ефективність та технологічні можливості важливі для користувачів.

Наша комплексна система керування опаленням прагне створити сприятливі умови для користувачів, поєднуючи простоту встановлення, високий рівень взаємодії контролерів у мережі, зручне мобільне керування та гнучкість варіантів встановлення.

1.4.1 Комплексна система

Комплексна система - це система, яка складається з великої кількості взаємопов'язаних та взаємодіючих елементів або компонентів, які спільно працюють для виконання специфічної мети або функції. Вона може включати в себе різноманітні підсистеми, модулі та інші частини, які взаємодіють для досягнення загальної мети чи розв'язання складної задачі.

Комплексна система в контексті даного проекту може бути визначена як інтегрована система, що об'єднує різноманітні елементи та підсистеми для досягнення спільної мети - ефективного керування температурою в багатозональних приміщеннях. Така система включає в себе апаратні та програмні компоненти, які працюють у взаємодії для забезпечення оптимальних умов у кожній зоні.

1.5 Висновок до розділу 1

У даному розділі був проведений аналіз предметної області, що включав розгляд причин для керування температурою приміщення, огляд сучасних систем опалення та систем керування опаленням, а також аналіз конкурентів. На основі цього аналізу були сформульовані основні переваги та інновації комплексної системи децентралізованого керування опаленням, що розробляється в рамках даного проекту.

Основні переваги та інновації системи включають:

- простоту встановлення, що робить її доступною для широкого кола користувачів;
- робота контролерів у децентралізованій мережі, що забезпечує збалансований розподіл тепла в різних зонах;
- можливість дистанційного керування опаленням через мобільний додаток.

- різноманітні варіанти встановлення, що дозволяють вибрати оптимальний варіант для конкретного приміщення;
- енергоефективність та комфорт, що забезпечується інтелектуальним керуванням та адаптацією до змін зовнішніх умов.

Ці переваги та інновації роблять систему децентралізованого керування опаленням привабливим рішенням для користувачів, які прагнуть до комфорту, ефективності та технологічних можливостей.

2 АНАЛІЗ, РОЗРОБКА ТА ТЕСТУВАННЯ АПАРАТНОЇ ЧАСТИНИ

Аналіз, розробка та тестування апаратної частини у цьому контексті означає дослідження, проектування та перевірку фізичних компонентів, необхідних для ефективної роботи системи. Цей етап включає в себе вивчення потреб системи, створення та вдосконалення апаратних елементів, а також виконання ретельних тестів, щоб забезпечити їхню надійність та оптимальну функціональність.

2.1 Вибір компонентів апаратної частини

Перед мікроконтролером буде поставлено на виконання наступне:

- зв'язок з пристроєм вводу – смартфоном через BLE;
- створення мережі MESH між контролерами для обміну даних по температурі та заданих параметрах температури до контролю;
- заміри температури зовнішнім датчиком;
- живлення елементів керування від 5В;
- підтримка заданої температури за допомогою реле;
- робота хвильового алгоритму для адаптації/створення комфортних умов з меншим навантаженням на мережу.

Для розробки системи керування опаленням було вибрано проектування пристрою з використанням плати розробника ESP32-WROOM-32D [28]. Використання плати Arduino має ряд переваг, таких як простота підключення різноманітних датчиків, світлодіодів та приводів. Також важливою є можливість одночасного зчитування та обробки даних вбудованим двоядерним процесором ESP32-D0WDQ6. Кожне ядро функціонує незалежно, забезпечуючи швидку та масштабовану роботу пристрою.

Наявність бездротового зв'язку і можливість зчитування даних в режимі реального часу робить Arduino перевагою порівняно з іншими системами. З урахуванням економічності та доступності цих пристроїв, використання Arduino є оптимальним вибором.

На рисунку 2.2 та у таблиці 2.3 описано ключові компоненти, інтерфейси та елементи керування плати ESP32.

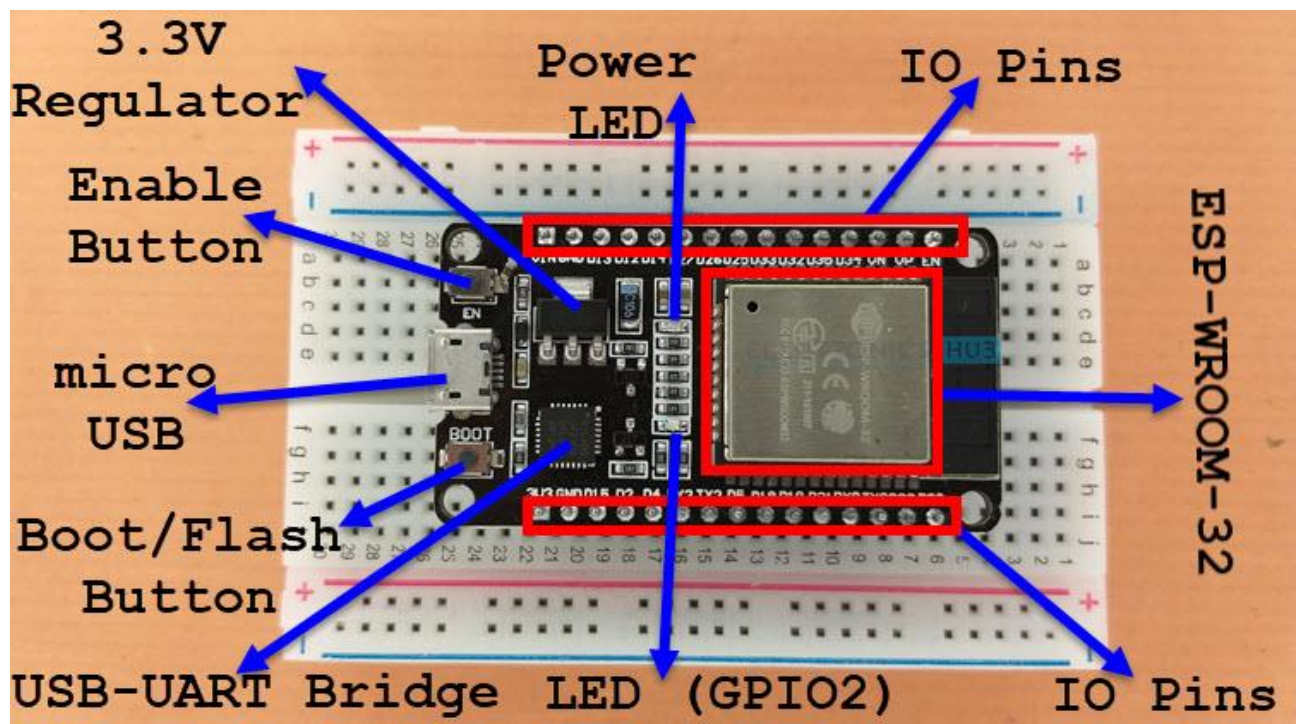


Рисунок 2.1 - Плата розробника ESP32 [30]

Ця плата має в своєму складі ряд важливих модулів:

- Wi-Fi та Bluetooth для бездротового зв'язку.
- Апаратні інтерфейси: SD, UART, SPI, SDIO, I2C, LED PWM, Motor PWM, I2S, I2C, IR.
- Вбудовані датчики, включаючи температурний датчик, що ідеально підходить для контролю температури.

- Широкий діапазон робочих температур (-40 °C ~ 85 °C), що дозволяє працювати пристрою в екстремальних умовах.
- Можливість оновлення програмного забезпечення через UART або OTA (Over-The-Air) оновлення по мережі.
- Підтримка налаштувань користувача, набір інструкцій AT, хмарні сервери та мобільні додатки для Android / iOS.

Таблиця 2.1 – Основні компоненти ESP32 [52]

Основні компоненти	Опис
ESP32-WROOM-32	Модуль з ESP32 в основі.
EN	Кнопка скидання.
Boot	Кнопка завантаження. Утримуючи натиснутою кнопку Boot, а потім натиснувши EN, запускається режим завантаження мікропрограми для завантаження мікропрограми через послідовний порт.
USB-to-UART Bridge	Один чіп моста USB-UART забезпечує швидкість передачі даних до 3 Мбіт/с.
Micro USB Port	Інтерфейс USB. Блок живлення плати, а також інтерфейс зв'язку між комп'ютером і модулем ESP32-WROOM-32.
5V Power On LED	Вмикається при підключенні до плати USB або зовнішнього джерела живлення 5В.
I/O	Більшість контактів на модулі ESP розбиті на контактні роз'єми на платі. Ви можете запрограмувати ESP32 для активації кількох функцій, таких як ШІМ, АЦП, ЦАП, I2C, I2S, SPI тощо.

На сьогоднішній день існують багато різних систем опалення, кожна з яких має свої переваги та недоліки. Від недорогих систем для нових підприємств до складних "розумних" систем для будинків. Однак багато з них вимагають складної установки та обслуговування, що призводить до додаткових витрат.

Система децентралізованого керування опаленням на базі ESP32 та BLE MESH розв'язує ці проблеми, надаючи простоту у встановленні та налаштуванні, а також забезпечуючи зв'язок між різними обігрівачами через BLE MESH, що дозволяє забезпечити оптимальну температуру в кожній зоні.

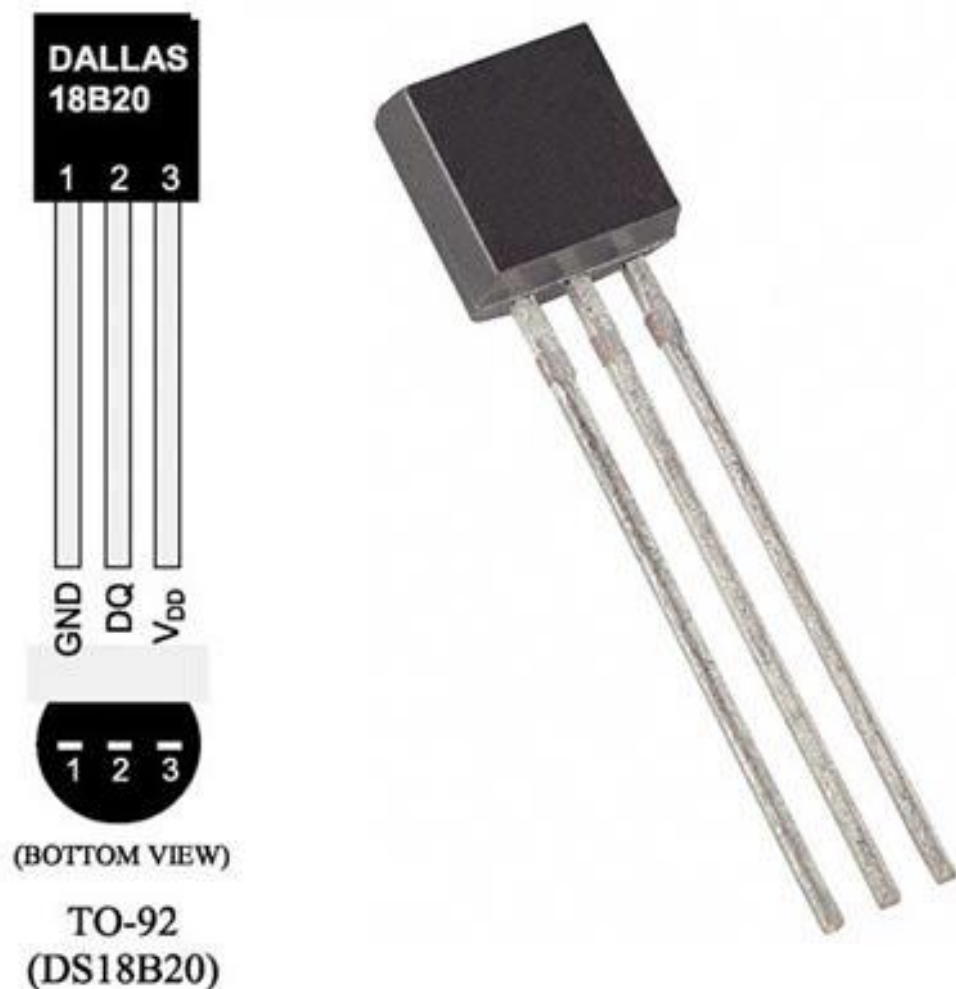


Рисунок 2.2 - Датчик температури DS18B20 [36]

Для надійного та прямого зчитування температури використовується клон датчика Dallas DS18B20, який має аналогічні характеристики схему виводів та видів якого показано на рисунку 2.3. Цей датчик працює в широкому діапазоні температур від -55 до $+125$ °C і надає точні виміри без додаткових перетворень. Його абсолютна похибка становить лише 0.5 °C в діапазоні від -10 до $+85$ °C. Час на зчитування повних 12-бітних значень становить приблизно 750 мілісекунд.

Для підключення цього датчика потрібно використовувати резистор з опором 4.7 кОм.

Окрім цього, внутрішня енергонезалежна пам'ять датчика дозволяє зберігати верхні та нижні межі температурних установок. Даний мікросхема також має вбудований логічний механізм для пріоритетної сигналізації, яка виводить сигнал про виходження температури за межі встановлених порогів. Крім цього, завдяки інтерфейсу One-Wire, існує можливість адресації необмеженої кількості подібних пристроїв на одному провіднику. [41]

2.2 Розробка схеми та збір апаратної частини

В даному розділі розглянуто схему та збірку апаратної частини для системи контролю та управління температурою, яка включає в себе мікроконтролер ESP32, два датчики температури DS18B20, 2 реле для керування нагрівачами та блок живлення AC-DC 5V 1A.

Принципова схема – це графічне зображення електричної або електронної системи, яке використовує стандартизовані символи та лінії для показу зв'язків між різними елементами цієї системи. Це своєрідна "мапа" системи, яка дозволяє швидко та зрозуміло розібратися в її структурі та функціональності. [74]

Основні компоненти принципової схеми включають:

1. Елементи: Стандартизовані символи, які представляють різні види електронних чи електричних компонентів, такі як резистори, конденсатори, транзистори тощо.

2. Лінії з'єднань: Лінії, які показують електричні з'єднання між елементами. Зазвичай використовуються прямі лінії, щоб показати електричні проводи.

3. Написи та Анотації: Текстова інформація, яка допомагає пояснити функції та параметри компонентів або з'єднань.

Принципові схеми використовуються для проектування, розробки та розуміння роботи електричних та електронних систем. Вони можуть бути

використані в різних галузях, включаючи електроніку, електротехніку промисловість тощо.

Друкована схема (PCB) – це плата, на якій розташовані та підключені електронні компоненти для створення конкретної функціональної системи. У нашому випадку, PCB містить компоненти для керування температурою в приміщенні.

На даних схемах наведено наступні елементи:

- C1 – конденсатор з ємністю 0.1 μ ;
- CN1-CN2 – роз'єм підключення навантаження;
- D1 – захисний діод 1N4007;
- D2 – червоний індикаційний світлодіод;
- D3 – захисний діод 1N4007;
- D4 – червоний індикаційний світлодіод;
- M1 – блок живлення AC-DC ізольований, 5V 700mA;
- Q1-Q2 – транзистори BC547 для керування реле;
- R1-R2 – підтягуючі резистори по 4.7k Ω ;
- R3-R6 – резистори обмежування струму на світлодіод по 1k Ω ;
- RE1-RE2 – реле керування нагрівачем;
- U1 – мікроконтролер/плата розробки NODEMCU ESP32 DEV KIT;
- U2-U3 – датчики температури DS18B20.

Станом на зараз – дані блоки було зібрано власноруч з подібних компонентів що було наведено вище у переліку. Дані схеми було розроблено у середовищі easyEDA – онлайн редактор що дозволяє розробити та замовити схеми вироблені на професійному обладнанні. Для подальшої розробки, економії місця та ресурсів з збільшенням надійності буде замовлена міні партія розробленою нами плат. [57]

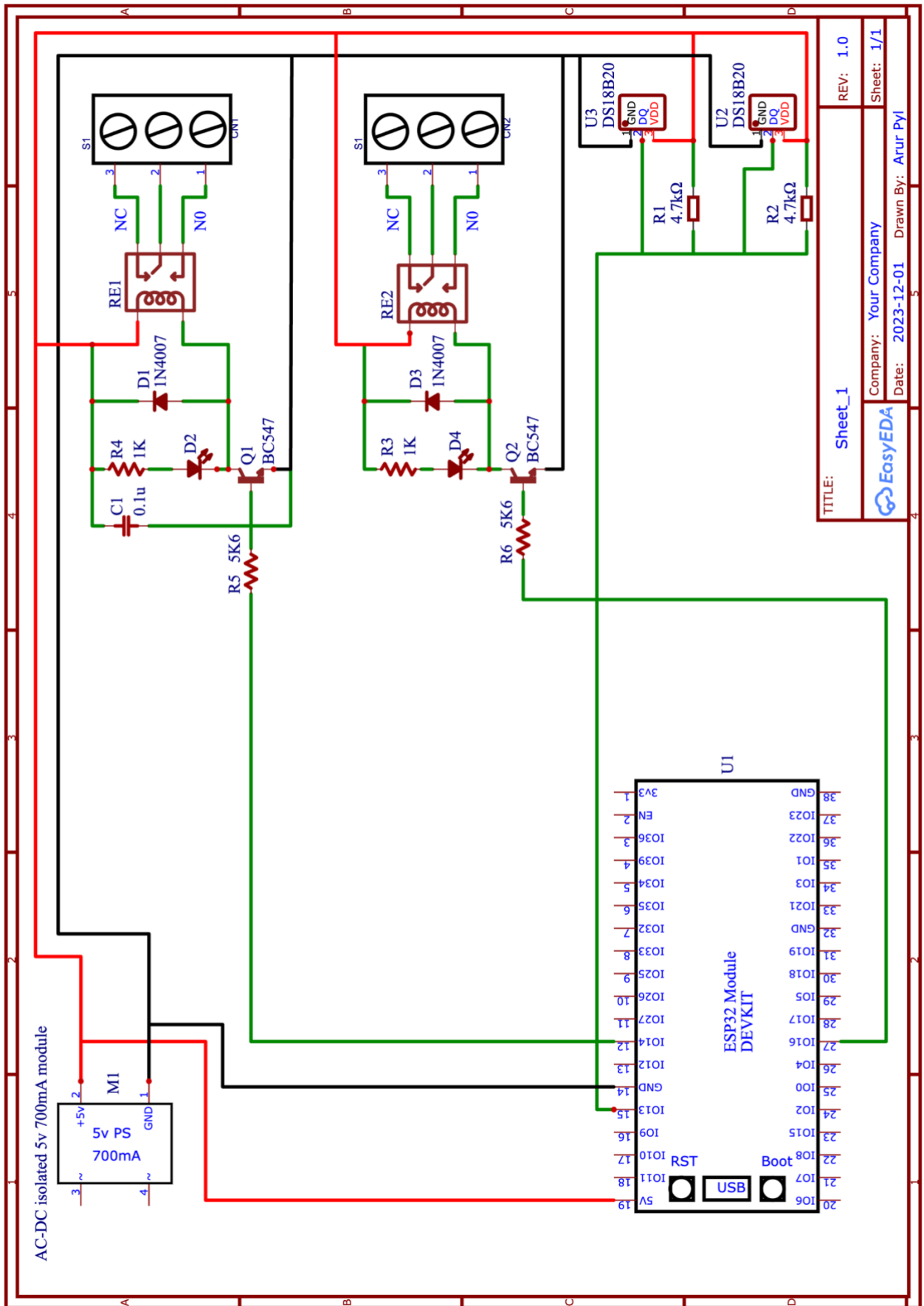


Рисунок 2.3 – Схема принципова апаратної частини

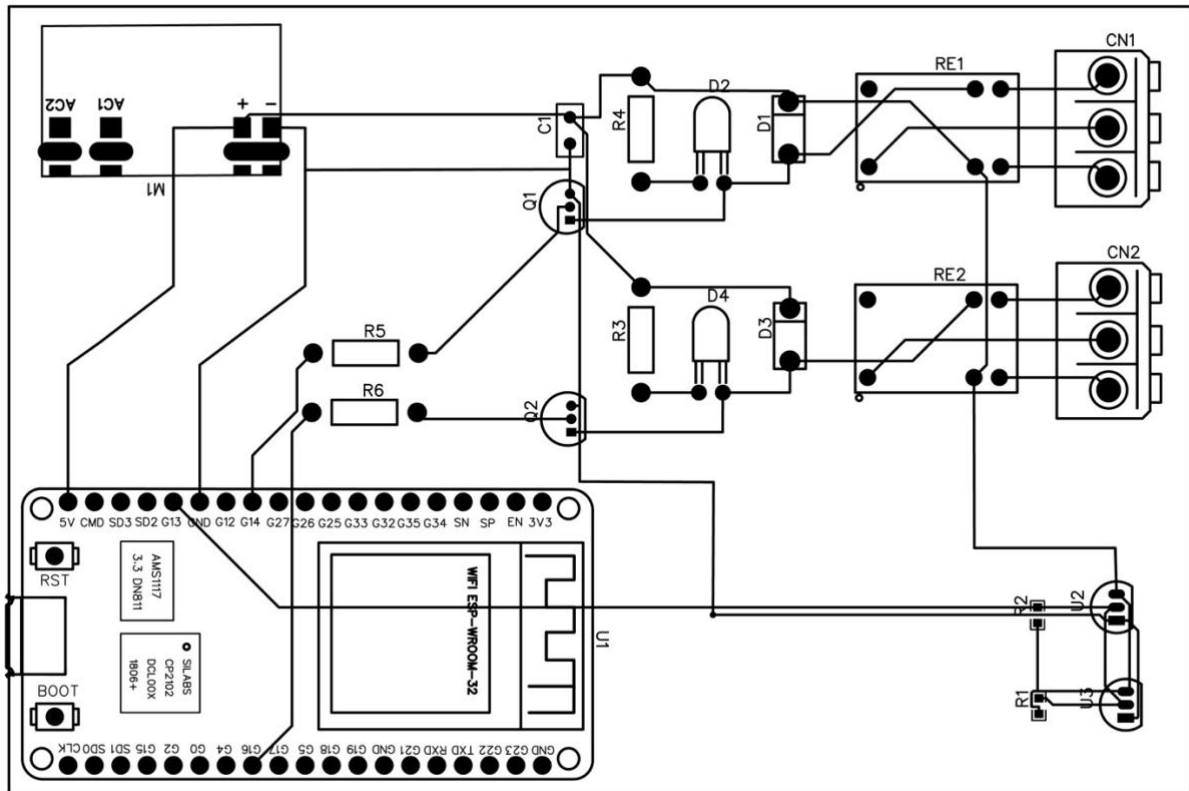


Рисунок 2.4 – Друкована плата апаратної частини з компонентами

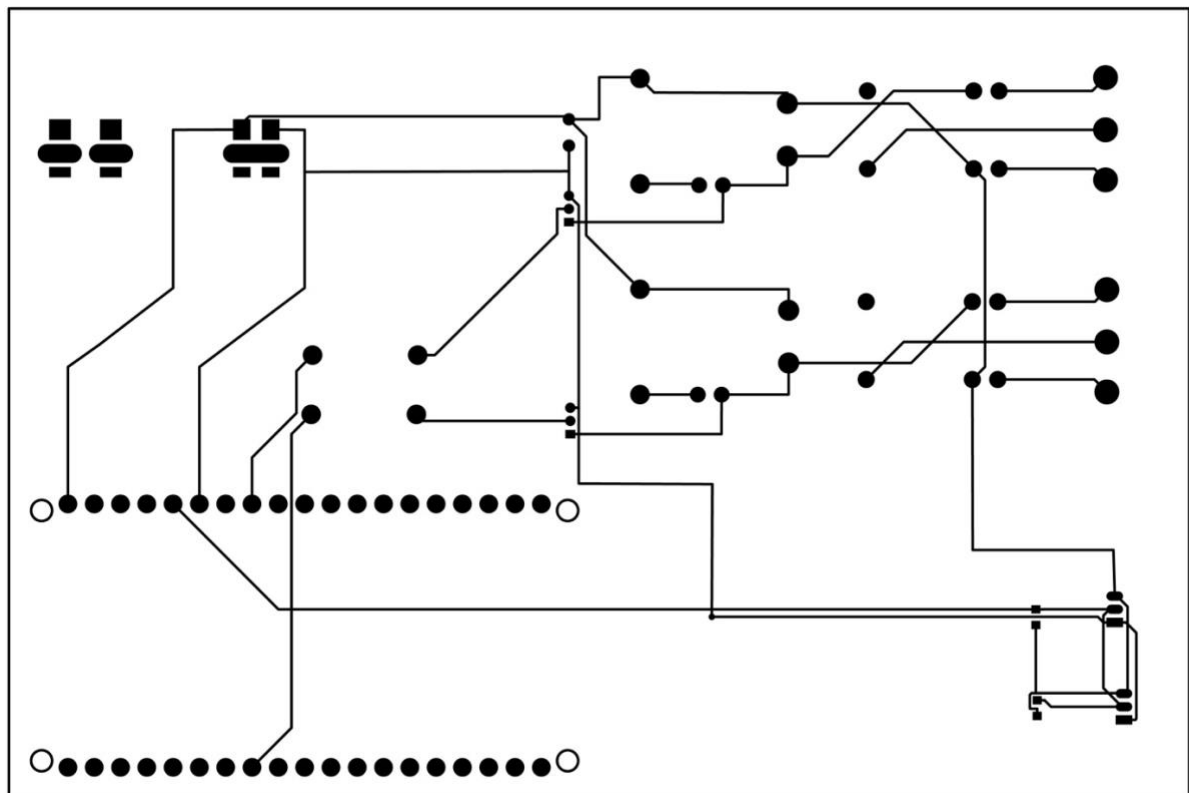


Рисунок 2.5 – Друкована плата апаратної частини

2.3 Огляд та тестування зібраних варіантів

Огляд передбачає детальне розглядання та аналіз ключових елементів комплексної системи децентралізованого опалення. Це включає аналіз модулів контролю, оцінку датчиків температури та інших важливих компонентів. Огляд спрямований на розуміння можливостей та взаємодії всіх елементів системи.

У відношенні даної роботи – тестування означає проведення ряду експериментальних випробувань та перевірок для підтвердження працездатності та надійності системи контролю. Це включає в себе тестування роботи датчиків температури, перевірку взаємодії з реле та іншими компонентами, а також перевірку реакції системи на зміни у вхідних параметрах. Тестування спрямоване на визначення того, наскільки добре система виконує свої функції та як вона взаємодіє з навколишнім середовищем.

На рисунку 2.6 зображено датчик температури який було поміщено у корпус з роз'ємом JASK 3.5 для швидкого встановлення або заміни при виході з ладу. При тестуванні зібраних датчиків – було виявлено різницю показання температури між ними до 0.1 С – що є гарним результатом. Дані датчики будуть зчитувати температуру за мінімальної похибки.



Рисунок 2.6 – датчик температури DS18B20 з роз'ємом

Зображено різні варіанти модулів контролю системи нагрівачів, на рисунку 2.7 зображено модуль що може поєднатись з будь-яким електричним нагрівачем.



Рисунок 2.7 – модуль контролю системи нагрівачів з послідовним підключенням нагрівача – вигляд збоку

На рисунку 2.8 зображено загальний вигляд розроблених та створених модулів для комплексної системи децентралізованого опалення приміщенням. Два модулі без розетки – навісні/інтегровані з заводу модулі у нагрівачі. Наразі вони адаптовані для роботи з елементами Пельт'є у зібраному контейнері з емуляцією розділених кімнат для визначення впливу та тестування алгоритму керування температурою.



Рисунок 2.8 – загальний вигляд на різні види контролерів для комплексної системи нагрівачів

На рисунках 2.9 та 2.10 зображено входи/виходи на навісних/інтегрованих з заводу модулях у нагрівачі серед яких є:

- DC IN – вхід 12V для живлення елементів Пельтьє;
- DC OUT - вхід 12V для живлення елементів Пельтьє з керуванням через подвійне реле з можливістю зміни полярності для нагріву/охолодження;
- T1 та T2 – входи JACK 3.5 для температурних датчиків DS18B20 у корпусі з підповідним роз'ємом;
- 220VAC IN – вхід мережі 220 для заживлення даного модулю – модуль має вбудований перетворювач на 5V DC.



Рисунок 2.9 – навісний модуль контролю системи нагрівачів - порт виходу/зчитування температури з зовнішніх датчиків



Рисунок 2.10 – модуль контролю системи нагрівачів - порти входу

2.4 Висновок до розділу 2

У ході аналізу, розробки та тестування апаратної частини системи децентралізованого керування опаленням на базі ESP32 та BLE MESH було досягнуто наступних результатів:

- вибрано оптимальний набір компонентів для реалізації системи, що забезпечує її надійність, функціональність та економічність;
- розроблено принципову схему та друковані плати для апаратної частини системи;
- зібрано та протестовано прототипи апаратної частини системи.

Результати тестування показали, що система працює коректно та відповідає поставленим вимогам. Зокрема, було виявлено, що датчики температури мають похибку до 0.1 С, а модулі контролю системи нагрівачів можуть працювати з різними типами електричних нагрівачів.

На основі отриманих результатів можна зробити висновок, що апаратна частина системи децентралізованого керування опаленням на базі ESP32 та BLE MESH розроблена та протестована успішно. Система готова до подальшої розробки та впровадження.

Ось деякі перспективи подальшого розвитку системи:

- додавання підтримки додаткових датчиків, наприклад, датчиків вологості, освітленості тощо;
- поліпшення ефективності системи, наприклад, зниження енергоспоживання.

3 АНАЛІЗ ТЕХНОЛОГІЙ, ОПИС, РОЗРОБКА ТА ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ МІКРОКОНТРОЛЕРУ

Мікроконтролер ESP32 відіграє ключову роль у функціональності розроблюваної системи. Основні завдання, які він виконує, включають:

1. Комунікація через BLE з пристроєм керування/перегляду – смартфоном на базі iOS. Мікроконтролер забезпечує надійний зв'язок з смартфоном за допомогою технології BLE (Bluetooth Low Energy). Це дозволяє здійснювати взаємодію та обмін даними між пристроями, зокрема передачу заданої температури та отримання поточних значень.

2. Збереження отриманих даних з смартфона у енергонезалежну пам'ять – мікроконтролер використовує свою внутрішню пам'ять для збереження отриманих від смартфона даних. Це дозволяє зберігати інформацію навіть під час вимкнення живлення.

3. Створення/підтримка мережі MESH подібних контролерів – мікроконтролер допомагає встановлювати та підтримувати мережу MESH, що сприяє взаємодії між контролерами. Це робить систему більш гнучкою, децентралізованою та розширюваною.

4. Обмін інформацією по встановленій температурі/віддаленості – мікроконтролер здійснює обмін інформацією про встановлену температуру та віддаленість між контролерами в мережі MESH. Це важливо для синхронізації роботи системи та управління опаленням у різних зонах.

5. Підтримка заданої температури – мікроконтролер відповідає за підтримку та регулювання температурних параметрів у встановлених межах, забезпечуючи комфорт в приміщенні.

6. Використання хвильового алгоритму для підвищення комфорту та корекції заданої температури відповідно до даних з MESH мережі – мікроконтролер використовує хвильовий алгоритм для адаптації роботи системи опалення з урахуванням отриманих даних від інших контролерів у мережі

MESH. Це сприяє оптимізації енергоспоживання та підвищенню комфорту користувача.

Такий підхід дозволяє забезпечити ефективне та інтелектуальне управління системою опалення за допомогою ESP32.

У даній роботі для розробки програмного забезпечення для мікроконтролеру було використано наступні складові:

1. C++ – це мова програмування загального призначення, яка поєднує в собі можливості низькорівневого програмування та вищих абстракцій. Використовується для розробки різноманітних програм, включаючи системний та вбудований софтвер, десктопні застосунки, веб-розробку та ігри. [83]

2. Arduino – це відкрита платформа для розробки електроніки та простих проєктів IoT (Internet of Things). Основним елементом є мікроконтролер, який можна програмувати за допомогою мови Arduino (на основі C++). Arduino дозволяє легко створювати пристрої, які взаємодіють з реальним світом. [84]

3. ArduinoIDE – це інтегроване середовище розробки для програмування мікроконтролерів Arduino. Воно містить редактор коду, засоби вивантаження програм на плату та інші інструменти, спрощуючи процес розробки для платформи Arduino. [85]

4. ArduinoJSON – це бібліотека для роботи з JSON-даними в мікроконтролерах Arduino. Вона надає можливість легко обробляти та генерувати дані у форматі JSON на пристроях з обмеженими ресурсами. [82]

5. PainlessMesh – це бібліотека для створення бездротових мереж (mesh networks) на базі платформи Arduino. Вона дозволяє створювати взаємопов'язані пристрої, що можуть спільно працювати та обмінюватися даними у мережі. [86]

6. OneWire – це протокол взаємодії для передачі даних між мікроконтролером та пристроями, підключеними за одним дротом. Використовується для взаємодії з цифровими датчиками температури. [41]

7. DallasTemperature – це бібліотека, яка спрощує взаємодію з датчиками температури від Dallas Semiconductor (також відомих як Maxim Integrated). Вона дозволяє зручно отримувати дані про температуру в програмах для платформи Arduino. [42]

8. BLE Arduino Library – це бібліотека, яка надає програмістам простий інтерфейс для взаємодії з технологією Bluetooth Low Energy на мікроконтролерах ESP32 в середовищі Arduino IDE. [39, 40]

Основні Функціональності та Можливості:

- створення BLE-пристроїв – бібліотека дозволяє створювати BLE-пристрої з різноманітними характеристиками та послугами.
- центральні та Периферійні Ролі – ESP32 може виконувати як центральну (Central), так і периферійну (Peripheral) ролі в BLE-мережі.
- обмін Даними – підтримка обміну даними між ESP32 та іншими BLE-пристроями, такими як смартфони чи інші мікроконтролери.
- зручне керування – простий інтерфейс для взаємодії з характеристиками, дозволяючи легко читати та записувати дані.
- підтримка сервісів – можливість визначення та використання різноманітних сервісів для взаємодії з BLE-пристроями.

3.1 Розробка MESH модулю

У даному проєкті використовується технологія MESH для створення мережі взаємопов'язаних мікроконтролерів ESP32. Розробка MESH модулю включає в себе наступні аспекти: [34]

1. Ініціалізація мережі MESH

Код містить налаштування основних параметрів мережі, таких як префікс мережі, пароль та порт. Ці параметри гарантують унікальність та безпеку мережі.


```

#define MESH_PREFIX      "Mesh_username"
#define MESH_PASSWORD   "Mesh_password"
#define MESH_PORT        5555

mesh.init( MESH_PREFIX, MESH_PASSWORD, &userScheduler, MESH_PORT );

```

Рисунок 3.1 – ініціалізація мережі MESH

2. Відправка даних в мережі

Функція `sendMessage()` відправляє ширококомовний запит із зчитаними з датчика температур та іншими параметрами до всіх вузлів мережі. Це забезпечує взаємодію між вузлами та обмін інформацією.

```

void sendMessage () {
    String msg = obtain_readings();
    mesh.sendBroadcast(msg);
}

```

Рисунок 3.2 – відправка даних у мережу MESH

3. Отримання даних в мережі:

Функція `receivedCallback()` викликається при отриманні даних від інших вузлів. Дані перетворюються з формату JSON, а потім використовуються для управління параметрами системи.

```

void receivedCallback( uint32_t from, String &msg ) {
    Serial.printf("\nReceived from %u msg=%s\n", from, msg.c_str());
    JSONVar json_object = JSON.parse(msg.c_str());
    int node = json_object["Node"];
    double temp = json_object["Temperature"];
    int setNode = json_object["SetNode"];
    double setTemp = json_object["SetTemperature"];
}

```

Рисунок 3.3 – отримання даних у мережі MESH

4. Забезпечення надійності та взаємодії

Функції, такі як ``onNewConnection()``, ``onChangedConnections()`` та ``onNodeTimeAdjusted()``, реалізовані для взаємодії при нових підключеннях та коригуванні часу мережі, забезпечуючи стабільну та надійну роботу.

```
mesh.onNewConnection(&newConnectionCallback);
mesh.onChangedConnections(&changedConnectionCallback);
mesh.onNodeTimeAdjusted(&nodeTimeAdjustedCallback);
```

Рисунок 3.4 – забезпечення надійності з'єднання у мережі MESH

Розробка MESH модулю дозволяє створювати розподілені мережі, об'єднуючи різні вузли для спільного управління опалюванням та обміном інформацією в системі.

3.2 Розробка BLE модулю

В системі "розумних" нагрівачів використовується технологія Bluetooth Low Energy (BLE) для бездротового зв'язку між мікроконтролерами ESP32 та смартфонами на платформі Android. Цей BLE модуль відіграє ключову роль у забезпеченні взаємодії між різними частинами системи. [30, 39]

```
#include <BLEDevice.h>
#include <BLEServer.h>
#include <BLEUtils.h>
#include <BLE2902.h>
```

Рисунок 3.5 – Ініціалізація бібліотек для BLE

Код містить ініціалізацію BLE пристрою, створення BLE серверу та необхідних характеристик для обміну даними.

```

BLEDevice::init("Smart_Heater_BLE");
BLEServer pServer = BLEDevice::createServer();
BLEService pService = pServer->createService(SERVICE_UUID);

BLECharacteristic pCharacteristic = pService->createCharacteristic(
    CHARACTERISTIC_UUID_TX,
    BLECharacteristic::PROPERTY_NOTIFY
);
pCharacteristic->addDescriptor(new BLE2902());

BLECharacteristic pRxCharacteristic = pService->createCharacteristic(
    CHARACTERISTIC_UUID_RX,
    BLECharacteristic::PROPERTY_WRITE
)->setCallbacks(new MyCallbacks());

BLEDevice::init("Smart_Heater_BLE");
BLEServer pServer = BLEDevice::createServer();
BLEService pService = pServer->createService(SERVICE_UUID);

BLECharacteristic pCharacteristic = pService->createCharacteristic(
    CHARACTERISTIC_UUID_TX,
    BLECharacteristic::PROPERTY_NOTIFY
);
pCharacteristic->addDescriptor(new BLE2902());

BLECharacteristic pRxCharacteristic = pService->createCharacteristic(
    CHARACTERISTIC_UUID_RX,
    BLECharacteristic::PROPERTY_WRITE
)->setCallbacks(new MyCallbacks());

```

Рисунок 3.6 – Підняття серверу BLE

```

BLEAdvertising pAdvertising = BLEDevice::getAdvertising();
pAdvertising->addServiceUUID(SERVICE_UUID);
pAdvertising->setScanResponse(true);
pAdvertising->setMinPreferred(0x06);
pAdvertising->setMinPreferred(0x12);

pService->start();
BLEDevice::startAdvertising();

```

Рисунок 3.7 – Ініціалізація та надсилання даних

```

class MyCallbacks : public BLECharacteristicCallbacks {
    void onWrite(BLECharacteristic *pCharacteristic) {
        std::string value = pCharacteristic->getValue();
        // Обробка отриманих даних
    }
};
|

```

Рисунок 3.8 – Обробка отриманих даних

BLE модуль дозволяє кожному мікроконтролеру в системі взаємодіяти з іншими пристроями через BLE-з'єднання. Це забезпечує зручний та надійний спосіб обміну даними між контролерами та смартфонами.

Плюси/мінуси:

- ефективне та енергоефективне BLE з'єднання для обміну невеликими обсягами даних;
- підтримка BLE вбудована в більшість Android-смартфонів, що робить його зручним для використання;
- простота ініціалізації та використання BLE сервісів та характеристик;
- обмежений обсяг передаваних даних в порівнянні з іншими технологіями;
- можливість втрати з'єднання на великій відстані чи в умовах перешкод;
- не всі пристрої підтримують BLE.

3.3 Розробка модулю контролю/підтримання заданої температури

Розглянемо окремі аспекти коду, які відповідають за керування/підтримання заданої температури.

```

void loop() {
    mesh.update(); //start mesh network
}

```

Рисунок 3.9 – Оновлення Мережі та Передача Даних

У цій секції коду викликається метод `mesh.update()`, який відповідає за оновлення мережі Mesh.

```
//set values to relay(heating/cooling)
digitalWrite(relay_pin_1, LOW);
digitalWrite(relay_pin_2, HIGH);
```

Рисунок 3.10 – Встановлення Значень Реле

У цій частині встановлюються значення для реле: `relay_pin_1` та `relay_pin_2`. Вони використовуються для управління системою обігріву та охолодження.

```
//upper and lower threshold
TEMP_THRESHOLD_UPPER = tempFromBL + 1;
TEMP_THRESHOLD_LOWER = tempFromBL - 1;
```

Рисунок 3.11 – Визначення Порогових Значень Температури

У цій частині визначаються верхні та нижні порогові значення температури в залежності від `tempFromBL` (задана температура).

```
//change polarity of Peltier element (cooling/heating)
if (currentTemp < tempFromBL) {
  //cooling
  digitalWrite(relay_pin_1, LOW);
  digitalWrite(relay_pin_2, HIGH);
} else {
  //heating
  digitalWrite(relay_pin_1, HIGH);
  digitalWrite(relay_pin_2, LOW);
}
```

Рисунок 3.12 - Зміна Полярності Елемента Пельтьє

У цій частині здійснюється зміна полярності елемента Пельтьє в залежності від того, чи поточна температура менше чи більше за задану.

```
//temperature control
if (tempFromBL != 0 || currentTemp != 0) {
  if(currentTemp < TEMP_THRESHOLD_LOWER){
    digitalWrite(IRL, HIGH);
    digitalWrite(IRL_LED, HIGH);
  } else if(currentTemp > TEMP_THRESHOLD_UPPER){
    digitalWrite(IRL, LOW);
    digitalWrite(IRL_LED, LOW);
  }
} else {
  digitalWrite(IRL, LOW);
  digitalWrite(IRL_LED, LOW);
}
```

Рисунок 3.13 – Керування температури та реле

Реалізація контролю температури та керування реле в залежності від порогових значень показано на рисунку 3.13. Якщо поточна температура нижче порогу, вмикається індикатор та реле для нагріву; якщо вище порогу, вмикається індикатор та реле для охолодження.

Модуль контролю та підтримання температури визначає, чи потрібно включити нагрівач чи використувувати елемент Пельтьє для збереження заданої температури. Це забезпечує комфорт у керуванні температурою пристроїв та дозволяє ефективно використувувати енергію.

Плюси/Мінуси:

- Простий та ефективний спосіб контролю та підтримання температури;
- Використання датчика температури DS18B20 для точного зчитування температури;
- Можливість зміни режимів роботи (нагрів, охолодження) в залежності від поточної температури;
- Залежність від точності датчика температури;

- Обмежені можливості регулювання, порівняно з більш складними алгоритмами контролю температури.

3.4 Опис хвильового алгоритму для керування температурою у комплексній системі нагрівачів

Однією з основних вимог є "Легкість реалізації програмного та технічного забезпечення", що досягається за допомогою децентралізованої координації. Це відмінність та перевага у порівнянні з системами розумного дому. Децентралізована координація не потребує центрального контролера та впровадження процедур зміни програмного чи апаратного забезпечення при зміні кількості або місцезнаходження нагрівачів. [50]

Для контролю послідовності координації локальної системи використовується хвильовий метод. Цей метод може бути реалізований як синхронний чи асинхронний. У синхронному методі використовується генератор хвильових імпульсів синхронізації. Асинхронний алгоритм реалізується через передачу токена. Координатор кожного MSH (Movable Smart Heaters) визначає оптимальне управління, враховуючи його вплив на всі ε -середовище. Щоб уникнути нестабільності в координації через конфлікт між локальною та глобальною оптимізацією, вони модифікували хвильовий алгоритм Лі (1961) та ввели компромісний коефіцієнт ρ_k [3].

У дослідженні автора [2] розглядалися принципи децентралізованої координації в розподілених кіберфізичних системах управління технологічними процесами. В роботі була запропонована структура координатора, який включає такі модулі:

1. модель об'єкта;
2. інтерфейсний модуль;
3. модуль оцінки невизначених параметрів;
4. модуль кластеризації;

5. модуль прогнозування;
6. модуль оптимізації критеріїв;
7. модуль керування послідовністю узгодження;
8. модуль налаштування параметрів;
9. модуль зв'язку Wi-Fi MESH.

Зазначені в координаторі MSH модулі повинні відповідати декільком вимогам [3]:

- програмне та технічне забезпечення, а також додатковий контролер, необхідні для реалізації цих модулів, не повинні значно підвищувати вартість обігрівачів;
- алгоритми налаштування та координації мають бути високошвидкісними. Навіть при інерції теплових процесів у приміщеннях процеси налаштування та узгодження повинні відбуватися швидко, щоб забезпечити необхідний комфорт для користувачів. Додаткові вимоги до динаміки пов'язані з рухливістю нагрівачів; зміна їх взаємного розташування та включення/вимикання призводять до зміни структури системи координації та потребують додаткового налаштування;
- повна автоматизація. Користувач MSH не повинен проводити складні процедури налаштування системи. Достатньо запустити процес налаштування координації зі смартфона, встановити потрібну температуру та обрати тип критерію координації. До цього координатор включає модулі підключення та налаштування.

Процедура визначення набору MSH в будь-якому приміщенні на основі випробувань показана на рисунку 3.14. При активації першого MSH або зміні налаштувань комфортної температури чи положення MSH запускається процедура випробування. Координатор відправляє повідомлення іншим координаторам на MSH, які реєструють температури в різних областях. Після 5 циклів обігріву реле MSH висилає нове повідомлення. Потім проводиться

кореляційний аналіз температур і розрахунок параметрів координації. Оскільки теплові потоки пропорційні різниці температур, кореляційна функція розраховується як:

$$B_{T_i(T_k-T_i)} = B_{T_i T_k} - D_{T_i} \quad (3.1)$$

де D_{T_i} - це температурна дисперсія і-го обігрівача. [3]

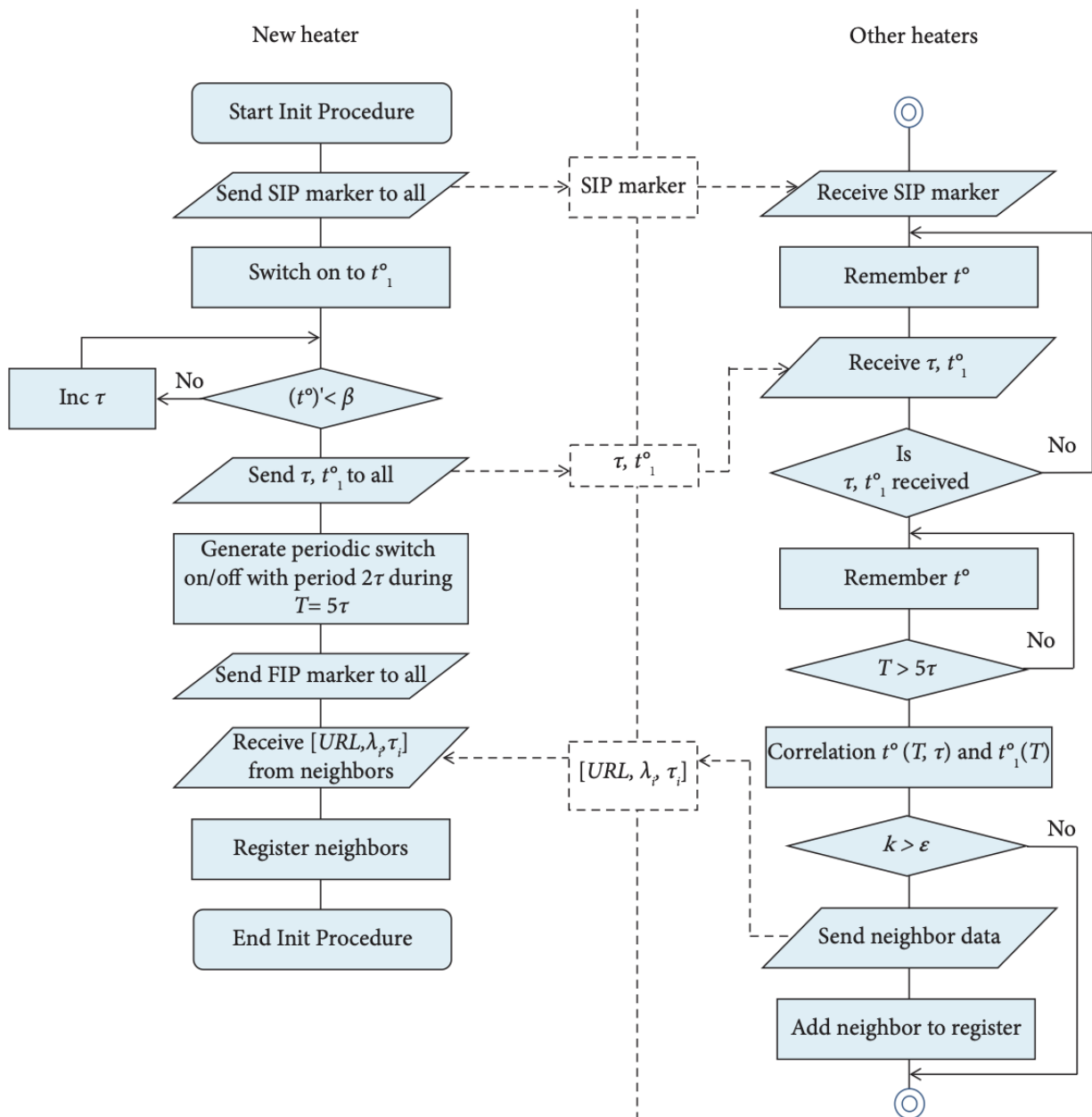


Рисунок 3.14 – Алгоритм визначення набору MSH на основі процедури ударних випробувань [3]

Хвильовий алгоритм відіграє ключову роль у вдосконаленні координації розподіленої системи керування температурою. Ми розглядаємо два варіанти: синхронний та асинхронний.

1. Синхронний Алгоритм: Використовує хвильовий генератор для імпульсів синхронізації. Важливий для узгодженого керування в умовах однорідності.

2. Асинхронний Алгоритм: Базується на передачі маркера та може бути важливим для реалізації координації в розподілених об'єктах з великою відстанню між елементами.

Алгоритм кластеризації та визначення набору:

1. Алгоритм Кластеризації: Метод визначення множини Mobile Smart Heaters (MSH) для ефективного управління в багатозональних приміщеннях. Враховує координати зон, відстань та коефіцієнти взаємного впливу.

2. Алгоритм Визначення Набору: Використовує процедуру ударних випробувань та тестування для оптимізації координації MSH. Передбачає кореляційний аналіз температур та розрахунок параметрів коригування.

Основні алгоритми координації та динамічного настроювання - перша вимога «Простота програмного забезпечення та технічної реалізації» забезпечується децентралізованою координацією. Це головна відмінність і перевага в порівнянні з системами «Розумний дім». Децентралізована координація не вимагає наявності центрального контролера та виконання процедур зміни його програмної та/або апаратної конфігурації при зміні кількості або розташування нагрівачів.

У випадку синхронної роботи контролерів обігріву реле, де $0 \leq \varepsilon \leq 1$; F_i - заданий стан i -тої області; T_{0k} - стан k -тої навколишньої області, визначений координатором; d_k - відстань від вказаної області до k -тої навколишньої області; λk - константа передачі тепла; τ - константа часу передачі тепла.

$$(k \in \mathbf{K}_{i\varepsilon}) \longrightarrow \left(\forall t: \frac{1}{F_i} \sum_{k \notin \mathbf{K}_{i\varepsilon}} \frac{T_{0k}}{8(\pi\lambda_k t/\tau)^{3/2}} e^{-d_{ki}^2/4\lambda_k t/\tau} < \varepsilon \right), \quad (3.2)$$

Зі збільшенням відстані $|Z - Z_i|$ контрольного пункту Z_i рухомого розумного обігрівача від заданої області, вплив контролю зменшується. Отже, при децентралізованій координації кожен координатор повинен враховувати лише ті керовані елементи, які знаходяться в його безпосередньому оточенні що зображено на рисунку 3.15, тобто кластер, межі якого визначаються модулем кластеризації.

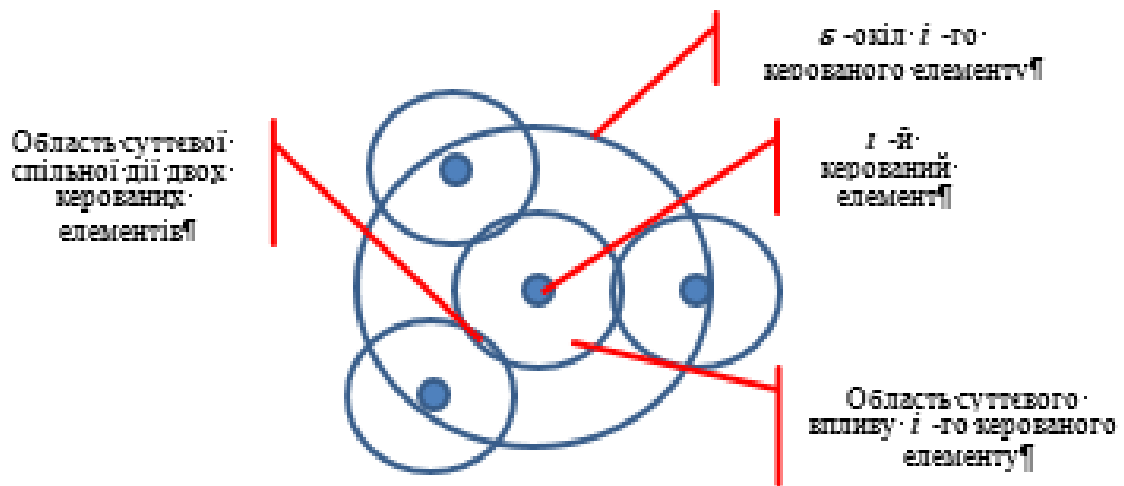


Рисунок 3.15 – Визначення ε -околу елемента [3]

- у розподіленому об'єкті рівномірно, то умові оптимальності задовольняє функція керування . Тоді з (3.2) радіус – околу можна оцінити так:

- при асинхронній роботі релейних регуляторів нагрівачів

$$d_{i\varepsilon} : \left\{ d_{i\varepsilon} = \min d_k, \quad d_k : \left[\sum_{k \notin \mathbf{K}_{i\varepsilon}} \max_t \frac{T_{0k}}{8(\pi\lambda t/\tau)^{3/2}} e^{-\frac{d_k^2}{4\lambda t/\tau}} \right] < \varepsilon \cdot F_0 \right\}, \quad (3.3)$$

- при синхронній роботі релейних регуляторів нагрівачів

$$d_{i\varepsilon} : \left\{ d_{i\varepsilon} = \min d_k, \quad d_k : \max_t \left[\sum_{k \in \mathbf{K}_{i\varepsilon}} \frac{T_{0k}}{8 \left(\pi \lambda \frac{t}{\tau} \right)^{3/2}} e^{-\frac{d_k^2}{4\lambda t/\tau}} \right] < \varepsilon \cdot F_0 \right\} \quad (3.4)$$

Дослідження функції $\frac{T_{0k}}{8 \left(\pi \lambda \frac{t}{\tau} \right)^{3/2}} e^{-\frac{d_k^2}{4\lambda t/\tau}}$ на максимум по t показує, що

максимум досягається при $t_m = \frac{d_k^2 \tau}{6\lambda}$. Підставимо t_m в умову (3.3) і

отримуємо:

$$d_{i\varepsilon} : \left\{ d_{i\varepsilon} = \min d_k, \quad d_k : \sum_{k \in \mathbf{K}_{i\varepsilon}} \frac{1.48 \cdot T_{0k}}{d_k^3} < \varepsilon \cdot F_0 \right\} \quad (3.5)$$

Такий сценарій застосування є практично недосяжним у реальних умовах, оскільки він передбачає ідеально однорідний та нескінченно розподілений об'єкт з великою кількістю елементів управління та мінімальною відстанню між ними. Однак цей випадок може бути розглянутий як екстремальний для тестування алгоритмів координації та є певною апроксимацією реальних завдань.

Використання таких методів дозволяє оптимізувати систему керування температурою, враховуючи розподіленість та відстань між елементами, що є ключовим у великих приміщеннях.

Застосування хвильового алгоритму сприяє удосконаленню системи керування температурою, забезпечуючи оптимальну координацію між різними елементами. Це особливо важливо для забезпечення комфортних умов в багатозональних приміщеннях, де елементи розташовані на різних відстанях один від одного.

3.5 Висновок до розділу 3

У даному розділі було розглянуто розробку програмного забезпечення для мікроконтролера ESP32, який використовується в системі децентралізованого керування опаленням.

Було використано наступні технології та бібліотеки:

- C++
- Arduino
- ArduinoIDE
- ArduinoJSON
- PainlessMesh
- OneWire
- DallasTemperature
- BLE Arduino Library

Розроблено наступні модулі:

- MESH модуль для створення мережі взаємопов'язаних мікроконтролерів
- BLE модуль для бездротового зв'язку між мікроконтролерами та смартфонами
- модуль контролю/підтримання заданої температури
- модуль хвильового алгоритму для керування температурою

Модуль MESH дозволяє створювати розподілені мережі, об'єднуючи різні вузли для спільного управління опалюванням та обміном інформацією в системі.

Модуль BLE дозволяє кожному мікроконтролеру в системі взаємодіяти з іншими пристроями через BLE-з'єднання. Це забезпечує зручний та надійний спосіб обміну даними між контролерами та смартфонами.

Модуль контролю/підтримання температури визначає, чи потрібно включати нагрівач чи використовувати елемент Пельтьє для збереження заданої

температури. Це забезпечує комфорт у керуванні температурою пристроїв та дозволяє ефективно використовувати енергію.

Хвильовий алгоритм є ефективним методом для децентралізованої координації в системах керування температурою. Він забезпечує оптимальне управління, враховуючи вплив кожного елемента на навколишнє середовище. Це особливо важливо для багатозональних приміщень, де елементи розташовані на різних відстанях один від одного.

Основні переваги хвильового алгоритму:

- децентралізованість: алгоритм не вимагає центрального контролера, що робить його більш простим у реалізації та налаштуванні.
- ефективність: алгоритм забезпечує оптимальне управління, враховуючи вплив кожного елемента на навколишнє середовище.
- швидкість: алгоритм є високошвидкісним, що важливо для забезпечення комфортних умов для користувачів.

Однак у хвильового алгоритму є і деякі недоліки:

- складність реалізації: алгоритм може бути складним у реалізації для великих систем з великою кількістю елементів.
- нестабільність: алгоритм може бути нестабільним в умовах, коли елементи розташовані на великих відстанях один від одного.

В цілому, хвильовий алгоритм є ефективним методом для децентралізованої координації в системах керування температурою. Він має ряд переваг, які роблять його привабливим вибором для таких систем.

4 АНАЛІЗ, РОЗРОБКА ТА ТЕСТУВАННЯ ФУНКЦІОНАЛЬНОСТІ МОБІЛЬНОГО ЗАСТОСУНКУ

4.1 Аналіз функціональних частин застосунку

Мобільний додаток розроблено з метою надання користувачам зручного та ефективного інструменту для керування комплексною системою опалення. Мета включає в себе створення інтуїтивно - зрозумілого інтерфейсу, що сприяє легкому використанню, та надання функцій для зручного моніторингу та контролю температури.

Аналіз потреб користувачів та особливостей системи опалення виділив важливість введення таких функцій, як:

- пошук та відображення доступних пристроїв для керування;
- зміна заданої температури за допомогою інтуїтивних кнопок;
- моніторинг та відображення поточної температури;
- відображення рівня сигналу (RSSI) для отримання у режимі реального часу якості з'єднання;
- можливість відключення/з'єднання.

Мобільний додаток пропонує зручний спосіб керування системою опалення, забезпечуючи:

- легкий пошук та підключення пристроїв;
- інтуїтивний інтерфейс для швидкої зміни температури;
- зручний моніторинг стану системи;
- забезпечення стабільного та надійного з'єднання.

4.2 Підтримка технології BLE

Додаток використовує технологію Bluetooth Low Energy (BLE) для забезпечення ефективної передачі даних та низького енергоспоживання між

мобільним пристроєм та ESP32, підвищуючи тим самим зручність використання та продуктивність.

Bluetooth - це технологія бездротової передачі даних на короткі відстані. []
Вона використовується для різних цілей, включаючи:

- обмін файлами між пристроями;
- підключення до периферійних пристроїв, таких як принтери та аудіогарнітури;
- бездротове управління пристроями.

Bluetooth був розроблений компанією Ericsson у 1994 році. Він був стандартизований у 1999 році і з тих пір і став одним з найпопулярніших бездротових протоколів у світі. [35]

Bluetooth працює в діапазоні частот 2,4 ГГц і підтримує швидкість передачі даних до 2 Мбіт/с. Метод розділення каналу за часом (TDD), який використовується Bluetooth, надає кожному пристрою власний час для передачі даних, забезпечуючи ефективність та надійність передачі. Це забезпечує більш надійну та ефективну передачу даних, ніж метод розділення каналу за частотою (FDD), який використовується в інших бездротових протоколах, таких як Wi-Fi. Він має максимальну дальність дії до 100 метрів.

Основні характеристики Bluetooth включають:

- низька потужність: Bluetooth-пристрої відзначаються економією енергії, що дозволяє їм працювати тривалий час від батареї;
- низька вартість: Bluetooth-пристрої відносно недорогі;
- легкість використання: підключення та використання Bluetooth-пристроїв є простими та зручними.

Bluetooth має кілька типів:

- Bluetooth Basic Rate (BR): Цей тип Bluetooth є найпоширенішим. Він використовується для передачі даних на швидкості до 1 Мбіт/с;
- Bluetooth Enhanced Data Rate (EDR): Цей тип Bluetooth забезпечує більш високу швидкість передачі даних, до 3 Мбіт/с;

- Bluetooth Low Energy (BLE): Цей тип Bluetooth забезпечує низьке енергоспоживання. Він використовується для пристроїв, яким не потрібна висока швидкість передачі даних, наприклад, для носіїв розумних годинників.

Bluetooth Low Energy (BLE) BLE – це різновид Bluetooth, який забезпечує низьке енергоспоживання. Він використовується для пристроїв, яким не потрібна висока швидкість передачі даних, наприклад, для носіїв розумних годинників, датчиків та розумних ламп. [39-40]

BLE має такі переваги перед BR та EDR:

- низьке енергоспоживання: BLE-пристрої споживають набагато менше енергії, ніж BR- або EDR-пристрої. Це дозволяє їм працювати протягом тривалого часу від батарей;
- довгий час роботи від батарей: BLE-пристрої можуть працювати від батарей протягом місяців або навіть років;
- малий розмір: BLE-модулі невеликі за розміром і можуть бути легко інтегровані в різні пристрої.

UID BLE-пристрої мають унікальні ідентифікатори UID. UID - це 128-бітне число, яке присвоюється кожному BLE-пристрою. UID використовується для ідентифікації BLE-пристрою в мережі.

Ми вибрали BLE для мобільного додатку для управління системами опалення з наступних причин:

- підтримка більшістю нових пристроїв на ринку, а особливо технікою Apple, так як пристроїв на iOS було написано застосунок для керування комплексною системою опалення;
- низьке енергоспоживання: BLE-пристрої споживають набагато менше енергії, ніж BR- або EDR-пристрої. Це важливо для мобільних пристроїв, які обмежені за енергією;

- довгий час роботи від батарей: BLE-пристрої можуть працювати від батарей протягом місяців або навіть років. Це важливо для систем опалення, які часто працюють протягом тривалого часу;
- малий розмір: BLE-модулі невеликі за розміром і можуть бути легко інтегровані в різні пристрої. Це важливо для систем опалення, які часто мають обмежене місце всередині.

4.3 Вибір та аналіз інструментів розробки – BLE – огляд ядра Bluetooth

Структура Bluetooth базується на основному фреймворку, що дозволяє програмам iOS взаємодіяти з пристроями Bluetooth Low Energy (BLE), забезпечуючи обмін даними. Наприклад, за допомогою програми можна виявляти, досліджувати та взаємодіяти з периферійними пристроями, такими як цифрові термостати та інші, які працюють із низьким енергоспоживанням [58].

Фреймворк виступає як абстракція для специфікації Bluetooth 4.0, спрощуючи взаємодію з пристроями BLE. Незважаючи на це, він приховує багато технічних деталей специфікацій розробника, що сприяє полегшенню розробки програм для взаємодії з пристроями Bluetooth Low Energy. Оскільки структура ґрунтується на специфікації, вона використовує деякі концепції та термінологію, взяті з цієї специфікації.

Для розробки програм для iOS, пов'язаних з версією iOS 10.0 або пізнішою, обов'язкове включення ключів у опис використання в файлі Info.plist для типів даних, до яких програмі потрібен доступ. Важливо включити ключ `NSBluetoothPeripheralUsageDescription` для отримання доступу до периферійних даних Bluetooth, інакше може відбутися збій в роботі програми.

4.3.1 Центральні, периферійні пристрої та їхня роль у зв'язку Bluetooth

У світі зв'язку Bluetooth з низьким енергоспоживання виокремлюють два основні учасники: центральний та периферійний пристрої. Відповідно до традиційної моделі клієнт-сервер, периферійний пристрій має в своєму розпорядженні дані, які можуть бути корисними для інших пристроїв. Центр використовує надану периферійними пристроями інформацію для виконання конкретних завдань. Наприклад, мікроконтролер може містити інформацію, яка може бути використана програмою на iOS для відображення поточної температури оточуючого середовища у зручному форматі.

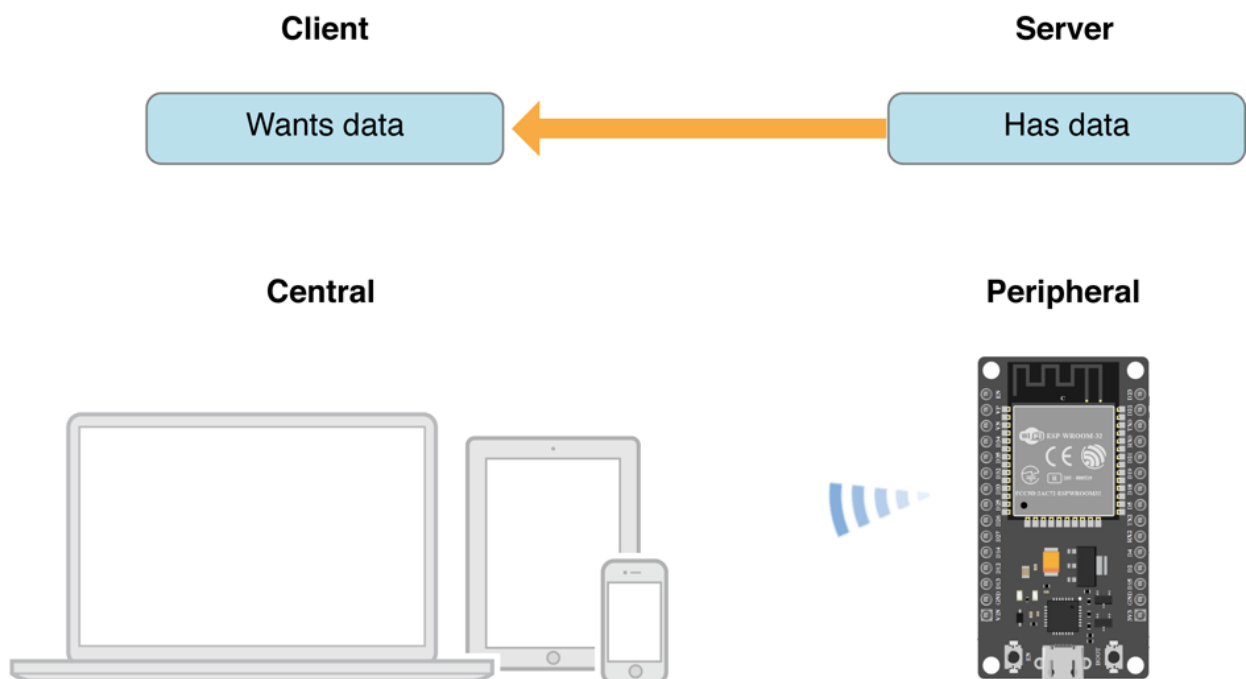


Рисунок 4.1 – Центральні та периферійні пристрої [58]

Трансляційні пакети є ключовим елементом для периферійних пристроїв, оскільки вони передають частину своїх даних. Такий пакет представляє собою невеликий обсяг даних, який містить корисну інформацію про периферійний пристрій, таку як його назва та основні функції. Наприклад, цифровий термостат

може транслювати дані про поточну кімнатну температуру. В контексті Bluetooth Low Energy, трансляція виступає як основний метод повідомлення периферійних пристроїв про свою присутність.

З іншого боку, центральний пристрій може сканувати та прослуховувати будь-який периферійний пристрій, який транслює інформацію, що його цікавить, як показано на малюнку 4.2. Центр може попросити підключитися до будь-якого периферійного пристрою, який виявив трансляцію.



Рисунок 4.2 – Трансляція та відкриття [58]

Мета підключення до периферійного пристрою полягає в тому, щоб почати, досліджувати та взаємодіяти з даними, які він пропонує.

Периферійні пристрої можуть містити одну або кілька служб або надавати корисну інформацію про силу підключеного сигналу. Сервіс/послуга – це набір даних і пов’язаних дій для виконання функції або функції пристрою (або його частин). Наприклад, одна з послуг мікроконтролеру може полягати в оприлюдненні даних про температуру навколишнього середовища із датчика температури.

Самі послуги складаються з характеристик або включених послуг (тобто посилань на інші послуги). Характеристика надає додаткові відомості про послугу периферійного пристрою.

Наприклад, щойно описана служба визначення температури навколишнього середовища може містити одну характеристику, яка описує передбачуване розташування мікроконтролеру приєднаного по технології MESH пристрою, і іншу характеристику, яка передає дані вимірювання температури

навколишнього середовища у нього. Рисунок 4.3 ілюструє одну можливу структуру служби та характеристики мікроконтролера.



Peripheral

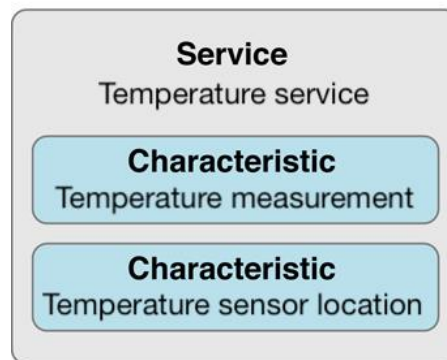


Рисунок 4.3 – Послуги та характеристики периферійного пристрою [58]

Після успішного встановлення з'єднання між централізованим та периферійним пристроєм, централізований пристрій може виявити повний перелік доступних послуг та характеристик, які пропонує периферійний пристрій. Варто враховувати, що транслявання даних може містити лише обмежену частину доступної інформації про послуги.

Центральний пристрій також може взаємодіяти зі службою периферійного пристрою, читаючи або записуючи значення характеристик цієї служби. Наприклад, програма може запитувати поточну температуру в кімнаті від цифрового термостата або надавати значення, за яким можна налаштувати кімнатну температуру.

Основні учасники та дані, які використовуються у зв'язку Bluetooth з низьким енергоспоживанням, ілюструються структурою Core Bluetooth простим та зрозумілим способом.

Об'єкти на центральній стороні:

- дії на центральній стороні зв'язку Bluetooth з низьким споживанням енергії відбуваються тоді, коли локальний центр взаємодіє з віддаленим периферійним пристроєм. У випадку, якщо локальний периферійний пристрій не налаштований та не використовується для відповіді на запити центрального пристрою, значна частина транзакцій Bluetooth обробляється на центральній стороні.

Локальні пристрої та віддалені периферійні пристрої:

- локальні центральні пристрої, представлені об'єктом CBCentralManager на центральній стороні, використовуються для управління виявленими або підключеними віддаленими периферійними пристроями, які представлені об'єктами CBPeripheral. Ці дії включають сканування, виявлення та підключення до рекламних периферійних пристроїв. Структура Core Bluetooth ілюструє це, представляючи як локальні центральні, так і віддалені периферійні пристрої, як показано на малюнку 4.4.

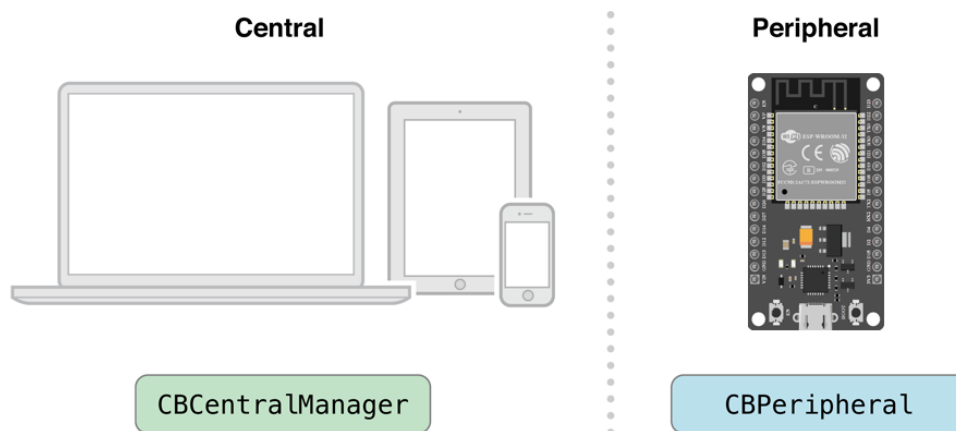


Рисунок 4.4 – Основні об'єкти Bluetooth на центральній стороні [58]

При взаємодії з даними на віддаленому периферійному пристрої (який є об'єктом `CBPeripheral`), основна увага приділяється його службам та характеристикам. В структурі Core Bluetooth послуги віддаленого периферійного пристрою відображаються за допомогою об'єктів `CBService`, а характеристики послуги віддаленого периферійного пристрою представлені об'єктами `CBCharacteristic`. Малюнок 4.5 наочно демонструє основну структуру послуг та характеристик віддаленого периферійного пристрою.

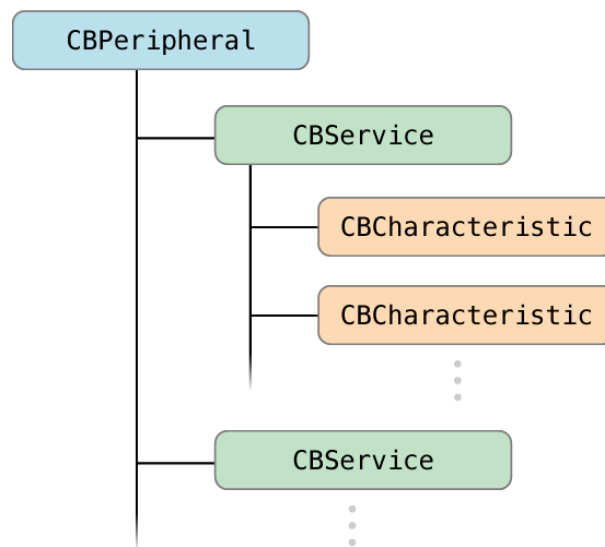


Рисунок 4.5 – Дерево послуг і характеристик віддаленого периферійного пристрою [58]

На локальній периферійній стороні представлений об'єкт `CBPeripheralManager`, який використовується для управління опублікованими службами в базі даних послуг і характеристик локального периферійного пристрою. Крім того, він служить для оголошення цих послуг на віддалених центральних пристроях, які є об'єктами `CBCentral`. Об'єкти диспетчера периферійних пристроїв використовуються для відповіді на запити читання та запису від цих віддалених централів. Малюнок 4.6 ілюструє, як локальні периферійні пристрої та віддалені центральні пристрої взаємодіють у структурі Core Bluetooth.

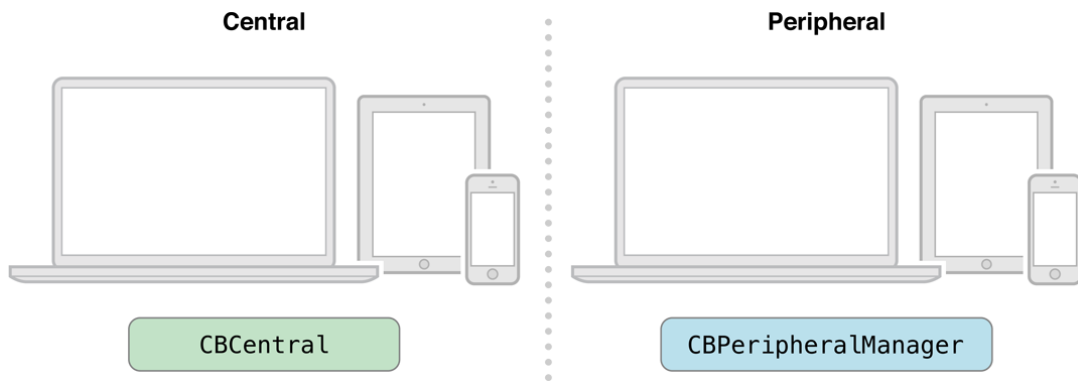


Рисунок 4.6 – Основні об'єкти Bluetooth на периферійній стороні [58]

Дані локального периферійного пристрою відображаються у вигляді змінних версій служб та характеристик, які представлені об'єктами `CBMutableService` та `CBMutableCharacteristic`.

При налаштуванні та взаємодії з даними на локальному периферійному пристрої, який є об'єктом `CBPeripheralManager`, використовуються змінні версії його служб та характеристик. У структурі Core Bluetooth служби цього периферійного пристрою відображаються об'єктами `CBMutableService`, а характеристики служби – об'єктами `CBMutableCharacteristic`. На малюнку 4.7 показано основну структуру послуг та характеристик локального периферійного пристрою.

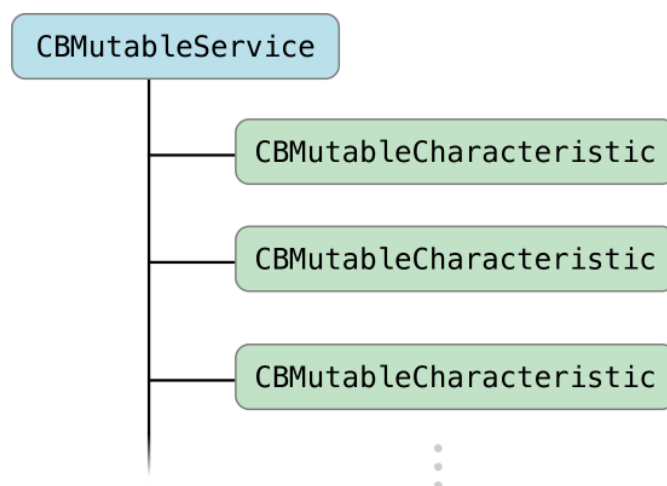


Рисунок 4.7 – Дерево послуг і характеристик локального периферійного пристрою [58]

4.4 Визначення середовища та мови програмування

Вибір розробки додатку для iOS на мові Swift та у середовищі розробки Xcode для керування контролером температури може бути обумовлений кількома об'єктивними та суб'єктивними причинами, незалежними від самої задачі керування комплексною системою опалення. Ось кілька причин:

- екосистема Apple: Вибір розробки для iOS на Swift і у Xcode логічний, якщо ви хочете зосередитися на платформі Apple. Swift і Xcode - це офіційні інструменти від Apple для розробки на їхніх пристроях, що дозволяє забезпечити найбільшу сумісність та оптимізацію для пристроїв iOS.
- широка Аудиторія Користувачів: iOS має значну кількість користувачів, і розробка для цієї платформи забезпечує широке охоплення аудиторії.
- інструменти розробки: Xcode надає розробникам широкі можливості для створення, налагодження та оптимізації додатків. Інтегрована середовище розробки Xcode спрощує розробку, дозволяючи легко працювати з інтерфейсами, додавати функціональність та виконувати тестування.
- мова Swift: Swift є сучасною та ефективною мовою програмування. Вона має чистий синтаксис та безпеку типів, що полегшує написання стабільного та зрозумілого коду.
- простота розгортання на пристроях Apple: розгортання додатків для iOS на пристроях Apple є простим завдяки App Store. Користувачі можуть легко завантажувати та оновлювати додаток, що полегшує розповсюдження та підтримку.

Розробка мобільного застосунку відбувається у середовищі Xcode 15 як показано на рисунку 4.8 та на мові програмування Swift 5.



Рисунок 4.8 – Версії середовища розробки Xcode

Xcode — це набір інструментів, які розробники використовують для створення програм для платформи Apple. Використовується Xcode, щоб керувати всім процесом розробки — від створення програми до тестування, оптимізації та надсилання її в App Store. [61]

Xcode містить редактор коду, вбудовані інструменти попереднього перегляду SwiftUI, які показують інтерфейс користувача програми і потужний налагоджувач з умовними точками зупинки.

Xcode також містить кілька інструментів розробки, які допомагають швидко створити прототип і протестувати програму. Simulator – для швидкого створення прототипів і тестування програми в симульованому середовищі, коли реальний пристрій недоступний. Доступні інструменти, щоб профілювати й аналізувати програму, покращувати продуктивність і досліджувати використання системних ресурсів. Надається можливість створення 3D-контенту за допомогою Reality Composer, тренування користувачьких моделей машинного навчання за допомогою Create ML і надає можливість оглядати області програми, недоступні за допомогою Accessibility Inspector.

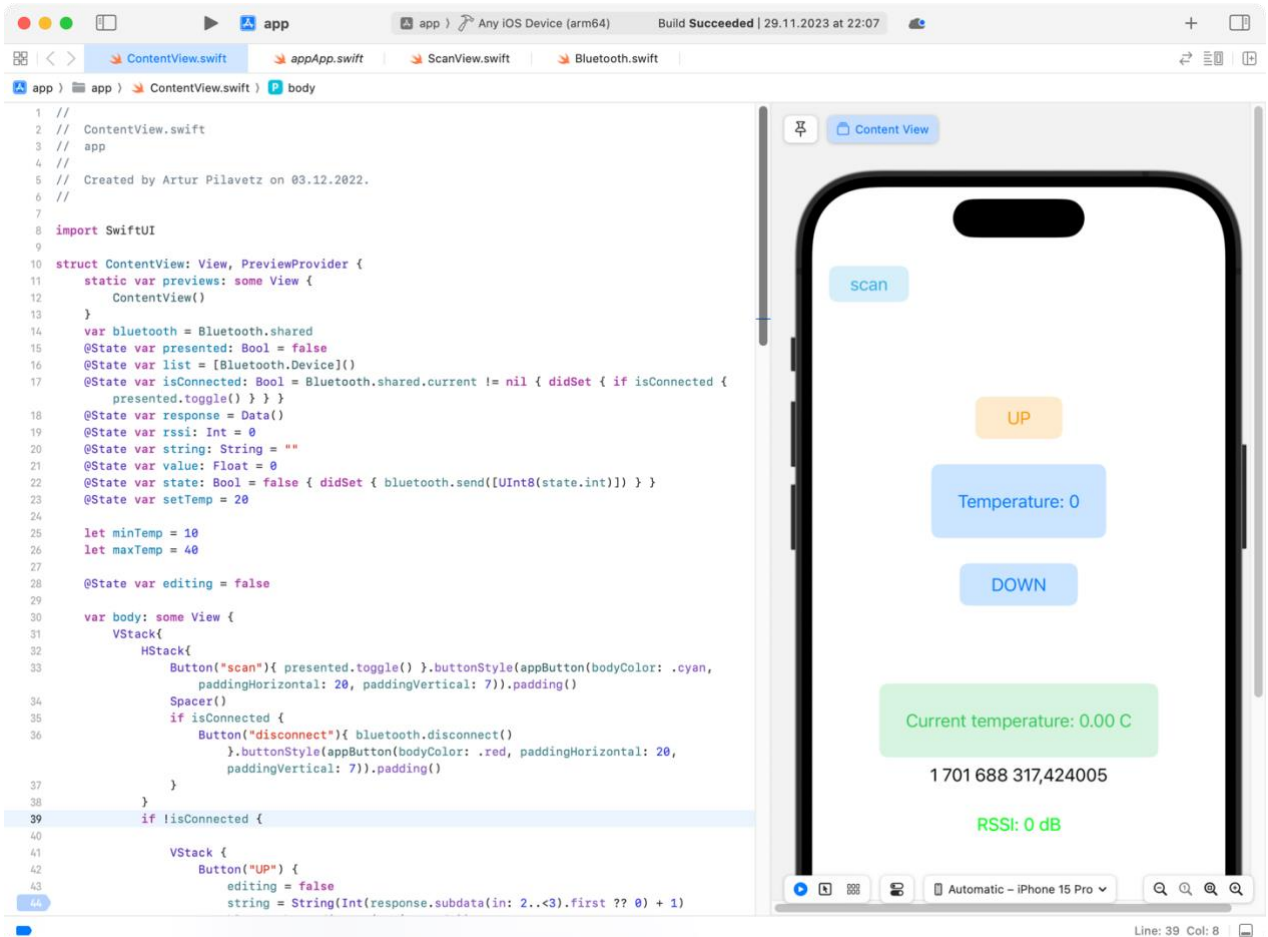


Рисунок 4.9 – Середовище розробки Xcode

Застосунок написано мовою програмування Swift 5 у середовищі розробки Xcode, більш детально про мову Swift наведено далі.

Swift – це об'єктно-орієнтована мова програмування, основний розвиток якої веде компанія Apple. Вона була представлена в 2014 році як наступниця Objective-C. Swift використовує статичну типізацію і є безпечною мовою. [60]

Основні характеристики Swift:

- об'єктно-орієнтована мова: Swift є об'єктно-орієнтованою мовою, що означає, що програми в Swift пишуться як сукупність об'єктів. Об'єкти мають властивості та методи, які описують їх поведінку;
- статична типізація: Swift є мовою з статичною типізацією, що означає, що типи даних змінних та параметрів визначаються при

компіляції. Це допомагає уникнути помилок типізації при виконанні програми;

- безпека: Swift є безпечною мовою, що означає, що вона має ряд механізмів, які допомагають запобігти помилкам. Наприклад, у Swift є автоматичне управління пам'яттю, що означає, що розробнику не потрібно вручну звільняти пам'ять, виділену для об'єктів.

Переваги Swift:

- простота: Swift є відносно простою мовою для вивчення. Вона має невеликий синтаксис і не вимагає багато знань про об'єктно-орієнтоване програмування;
- ефективність: Swift є ефективною мовою, що означає, що програми, написані на Swift, зазвичай працюють швидше, ніж програми, написані на інших мовах;
- безпека: Swift є безпечною мовою, що означає, що вона має ряд механізмів, які допомагають запобігти помилкам. Застосування Swift: Swift використовується для розробки мобільних додатків для iOS та iPadOS. Він також використовується для розробки веб-додатків, десктопних додатків та ігор.

Swift використовується для розробки мобільних додатків для iOS та iPadOS. Він також використовується для розробки веб-додатків, десктопних додатків та ігор.

4.5 Розробка мобільного застосунку

Розробка мобільного застосунку – це процес створення програмного забезпечення для мобільних пристроїв, таких як смартфони та планшети. Розробка мобільного застосунку може включати в себе інтеграцію з різними сервісами та технологіями для досягнення певної функціональності та

забезпечення зручного користувацького досвіду. Даний процес включає в себе такі етапи:

- Аналіз: аналіз вимог до застосунку, таких як функції, які він повинен виконувати, і користувачі, для яких він призначений.
- Дизайн: розробка дизайну застосунку, включаючи інтерфейс користувача та функціональність.
- Розробка: написання коду для застосунку.
- Тестування: тестування застосунку на наявність помилок і збоїв.
- Публікування: публікація застосунку в магазинах додатків таких як AppStore та PlayStore.

4.5.1 Визначення інтерфейсу користувача

Інтерфейс користувача (UI) – це спосіб взаємодії користувача з програмним забезпеченням. Він включає в себе всі елементи, з якими користувач взаємодіє, такі як кнопки, текстові поля, меню та значки.

Хороший інтерфейс користувача робить додаток простим у використанні та приємним для користувача.

Розробку інтерфейсу користувача мобільного додатку можна розділити на такі етапи:

1. Аналіз На першому етапі необхідно провести аналіз потреб користувачів. Які функції повинен виконувати додаток? Які дії будуть виконувати користувачі?

2. Концептуальний дизайн На другому етапі розробляються концептуальні макети інтерфейсу користувача. Макети дозволяють уявити, як буде виглядати додаток.

3. Дизайн На третьому етапі розробляються детальні макети інтерфейсу користувача. Макети включають в себе всі елементи інтерфейсу, такі як кнопки, текстові поля, меню та значки.

4. Розробка На четвертому етапі розробляється інтерфейс користувача за допомогою програмного забезпечення для розробки мобільних додатків.

5. Тестування На п'ятому етапі інтерфейс користувача тестується з використанням різних типів пристроїв і користувачів. Дизайн інтерфейсу користувача При проектуванні інтерфейсу користувача необхідно дотримуватися таких принципів:

Простота: Інтерфейс користувача повинен бути простим у використанні. Користувачі повинні легко розуміти, як користуватися додатком.

Надійність: Інтерфейс користувача повинен бути надійним. Він повинен працювати без збоїв і помилок.

Естетичний вигляд: Інтерфейс користувача повинен бути естетично приємним. Він повинен подобатися користувачам. Елементи інтерфейсу користувача інтерфейс користувача мобільного додатку включає в себе такі елементи:

- кнопки використовуються для виконання дій.
- текстові поля використовуються для введення тексту. Меню: Меню використовуються для відображення списків елементів.
- значки використовуються для представлення інформації.

Хороший інтерфейс користувача робить додаток простим у використанні та приємним для користувача.

4.5.2 Розробка інтерфейсу користувача

Розробка інтерфейсу користувача буде виконуватись за допомогою фреймворку – SwiftUI.

SwiftUI – це інструмент для зручного визначення, як додаток буде виглядати та як користувачі будуть з ним взаємодіяти. Він дозволяє описувати вигляд і поведінку інтерфейсу додатка з декларативним синтаксисом, що полегшує розробку та зміну. Фреймворк також надає інструменти для обробки

подій та управління потоком даних, щоб забезпечити ефективну взаємодію з користувачем. [87]

Основні особливості SwiftUI, які є корисні для розробки програм для системи управління системою опалення, включають:

- декларативний синтаксис: SwiftUI дозволяє описувати структуру та вигляд вашого інтерфейсу з декларативним синтаксисом, замість імперативного. Це полегшує створення та зміну елементів інтерфейсу.
- real-time preview: надається можливість бачити зміни в реальному часі без перезапуску додатка. Це дозволяє швидше визначати, як буде виглядати інтерфейс при внесенні змін.
- вбудовані елементи управління: SwiftUI надає широкий спектр вбудованих елементів управління, таких як тексти, кнопки, списки і т.д. Це полегшує створення різноманітних інтерфейсів.
- обробка взаємодії: SwiftUI надає можливості для взаємодії з користувачем, такі як введення даних, обробка жестів та анімації.
- взаємодія з Swift: SwiftUI інтегрований з мовою програмування Swift, що дозволяє використовувати всю потужність Swift для обробки даних та логіки програми.

Під час розробки інтерфейсу користувача мобільного додатку було приділено особливу увагу вивченню деталей та визначенню власного стилю, що надає додатку естетичної привабливості та високого рівня користувацького досвіду.

Стилізація кнопок – створено унікальний стиль кнопок, який визначається не лише кольором, але й відступами по горизонталі та вертикалі, а також кастомним заокругленням кутів. Це надає користувачеві відчуття оригінальності та дбайливого підходу до дизайну.

Відображення сигналу BLE – важливим елементом інтерфейсу є зміна кольору RSSI (Received Signal Strength Indicator), що відображає якість сигналу

VLE зв'язку. Користувачі можуть легко оцінити стабільність зв'язку: зелений колір вказує на сильний сигнал, тоді як червоний попереджає про можливу втрату зв'язку, що полегшує контроль за підключенням.

Процес створення інтерфейсу в середовищі Xcode показано на рисунку, був завершений із застосуванням фреймворку SwiftUI та сучасних розробницьких підходів, що дозволяє забезпечити зручність керування застосунком.

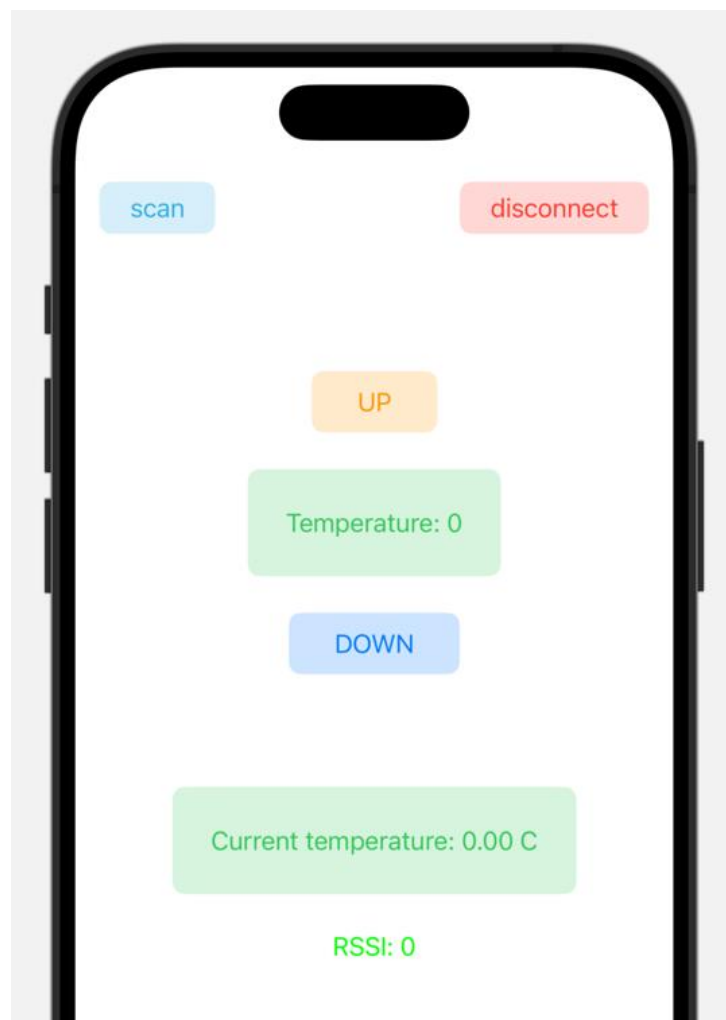


Рисунок 4.10 – Розробка інтерфейсу у середовищі Xcode

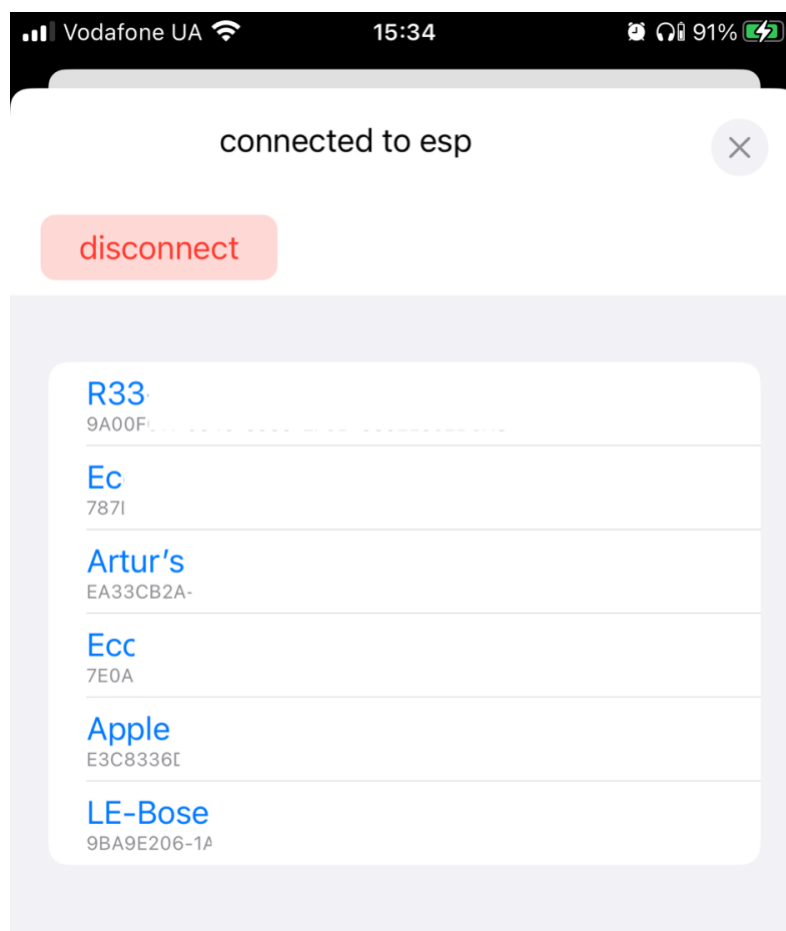


Рисунок 4.11 – відображення списку пристроїв навколо через BLE

На рисунку 4.11 представлено інтерфейс, який відображає список пристроїв, доступних через BLE. Це спрощує користувачеві вибір пристрою та створює зручні умови для навігації в додатку.

Представлено кнопки зміни заданої температури. Відтепер користувач може контролювати та налаштовувати температурні параметри, максимізуючи комфорт та ефективність системи.

На наступних зображеннях демонструється можливість зміни температури з урахуванням комфортних умов, підкреслюючи важливість створення оптимального середовища для користувача.

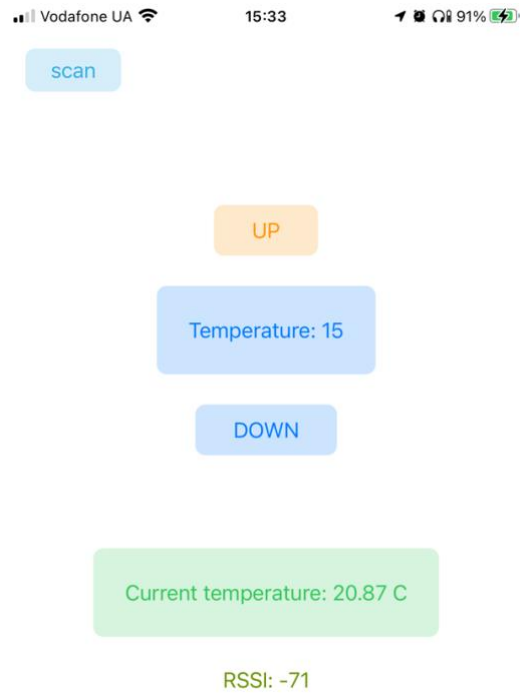


Рисунок 4.12 – зміна поля заданої температури відносно заданої – можливий дискомфорт

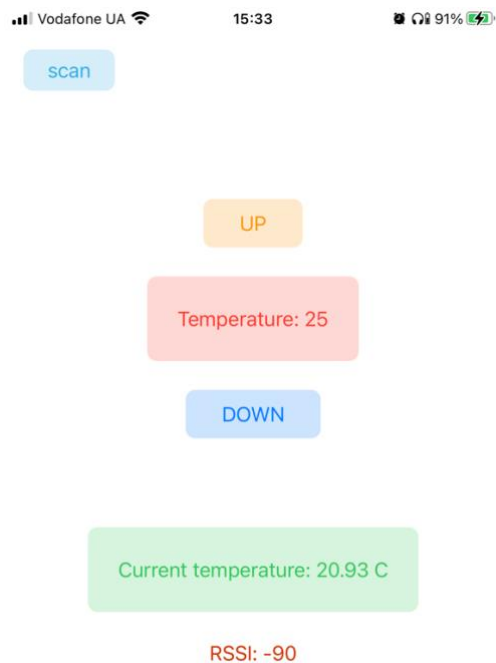


Рисунок 4.13 – зміна поля заданої температури відносно заданої – можливий перевитрати енергії та дискомфорт

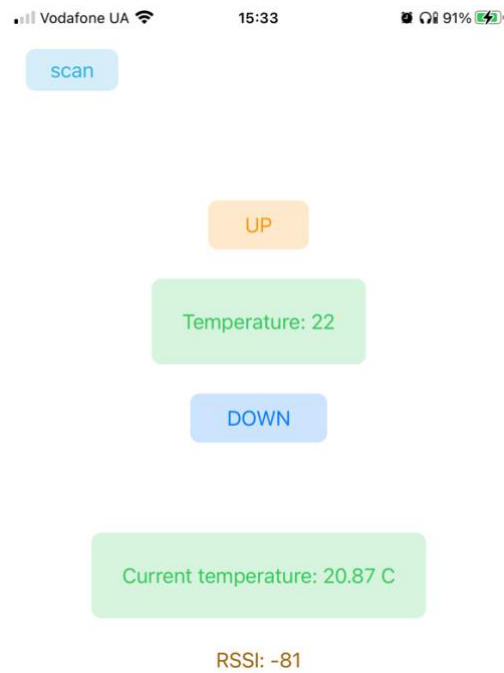


Рисунок 4.14 – зміна поля заданої температури відносно заданої – комфортні умови

Всі ці аспекти виконані із завданням не лише забезпечити ефективність та функціональність, але й створити естетично приємне та інтуїтивно зрозуміле користувацьке оточення для взаємодії з комплексною системою децентралізованого керування опаленням.

4.5.2 Розробка функцій мобільного застосунку

Функції мобільного застосунку розробляються як частини програмного забезпечення, які виконують конкретні завдання чи функції для користувача. Кожна функція призначена для виконання певної операції чи набору операцій, які додають визначену вартість або можливості до застосунку.

CoreBluetooth – це фреймворк, який надає можливості для взаємодії з пристроями, що підтримують технологію Bluetooth Low Energy (BLE) на платформі iOS та macOS. BLE є енергоефективним протоколом бездротового

зв'язку, який дозволяє обмінюватися даними між різними пристроями на низькому рівні споживання енергії. [58]

Можливості фреймворку CoreBluetooth:

- CoreBluetooth дозволяє додатку сканувати та виявляти BLE-пристрої в навколишньому середовищі;
- фреймворк дозволяє встановлювати з'єднання з BLE-пристроями та обмінюватися даними з ними;
- здатність обробки отриманих даних та надсилання їх на BLE-пристрій;
- можливість взаємодії з різними станами BLE, такими як активний режим та режим очікування для оптимізації витрат енергії.

Можливості BLE:

- енергоефективність: BLE є енергоефективним протоколом, що робить його ідеальним для сценаріїв, де важлива тривалість роботи на батареї, таких як у сфері децентралізованого керування опаленням.
- BLE може ефективно працювати в режимі очікування, що важливо для систем опалення, які постійно обмінюються великою кількістю даних між собою.
- BLE дозволяє дистанційно керувати пристроями, що важливо для системи опалення, яка може бути розподілена по різних місцях.
- з легкістю встановлюється з'єднання між пристроями, а також легко взаємодіє з іншими пристроями, що важливо для децентралізованих систем.

CoreBluetooth стає ключовим інструментом для реалізації зв'язку з системами децентралізованого керування опаленням, забезпечуючи ефективну та низькозатратну взаємодію між мобільним додатком і пристроями, такими як мікроконтролери та сенсорами температури. [58]

Пошук BLE пристроїв та відображення знайдених у окремій таблиці зображено на рисунку 4.15.

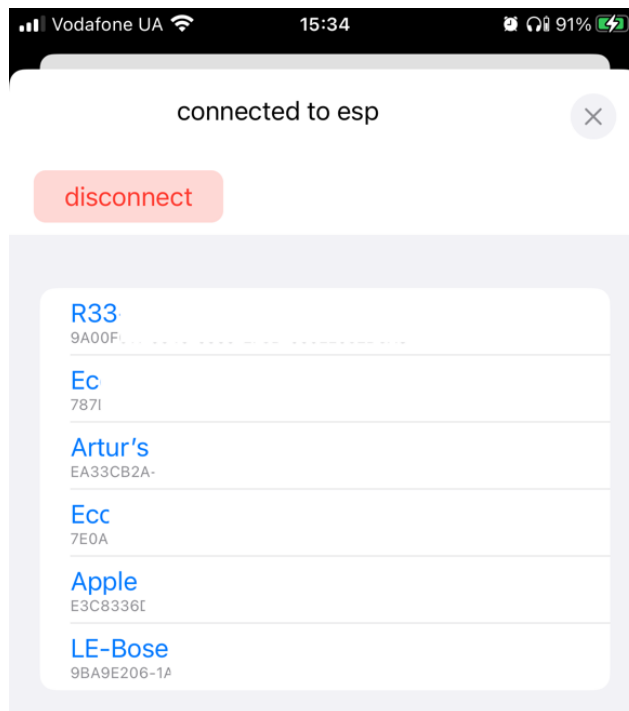


Рисунок 4.15 – відображення знайдених BLE пристроїв

Приєднання до вибраного пристрою-мікроконтролеру зображено на рисунку 4.16.

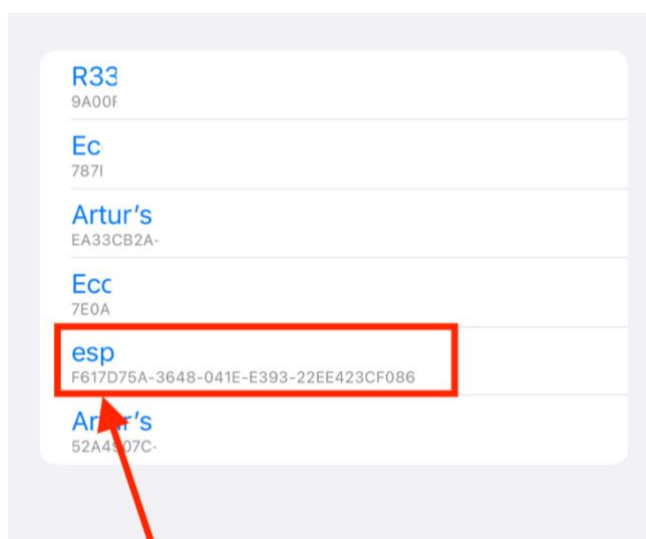


Рисунок 4.16 – вибір та приєднання до мікроконтролеру

Передача інформації до мікроконтролеру відбувається при кліку на кнопки збільшення або зменшення температури зображено на рисунку 4.17



Рисунок 4.17 – відображення кнопок зміни температури на мікроконтролері

Відображення отриманої інформації з мікроконтролеру - температура що збережена на мікроконтролері/сила сигналу у dB та поточна температура з датчику температури зображено на рисунку 4.18.

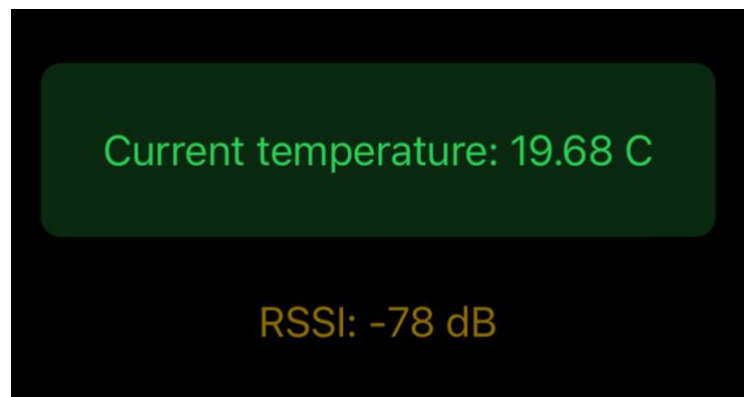


Рисунок 4.18 – відображення отриманої інформації з мікроконтролеру

Від'єднання – від'єднання від мікроконтролеру простим дотиком на кнопку disconnect як зображено на рисунку 4.19.

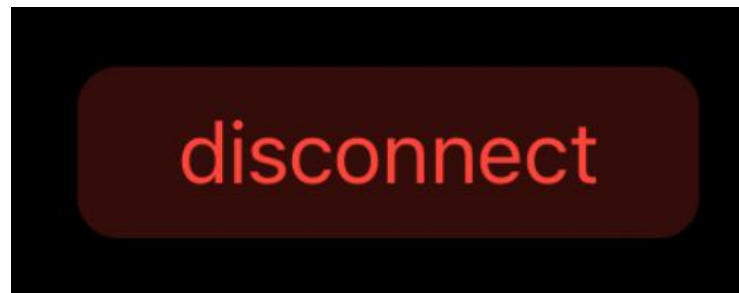


Рисунок 4.19 – кнопка для від’єднання телефону від мікроконтролеру

4.5 Визначення тестування мобільного застосунку

Як обговорювалося раніше, iOS обмежена лише обладнанням Apple або пристроями Apple. Це справді велике полегшення. Однак існує безліч пристроїв Apple та їх версій, які підтримують iOS. [68]

Підсумок полягає в тому, що Apple має закриту систему, на відміну від Android, яка є відкритою системою. Релізи ОС або пристроїв добре сплановані.

Це додаткова перевага, оскільки:

- розмір пристроїв, які доступні або будуть випущені, є фіксованим, і як спеціаліст з контролю якості ми повинні мати дуже чітке уявлення про те, які пристрої вийшли з ринку. Контролю якості стає легко вибрати тестовий стенд/симулятор для тестування;
- як і для пристроїв, нам не потрібно проводити глибокий аналіз ОС, оскільки це закрита система, тому для визначення випробувального стенду для тестування ОС потрібно менше часу (і зусиль);
- Apple має великий вибір власних інструментів автоматизації, хоча їх трохи складно освоїти;
- Apple має суворі вказівки щодо подання заявки, це є великою підмогою, а не отриманням відмови після подання та хорошими шансами на успіх, на відміну від інших ОС, де немає суворих вказівок.

- функціональність пристрою та самої ОС є фіксованою та простою, отже, це зменшує шанси втратити способи роботи програми. В iOS немає способу примусово зупинити програму, тоді як ми можемо вбити та примусово зупинити програми на android. Таким чином тут зменшено складності для тестування.

Це деякі з переваг, які ми отримуємо від продуктів Apple, але не обов'язково, що це переваги кожного продукту чи програми. У той час як для додатків, розроблених на крос-платформі, з iOS важко працювати.

4.5.1 Тестування функціональності

У процесі тестування функціональності мобільного додатку проводиться ретельна перевірка ключових аспектів його роботи. Основний акцент робиться на ефективності та надійності реалізації функціоналу, який забезпечує зручність взаємодії для користувача. [72]

Під час пошуку пристроїв, система відображає їхні імена та унікальні ідентифікатори без спотворень чи викривлень. Приєднання до пристроїв відбувається з мінімальною затримкою, що визначено як припустимий час для повного обміну даними та встановлення зв'язку.

Система автоматично виявляє та від'єднується в разі слабкого сигналу, забезпечуючи надійну роботу навіть в умовах невеликої доступності сигналу. При натисканні кнопок "up" або "down" генерується запит до мікроконтролера, який в свою чергу відправляє відповідь на мобільний пристрій. Отримана інформація, що включає змінену та збережену температуру на мікроконтролері, коректно відображається на екрані телефону.

Всі функції додатку відповідають визначеним властивостям та виконуються без помилок, що гарантує коректну та надійну роботу додатку в реальних умовах використання.

4.5.2 Тестування продуктивності

Тестування продуктивності спрямоване на оцінку та забезпечення ефективності додатку в різних умовах навантаження. Цей процес дозволяє визначити стійкість додатку та забезпечити його оптимальну продуктивність під час роботи в реальних умовах. [72]

Визначення часу, який система витрачає на обробку запитів користувача та надання відповіді у різних станах може варіюватись від 7.5мс до 4х секунд.

Загальної швидкодія додатку під час взаємодії з різними функціональними частинами не викликає великих навантажень на систему і є звично як для SwiftUI основаних застосунків.

Визначення стійкості та рівня витрат енергії додатку під час тривалої роботи та взаємодії з іншими системними ресурсами показало що система не викидає з пам'яті даний застосунок, що показує відсутність memory leak, низьке споживання ОЗУ та енергії, через не велику частоту оновлення даних з мікроконтролеру – кожних 1000мс.

Тестування додатку під час одночасної роботи з великою кількістю користувачів для оцінки його масштабованості показало, що якщо мікроконтролер приєднано до застосунку однієї персони, інша не зможе приєднатись та знайти мікроконтролер до моменту поки перши не від'єднається.

4.6 Висновок до розділу 4

У результаті проведених робіт було розроблено мобільний додаток для управління комплексною системою опалення. Розробка мобільного додатку для керування комплексною системою опалення є складним і багатогранним завданням, яке вимагає аналізу потреб користувачів, розробки ефективного інтерфейсу користувача та реалізації надійного та функціонального коду. У процесі розробки даного додатку було використано сучасні інструменти та

підходи, що дозволило забезпечити високу якість та ефективність додатку.

Додаток забезпечує наступні функції:

- пошук та відображення доступних пристроїв для керування;
- зміна заданої температури за допомогою інтуїтивних кнопок;
- моніторинг та відображення поточної температури;
- відображення рівня сигналу (RSSI) для отримання у режимі реального часу якості з'єднання;
- можливість відключення/з'єднання.

Додаток розроблено з використанням технології Bluetooth Low Energy (BLE), яка забезпечує низьке енергоспоживання та ефективну передачу даних між мобільним пристроєм та ESP32.

Інтерфейс користувача розроблено з урахуванням принципів простоти, надійності та естетичності.

Додаток протестовано на пристроях з iOS 15 та вище.

Застосування розробленого додатку дозволить користувачам поліпшити комфорт та ефективність роботи систем опалення.

ВИСНОВКИ

На основі проведеного дослідження та розробок, була створена комплексна система децентралізованого опалення, яка включає в себе апаратну частину, програмне забезпечення та мобільний додаток для зручного управління.

Висновки з різних аспектів роботи системи:

1. Ефективність та переваги – система продемонструвала високу ефективність та надійність в роботі. Зокрема, апаратна частина на базі мікроконтролерів ESP32 та BLE MESH технології забезпечила стабільне регулювання температури в системі опалення. Порівняно з традиційними системами, розроблена система має кілька переваг:

- простота встановлення та налаштування – інтуїтивний інтерфейс додатку дозволяє користувачам швидко встановити та налаштувати систему;

- децентралізована архітектура – використання BLE MESH технології дозволяє створювати розподілені мережі, уникнути однозначної точки відмови та підвищити надійність системи;

- адаптація до змін зовнішніх умов – система може ефективно працювати в різних умовах, адаптуючись до змін температури та інших параметрів оточуючого середовища;

- можлива енергоощадність – впровадження системи може призвести до оптимізації витрат енергії та зменшення експлуатаційних витрат.

2. Апаратна частина – апаратна частина системи, заснована на мікроконтролері ESP32 та датчику температури, продемонструвала надійну роботу. Мікроконтролер взаємодіє з датчиком, створює та підтримує мережу MESH, а також відповідає за підтримку заданої температури. Додатково, мобільний додаток розроблено з використанням сучасних технологій, забезпечуючи зручний та інтуїтивно зрозумілий інтерфейс.

3. Потенціал та перспективи – розроблений додаток має значний потенціал у полі систем опалення. Він не лише забезпечує зручне керування

опалювальною системою, а й може сприяти ефективному використанню енергії та створенню комфортного мікроклімату для користувачів. Для подальшого розвитку та вдосконалення можна розглядати додавання нових функцій:

- підтримка додаткових датчиків – додаткові сенсори, такі як вологості чи освітлення, можуть збагатити інформацію та покращити функціональність системи;

- створення профілів користувачів – можливість зберігання та використання різних налаштувань для різних користувачів чи сценаріїв.

- групове керування пристроями – можливість об'єднувати пристрої в групи та керувати ними одночасно;

- віддалене керування через Інтернет – забезпечення можливості віддаленого керування системою опалення через Інтернет;

- попередження про можливі проблеми – введення системи сповіщень та попереджень щодо можливих неполадок або несправностей;

- підказки та рекомендації – додавання інтерактивних підказок та рекомендацій для оптимального використання системи опалення.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Публікація роботи для конференції – URL:
<https://conferences.vntu.edu.ua/index.php/all-fksa/all-fksa-2024/paper/view/19655/16323>
2. В. М. Дубовой, М. С. Юхимчук. Децентралізоване координаційне керування розподіленими кібер-фізичними системами з неперервними об'єктами : монографія. Вінниця : ВНТУ, 2022. 230 с.
3. Yukhymchuk M., Dubovoi V., Kovtun V. «Decentralized Coordination of Temperature Control in Multiarea Premises». Complexity. 2022. ID 2588364, 18 pages – URL: <https://doi.org/10.1155/2022/2588364>.
4. F. Bauman, H. Zhang, E. Arens et al., Advanced Integrated Systems Technology Development: Personal Comfort Systems and Radiant Slab Systems, UC Berkeley: Center for the Built Environment, Berkeley, CA, USA, 2015.
5. Friendly Intelligent Energy Management System for Existing Residential Buildings, <http://www.fiemser.eu/>
6. Energy WATCHer – URL:
http://cordis.europa.eu/pub/fp7/ict/docs/sustainable-growth/fp7-beywatch_en.pdf
7. Smart control of demand for consumption and supply to enable balanced, energy-positive buildings and neighborhoods – URL:
<https://www.fp7-smartcode.eu/>
8. Center for Advanced Studies in Adaptive Systems – URL: <http://casas.wsu.edu/>
9. Smart Home, Smart HEMS, Smart heating: An Overview of the Latest Products and Trends: – URL:
https://www.researchgate.net/publication/320083433_Smart_Home_Smart_HEMS_Smart_heating_An_Overview_of_the_Latest_Products_and_Trends

10. S. K. Das, D. J. Cook, A. Battacharya, E. O. Heierman, and T.-Y. Lin, "The role of prediction algorithms in the MavHome smart home architecture," *IEEE Wirel. Commun.*, vol. 9, no. 6, pp. 77–84, Dec. 2002.
11. K. Gopalratnam and D. J. Cook, "Active LeZi: An incremental parsing algorithm for sequential prediction," *Int. J. Artif. Intell. Tools*, vol. 13, no. 04, pp. 917–929, 2004.
12. S. P. Rao and D. J. Cook, "Predicting inhabitant action using action and task models with application to smart homes," *Int. J. Artif. Intell. Tools*, vol. 13, no. 01, pp. 81–99, 2004.
13. E. O. Heierman and D. J. Cook, "Improving home automation by discovering regularly occurring device usage patterns," in *Data Mining, 2003. ICDM 2003. Third IEEE International Conference on*, 2003, pp. 537–540.
14. Желих В. М. Особливості опалення виробничих приміщень / В. М. Желих, О. Т. Возняк, Ю. С. Юркевич // *Теорія і практика будівництва. Національний університет «Львівська політехніка»*. – 2007. – № 602. – С. 57–61.
15. Желих В. М. Тепловое состояние технологической зоны животноводческих помещений при использовании инфракрасных нагревателей / В. М. Желих, Н. М. Фиалко, В. Г. Прокопов // *Промышленная теплотехника*. – 2011. – № 2. Том 33. – С. 19–23.
16. Bauman, Fred, Hui Zhang, Edward Arens, Paul Raftery, Caroline Karmann, Jingjuan Dove Feng, Yongchao Zhai, Darryl Dickerhoff, Stefano Schiavon, and Xiang Zhou. 2015. "Advanced Integrated Systems Technology Development: Personal Comfort Systems and Radiant Slab Systems." *EScholarship*. <https://escholarship.org/uc/item/88p8v7zb.pdf>.
17. Zhang, Hui, Edward Arens, and Yongchao Zhai. 2015. "A Review of the Corrective Power of Personal Comfort Systems in Non-Neutral Ambient Environments." *Building and Environment* 91 (September): 15–41. doi:10.1016/j.buildenv.2015.03.013.

18. Shahzad, Sally, John Kaiser Calautit, Katrina Calautit, Ben Hughes, and Angelo I. Aquino. 2018. "Advanced Personal Comfort System (APCS) for the Workplace: A Review and Case Study." *Energy and Buildings* 173 (August): 689–709. doi:10.1016/j.enbuild.2018.02.008.
19. Rupp, Ricardo Forgiarini, Natalia Giraldo Vásquez, and Roberto Lamberts. 2015. "A Review of Human Thermal Comfort in the Built Environment." *Energy and Buildings* 105 (October): 178–205. doi:10.1016/j.enbuild.2015.07.047.
20. Veselý, Michal, and Wim Zeiler. 2014. "Personalized Conditioning and Its Impact on Thermal Comfort and Energy Performance – A Review." *Renewable and Sustainable Energy Reviews* 34 (June): 401–408. doi:10.1016/j.rser.2014.03.024.
21. Schiavon, Stefano, and Arsen K. Melikov. 2008. "Energy Saving and Improved Comfort by Increased Air Movement." *Energy and Buildings* 40 (10): 1954–1960. doi:10.1016/j.enbuild.2008.05.001.
22. H. Yan and Y. Han, "Decentralized adaptive multi-dimensional Taylor network tracking control for a class of large-scale stochastic nonlinear systems," in *International Journal of Adaptive Control and Signal Processing*, vol. 33, no. 4, pp. 664–683, Wiley, 2019 – URL: <https://doi.org/10.1002/acs.2978>.
23. Yang, Bin, Chandra Sekhar, and Arsen K. Melikov. 2010. "Ceiling Mounted Personalized Ventilation System in Hot and Humid Climate—An Energy Analysis." *Energy and Buildings* 42 (12): 2304–2308. doi:10.1016/j.enbuild.2010.07.022.
24. Watanabe, Shinichi, Toshimichi Shimomura, and Hironori Miyazaki. 2009. "Thermal Evaluation of a Chair with Fans as an Individually Controlled System." *Building and Environment* 44 (7): 1392–1398. doi:10.1016/j.buildenv.2008.05.016.
25. P. Yang, Y. Xia, M. Yu, W. Wei, and Y. Peng, "A decentralized coordination control method for parallel bidirectional power converters in a hybrid AC–DC

- microgrid,” in IEEE Transactions on Industrial Electronics, vol. 65, no. 8, pp. 6217–6228, Institute of Electrical and Electronics Engineers (IEEE), 2018.
26. Andersen, Michael, Gabe Fierro, Sam Kumar, Joyce Kim, Edward Arens, Hui Zhang, Paul Raftery, and David Culler. 2016. “Well-Connected Microzones for Increased Building Efficiency and Occupant Comfort.” Proceedings of ACEEE Summer Study, 2016, American Council for an Energy-Efficient Economy (ACEEE), August. – URL:
http://aceee.org/files/proceedings/2016/data/papers/12_1146.pdf
27. Paolo Zampognaro · Giovanni Paragliola · Vincenzo Falanga Definition of an FHIR-based multiprotocol IoT home gateway to support the dynamic plug of new devices within instrumented environments. Journal of Reliable Intelligent Environments 2021 – URL: <https://doi.org/10.1007/s40860-021-00161-2>
28. The INTERNET of THINGS with ESP32 – URL: <http://esp32.net>
29. Термоконтроль використовуючи термодатчики – URL: <https://uk.top-home-tips.com/9769687-thermostat-for-heating-boiler-principle-of-operation-types-connection-diagrams>
30. ESP32 BLE peripheral for iOS development – URL:
<https://www.cuvenx.com/post/ble-peripheral-ios-development-part1>
31. Принципова електрична схема – URL:
https://uk.wikipedia.org/wiki/Принципова_електрична_схема
32. Довідник MOSFET – URL:
<https://alltransistors.com/en/mosfet/transistor.php?transistor=3466>
33. Як відрізнити структурну схему від функціональної – URL:
<https://ukrpublic.com/navchannia/yak-vidrizniti-strukturnu-skhemu-vid-funksionalnoji.html>
34. ESP-WIFI-MESH – URL: <https://docs.espressif.com/projects/espressif/en/stable/esp32/api-guides/esp-wifi-mesh.html>
35. Bluetooth documentation – URL:
<https://www.bluetooth.com/blog/a-developers-guide-to-bluetooth/>

36.DS18B20 sensor – URL:

[https://wiki.keyestudio.com/Ks0316_Keyestudio_DS18b20_Temperature_Detector_Sensor_\(Black_and_Eco-friendly\)](https://wiki.keyestudio.com/Ks0316_Keyestudio_DS18b20_Temperature_Detector_Sensor_(Black_and_Eco-friendly))

37.D. Blum, Z. Wang, C. Weyandt et al., “Field demonstration and implementation analysis of model predictive control in an office HVAC system,” in *Applied Energy*, vol. 318, Elsevier BV, Article ID 119104, 2022.

38.H. Yan and Y. Han, “Decentralized adaptive multi-dimensional Taylor network tracking control for a class of large-scale stochastic nonlinear systems,” in *International Journal of Adaptive Control and Signal Processing*, vol. 33, no. 4, pp. 664–683, Wiley, 2019 – URL: <https://doi.org/10.1002/acs.2978>.

39.BLE documentation – URL:

<https://www.bluetooth.com/bluetooth-resources/intro-to-bluetooth-low-energy/>

40.Bluetooth LE & Bluetooth on espressif – URL:

<https://docs.espressif.com/projects/esp-faq/en/latest/software-framework/ble-bt.html>

41.OneWire documentation – URL:

<https://www.arduino.cc/reference/en/libraries/onewire/>

42.DallasTemperature documentation – URL:

<https://www.arduino-libraries.info/libraries/dallas-temperature>

43.J. H. van Schuppen and T. Villa, Eds. “Coordination control of distributed systems,” in *Lecture Notes in Control and Information Sciences*, Springer International Publishing, 2015.

44.J. F. Nicol and S. Roaf, “Rethinking thermal comfort,” in *Building Research & Information*, vol. 45, no. 7, pp. 711–716, Informa UK Limited, 2017.

45.Здоренко, В. Г., Д. Р. Гуртова, and Є. В. Папіровий. "Дослідження системи автоматичного керування потужністю електричних нагрівачів." *Мехатронні системи: інновації та інжиніринг*. Київський національний університет технологій та дизайну, 2020.

- 46.P. Antoniadou and A. M. Papadopoulos, “Occupants’ thermal comfort: state of the art and the prospects of personalized assessment in office buildings,” in *Energy and Buildings*, vol. 153, pp. 136–149, Elsevier BV, 2017.
- 47.M. Wang, E. Wolfe, D. Ghosh et al., “Localized cooling for human comfort,” in *SAE International Journal of Passenger Cars - Mechanical Systems*, vol. 7, no. 2, pp. 755–768, SAE International, 2014.
- 48.H. Amai, S. Tanabe, T. Akimoto, and T. Genma, “Thermal sensation and comfort with different task conditioning systems,” in *Building and Environment*, vol. 42, no. 12, pp. 3955–3964, Elsevier BV, 2007.
- 49.Ioannou and L. C. Itard, “Energy performance and comfort in residential buildings: sensitivity for building parameters and occupancy,” *Energy and Buildings*, vol. 92, pp. 216–233, 2015.
- 50.L. J. Underhill, W. S. Dols, S. K. Lee, M. P. Fabian, and J. I. Levy, “Quantifying the impact of housing interventions on indoor air quality and energy consumption using coupled simulation models,” in *Journal of Exposure Science and Environmental Epidemiology*, vol. 30, no. 3, pp. 436–447, Springer Science and Business Media LLC, 2020.
- 51.Anastasiadi and A. I. Dounis, “Co-simulation of fuzzy control in buildings and the HVAC system using BCVTB,” in *Advances in Building Energy Research*, vol. 12, no. 2, pp. 195–216, Informa UK Limited, 2017.
- 52.ESP32-DevKitC Getting Started Guide – URL:
<https://docs.espressif.com/projects/esp-idf/en/latest/esp32/hw-reference/esp32/get-started-devkitc.html>
- 53.Framework SwiftUI – URL:
<https://developer.apple.com/documentation/swiftui/>
- 54.Transferring Data Between Bluetooth Low Energy Devices – URL:
https://developer.apple.com/documentation/corebluetooth/transferring_data_between_bluetooth_low_energy_devices
- 55.ЕМЕРДЖЕНТНІСТЬ – ФЕНОМЕН СКЛАДНИХ СИСТЕМ – URL:

- <http://journals.khnu.km.ua/vestnik/wp-content/uploads/2021/01/14-13.pdf>
56. What are complex systems? – URL: <https://uwaterloo.ca/complexity-innovation/about/what-are-complex-systems>
57. EasyEDA – URL:
<https://easyeda.com>
58. About Core Bluetooth – URL:
https://developer.apple.com/library/archive/documentation/NetworkingInternetWeb/Conceptual/CoreBluetooth_concepts/AboutCoreBluetooth/Introduction.html
59. Core Bluetooth (BLE) – Swift – URL:
<https://medium.com/macoclock/core-bluetooth-ble-swift-d2e7b84ea98e>
60. Swift – URL:
<https://developer.apple.com/swift/>
61. Xcode documentation – URL:
<https://developer.apple.com/documentation/xcode>
62. Сучасні системи опалення – URL:
<https://energy-safe.com.ua/suchasni-sistemi-opalennya-povitryanij-obigriv-vid-kamina/>
63. Опалення для будинку. Порівняння різних видів опалювальних систем – URL: <https://teploceramic.ua/ua/sistemy-otopleniya-doma-sravnenie-ch-1.html>
64. Водяне опалення: переваги та недоліки – URL:
<https://mycond-heatpump.com.ua/ua/vodyane-opalennya-perevagi-ta-nedoliki/>
65. Що таке тепловий насос і як він працює – URL:
<https://idm.ua/publications/chto-takoe-teplovoy-nasos-i-kak-on-rabotaet/>
66. "Розробка мобільних додатків", автор: Річард М. Грін, видавництво: "Вільямс"
67. "Мобільна розробка: від ідеї до готового додатку", автор: Річард М. Грін, видавництво: "Вільямс"
68. Apple Developer – URL: <https://developer.apple.com/>

69. "Аналіз потреб користувачів у мобільних додатках", автори: Р. М. Грін, М. С. Фішер, видавництво: "Вільямс".
70. "Розробка ефективного інтерфейсу користувача для мобільних додатків", автори: Р. М. Грін, К. Х. Вонг, видавництво: "Вільямс".
71. "Реалізація функціональності в мобільних додатках", автор: Р. М. Грін, видавництво: "Вільямс".
72. "Тестування мобільних додатків", автор: Р. М. Грін, видавництво: "Вільямс".
73. Сайт JLCPCB – URL: <https://jlcpcb.com/>
74. "Мікроконтролери: основи та застосування", автор: В. С. Гурін, видавництво: "Підручники та посібники", 2005
75. SparkFun – URL: <https://www.sparkfun.com/>
76. Digi-Key: – URL: <https://www.digikey.com/>
77. "Децентралізовані системи: принципи та методи", автори: В. І. Лисенко, М. І. Литвин, видавництво: "Наукова думка", 2001
78. "The Wave Algorithm", by Douglas B. West, 2016.
79. "Modern Control Engineering" by Katsuhiko Ogata
80. On decentralized connectivity maintenance for mobile robotic systems – URL: https://www.academia.edu/36822838/On_decentralized_connectivity_maintenance_for_mobile_robotic_systems
81. Environmental Sensors Market Size & Share Analysis - Growth Trends & Forecasts (2023 - 2028) – URL: <https://www.mordorintelligence.com/industry-reports/environmental-sensors-market>
82. ArduinoJSON – URL: <https://arduinojson.org/>
83. C++ Programming Language – URL: <https://www.cs.uah.edu/~rcoleman/Common/History/HistoryOfCPP.html>
84. What is Arduino? – URL: <https://www.arduino.cc/en/Guide/Introduction>
85. Arduino IDE 1 – URL: <https://docs.arduino.cc/software/ide-v1>

86.Painless MESH – URL:

<https://www.arduino.cc/reference/en/libraries/painless-mesh/>

87.SwiftUI documentation – URL:

<https://developer.apple.com/documentation/swiftui/>

88.Публікація роботи для міжнародної конференції. – URL:

<https://eu-conf.com/wp-content/uploads/2023/11/MODERN-WAYS-OF-DEVELOPMENT-OF-SCIENCE-AND-THE-LATEST-THEORIES.pdf>

Додатки

Додаток А – Технічне завдання
(обов'язковий)
ВНТУ

ЗАТВЕРДЖЕНО

Зав. кафедри АІТ ВНТУ,

д.т.н., проф.


Підпис Олег БІСІКАЛО
Ім'я ПРІЗВИЩЕ

«12» лютого 2023 р.


ТЕХНІЧНЕ ЗАВДАННЯ

на виконання магістерської кваліфікаційної роботи


Комплексна система децентралізованого керування опаленням

08-33.МКР.012.02.000 ТЗ

Студент групи ЗАКІТ-22м


Підпис Артур ПИЛЯВЕЦЬ
Ім'я ПРІЗВИЩЕ

Керівник проф., д.т.н.


Підпис Володимир ДУБОВОЙ
Ім'я ПРІЗВИЩЕ

Вінниця 2023

1. Назва та галузь застосування

1.1. Назва – Комплексна система децентралізованого керування опаленням.

1.2 Галузь застосування – опалення.

2. Підстава для проведення розробки.

Тема магістерської кваліфікаційної роботи затверджена наказом по ВНТУ №247 від 18.09.2023 р.

3. Мета та призначення розробки.

Метою магістерської кваліфікаційної роботи є забезпечення комфорту, зменшення навантаження на мережу та підвищення ефективності систем опалення.

4. Джерела розробки.

Магістерська кваліфікаційна робота виконується вперше. В ході проведення розробки повинні використовуватись такі документи:

1. Yukhymchuk M., Dubovoi V., Kovtun V. «Decentralized Coordination of Temperature Control in Multiarea Premises». Complexity. 2022. ID 2588364, 18 pages, <https://doi.org/10.1155/2022/2588364>.
2. Center for Advanced Studies in Adaptive Systems, Smart Home, Smart heating: An Overview of the Latest Products and Trends: https://www.researchgate.net/publication/320083433_Smart_Home_Smart_HEMS_Smart_heating_An_Overview_of_the_Latest_Products_and_Trends
3. F. Bauman, H. Zhang, E. Arens et al., Advanced Integrated Systems Technology Development: Personal Comfort Systems and Radiant Slab Systems, UC Berkeley: Center for the Built Environment, Berkeley, CA, USA, 2015.

5. Вихідні дані: загальна інформація про розподіленні децентралізовані системи опалювання їх будова, алгоритми, аналоги, програмне забезпечення для мікроконтролеру та апаратна частина, мобільний застосунок, діапазон керування температурою 15-30 °С, керування нагрівачем потужністю до 2200W.

6. Вимоги до розробки.

5.1. Перелік головних функцій:

- контроль температури у визначеній зоні;
- отримування показників температури від подібних контролерів у суміжних зонах;
- керування мікроконтролером з смартфона ;
- внесення правок у коригування власної температури на основі зібраних даних з контролерів у зоні дії мережі.

5.2. Основні технічні вимоги до розробки.

5.2.1. Вимоги до програмної платформи:

- MacOS 13+;
- Arduino IDE;
- Xcode 15+;
- iOS 16+.

5.2.2. Умови експлуатації системи:

- робота на мобільних додатках;
- можливість цілодобового функціонування системи;
- дані оновлюються і є актуальними.

6. Стадії та етапи розробки.

6.1 Пояснювальна записка:

1. Аналіз методів, принципів, підходів і засобів реалізації задачі автоматизації процесами в об'єкті управління відповідно до теми дипломної роботи. Постановка задач дослідження « 25 » вересня 2023 р.
2. Удосконалення технології прийняття рішень при автоматизації об'єкту управління « 28 » вересня 2023 р.
3. Визначення технічних характеристик системи « 1 » жовтня 2023 р.
4. Розробка апаратної частини системи « 10 » жовтня 2023 р.
5. Розробка програмного забезпечення системи « 15 » листопада 2023 р.

6.2 Графічні матеріали:

1. Розробка моделі системи « 18 » листопада 2023 р.
2. Тестування апаратної частини « 20 » листопада 2023 р.
3. Тестування програмного забезпечення « 25 » листопада 2023 р.

7. Порядок контролю і приймання.

7.1. Хід виконання роботи контролюється керівником роботи. Рубіжний контроль провести до 28 11 2023 р.

7.2. Атестація МКР здійснюється на попередньому захисті. Попередній захист магістерської кваліфікаційної роботи провести до 16 12 2023 р.

7.3. Підсумкове рішення щодо оцінки якості виконання роботи приймається на засіданні ЕК. Захист магістерської кваліфікаційної роботи провести до 20 12 2023 р.

Додаток Б – Ілюстративна частина
(обов'язковий)

ІЛЮСТРАТИВНА ЧАСТИНА

Комплексна система децентралізованого керування опаленням

ДИПЛОМНА РОБОТА

(ОСВІТНЬО-КВАЛІФІКАЦІЙНИЙ РІВЕНЬ МАГІСТР)

СТУДЕНТА ГРУПИ ЗАКІТ-22М

ПИЛЯВЦЯ АРТУРА ІГОРОВИЧА

КЕРІВНИК РОБОТИ

ДУБОВОЙ ВОЛОДИМИР МИХАЙЛОВИЧ

Д.Т.Н. ПРОФЕСОР КАФЕДРИ КСУ

Рисунок Б.1 – Титулка презентації

Актуальність, мета та практична цінність

Метою магістерської кваліфікаційної роботи є підвищення надійності, зручності використання, зменшення навантаження на мережу та підвищення ефективності на основі опалювальних приладів – мобільних нагрівачів.

Практична цінність. Дослідження має практичну цінність в умовах сучасної ситуації в Україні, де енергоощадні рішення для опалення стають особливо важливими через обмеженість паливно-енергетичних ресурсів, спричинену військовим конфліктом. Розроблена система може сприяти зменшенню енерговитрат і оптимізації опалювання великих приміщень.

Рисунок Б.2 – Актуальність, мета та практична цінність

Дослідження

1. Проведення аналізу предметної області

- Вивчення сучасних тенденцій та технологій у галузі опалення та енергоефективності.

2. Аналіз конкурентів

- Порівняльний аналіз схожих систем на ринку для визначення переваг та недоліків.

3. Вибір оптимальних рішень

- Обґрунтування вибору ESP32 та інших компонентів для оптимального функціонування системи.

Рисунок Б.3 – Розділ “Дослідження”

Загальний аналіз систем опалення

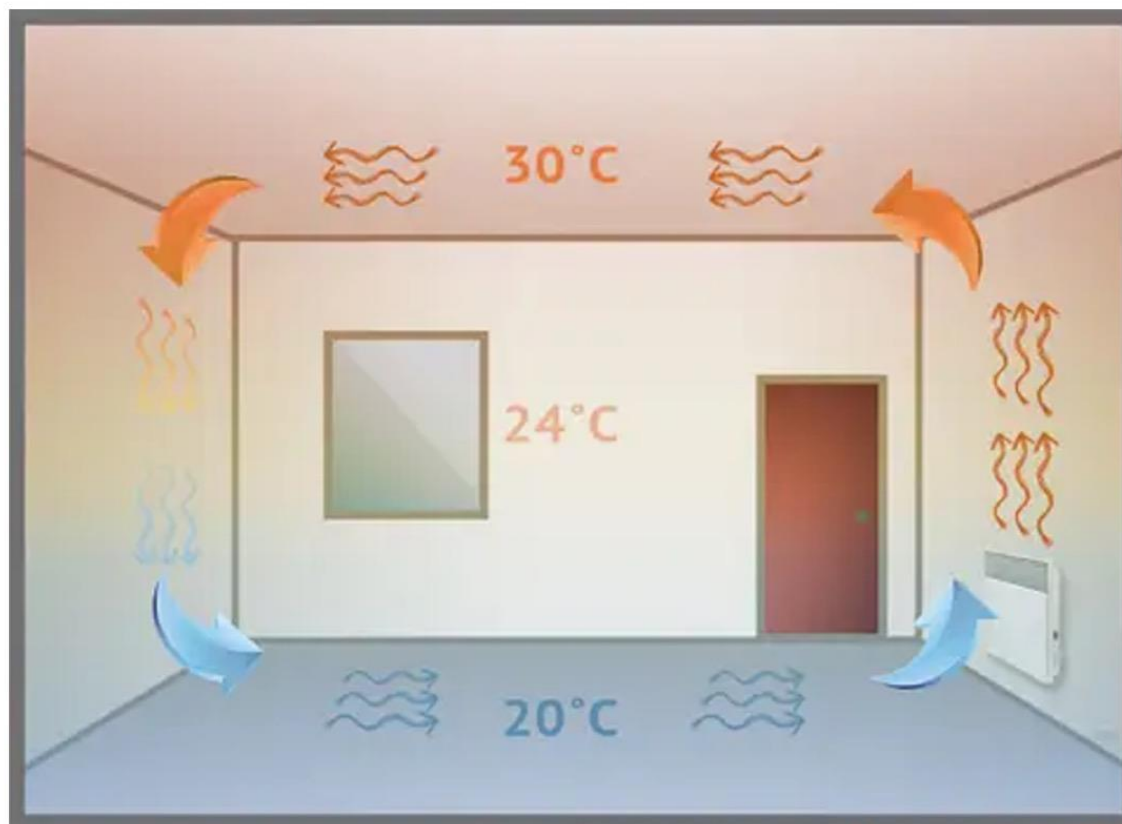


Рисунок Б.4 – Загальний аналіз систем опалення

Моделювання та розробка апаратної частини

1. Моделювання та симуляція системи опалення

- Відтворення різних сценаріїв для тестування ефективності та надійності системи в умовах реального експлуатації на створеному стенді.

2. Розробка апаратної частини

- Відповідно до аналізу та моделювання систем опалення – проведення розробки апаратних компонентів системи керування опаленням.

Рисунок Б.5 – Розділ “Моделювання та розробка апаратної частини”

Розробка плати модулю

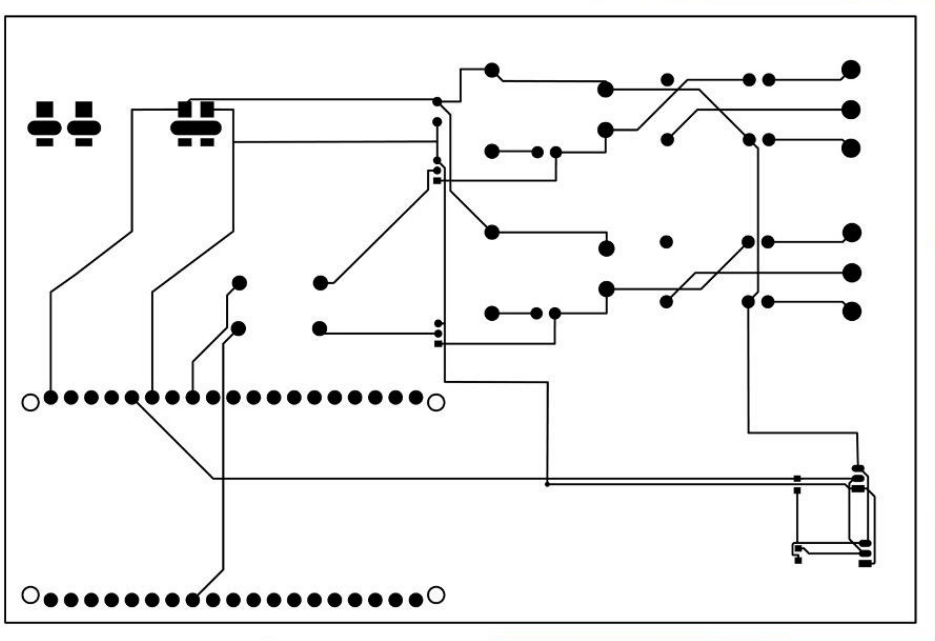
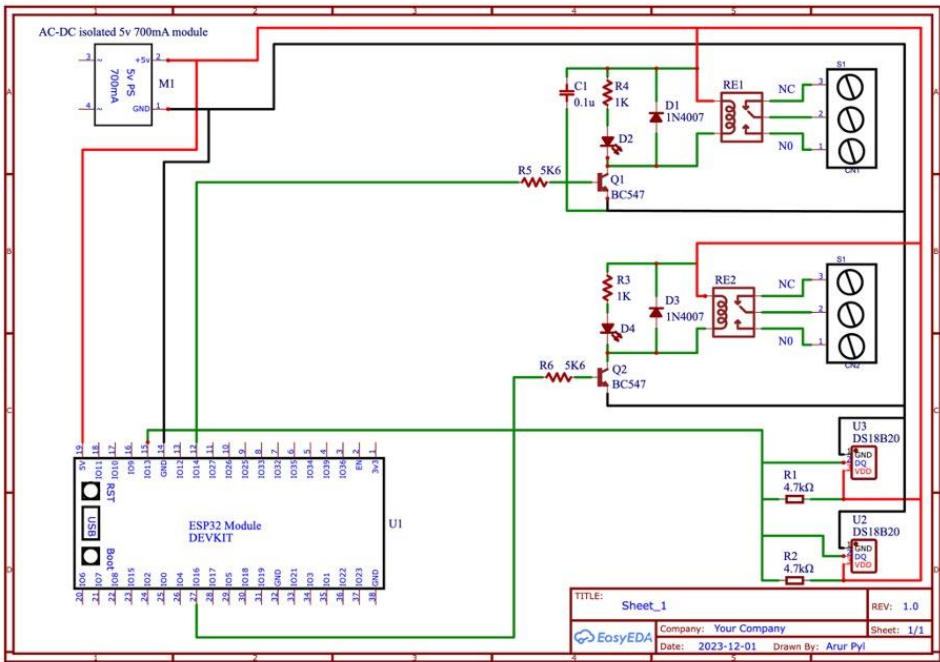


Рисунок Б.6 – Розробка принципової схеми

Розробка та тестування зібраних варіантів



Рисунок Б.7 – Розробка та тестування зібраних варіантів

Програмне забезпечення

1. Розробка програмного забезпечення для контролерів

1. Опис архітектури та функціональності програмного модуля для мікроконтролера ESP32;
2. Використання сучасних технологій, таких як C++ та Arduino IDE, для реалізації програмного забезпечення.

2. Принцип роботи комплексної системи

1. Модуль MESH;
2. Модуль BLE;
3. Модуль керування температурою.

3. Використання хвильового методу

1. Опис та важливість хвильового методу у комплексній системі децентралізованого керування опаленням.

4. Інтеграція з мобільним застосунком

1. Опис взаємодії програмного модуля з розробленим мобільним застосунком через технології BLE.

Рисунок Б.8 – Розділ “Програмне забезпечення”

Принцип роботи

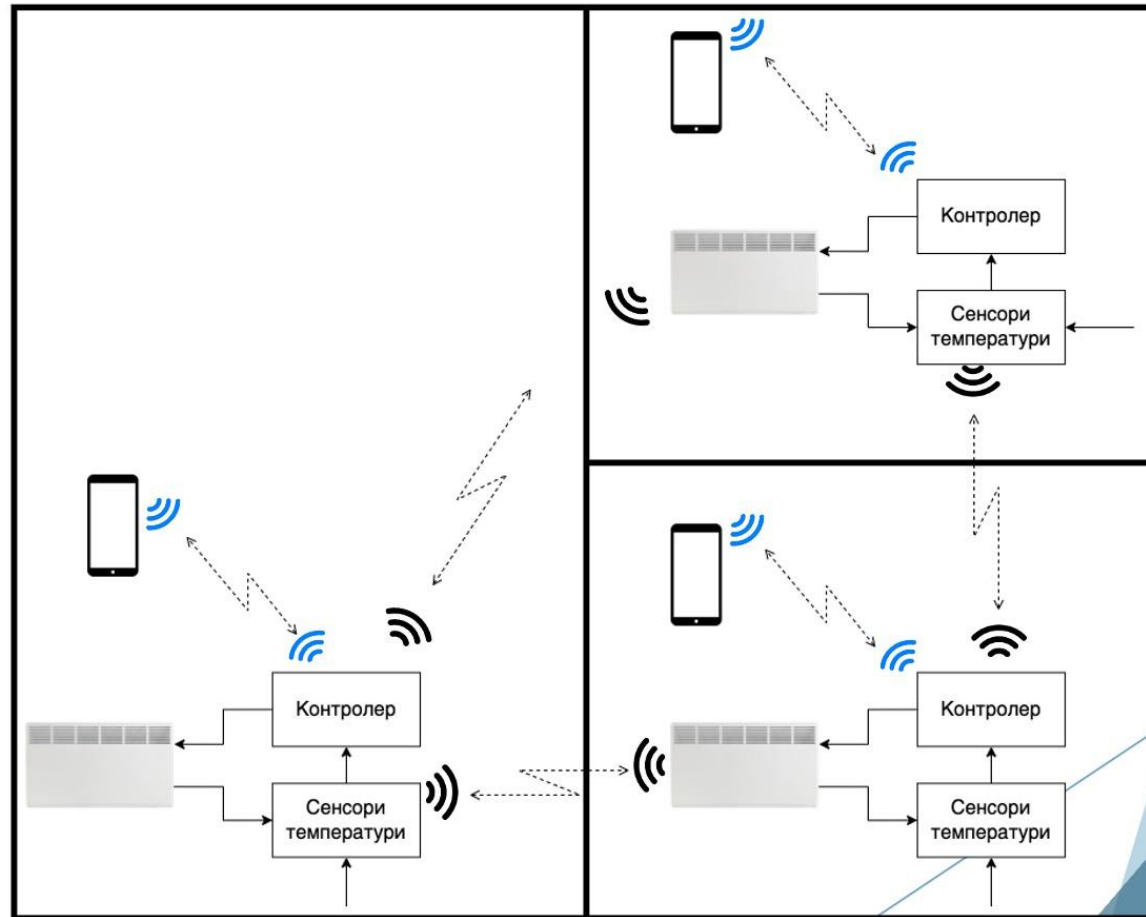


Рисунок Б.9 – Принцип роботи

Використання хвильового методу

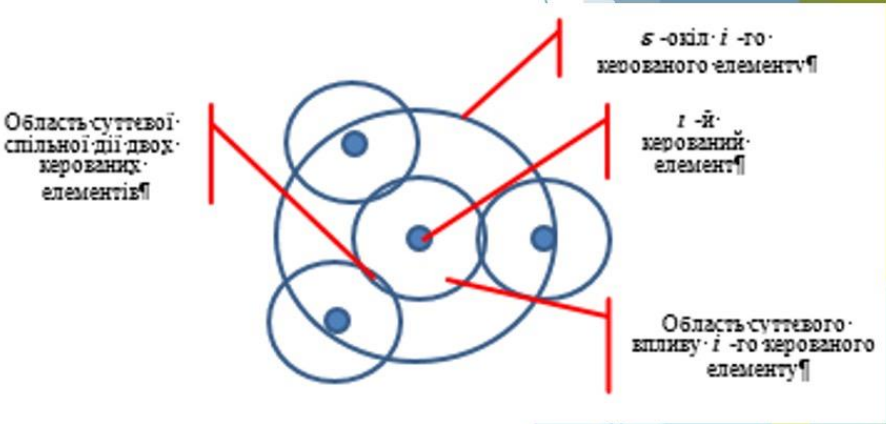
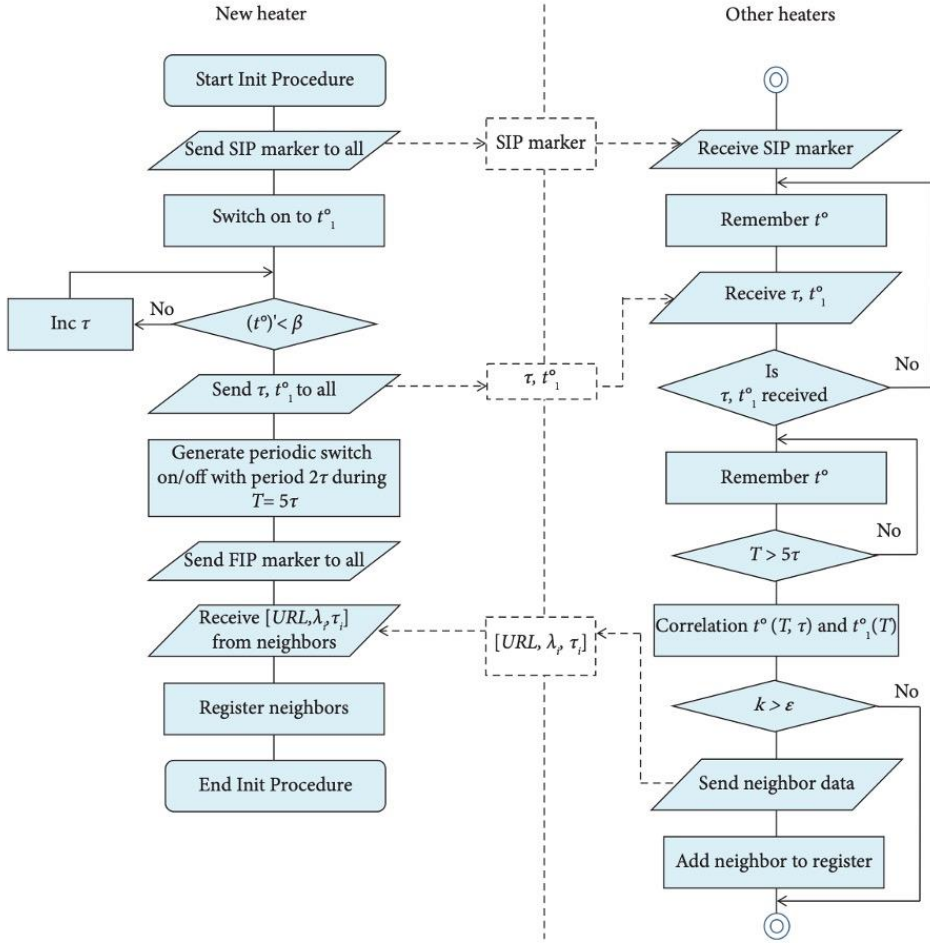


Рисунок Б.10 – Використання хвильового методу

Розробка та інтеграція мобільного застосунку

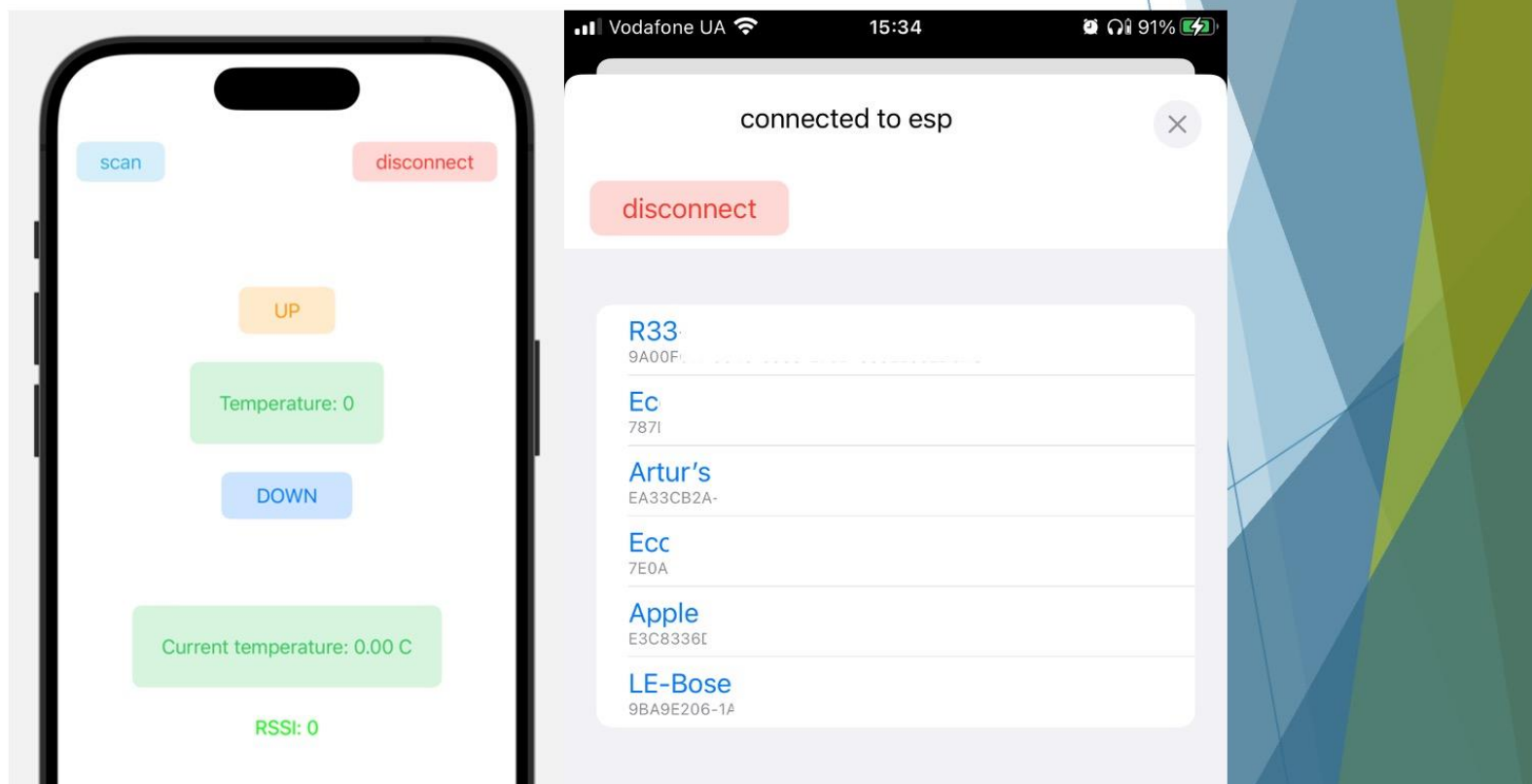


Рисунок Б.11 – Розробка та інтеграція мобільного застосунку

Висновки

- 1. В результаті проведеного дослідження та розробок була створена комплексна система децентралізованого опалення. Система включає в себе апаратну частину, програмне забезпечення та мобільний додаток для зручного управління. Висновки з роботи системи вказують на високу ефективність та переваги порівняно з традиційними системами, такі як простота встановлення, децентралізована архітектура та адаптація до змін зовнішніх умов.**
- 2. Апаратна частина системи, заснована на мікроконтролері ESP32 та BLE MESH технології, демонструє стабільне регулювання температури. Мобільний додаток, розроблений з використанням сучасних технологій, забезпечує зручний та інтуїтивно зрозумілий інтерфейс.**
- 3. Система має значний потенціал у сфері систем опалення, сприяючи ефективному використанню енергії та створенню комфортного мікроклімату.**

Рисунок Б.12 – Висновки

Додаток В – Лістинг програмної частини основної функції налаштувань (обов'язковий)

```

void setup() {
  Serial.begin(115200);
  sensors.begin();

  pinMode(IRL, OUTPUT);
  pinMode(IRL_LED, OUTPUT);
  pinMode(relay_pin_1, OUTPUT);
  pinMode(relay_pin_2, OUTPUT);
  // pinMode(ledPin, OUTPUT);

  mesh.setDebugMsgTypes( ERROR | STARTUP );
  mesh.init( MESH_PREFIX, MESH_PASSWORD, &userScheduler, MESH_PORT );
  mesh.onReceive(&receivedCallback);
  mesh.onNewConnection(&newConnectionCallback);
  mesh.onChangedConnections(&changedConnectionCallback);
  mesh.onNodeTimeAdjusted(&nodeTimeAdjustedCallback);

  userScheduler.addTask(taskSendMessage);
  taskSendMessage.enable();

  SerialBT.begin(bluetoothCellName); // Bluetooth device name
  Serial.println("Ready for pairing");
}

//mesh connections
void newConnectionCallback(uint32_t nodeId) {
  Serial.printf("New Connection, nodeId = %u\n", nodeId);
}
void changedConnectionCallback() {
  Serial.printf("Changed connections\n");
}
void nodeTimeAdjustedCallback(int32_t offset) {
  Serial.printf("Adjusted time %u. Offset = %d\n", mesh.getNodeTime(),offset);
}

```

Додаток Г – Лістинг програмної частини основної циклової функції
(обов'язковий)

```
void loop() {
  mesh.update(); //start mesh network
  //set values to relay(heating/cooling)
  digitalWrite(relay_pin_1, LOW);
  digitalWrite(relay_pin_2, HIGH);
  //upper and lower threshold
  TEMP_THRESHOLD_UPPER = tempFromBL + 1;
  TEMP_THRESHOLD_LOWER = tempFromBL - 1;

  //change polarity of Peltier element (cooling/heating)
  if (currentTemp < tempFromBL) {
    //cooling
    digitalWrite(relay_pin_1, LOW);
    digitalWrite(relay_pin_2, HIGH);
  } else {
    //heating
    digitalWrite(relay_pin_1, HIGH);
    digitalWrite(relay_pin_2, LOW);
  }

  //temperature control
  if (tempFromBL != 0 || currentTemp != 0) {
    if(currentTemp < TEMP_THRESHOLD_LOWER){
      digitalWrite(IRL, HIGH);
      digitalWrite(IRL_LED, HIGH);
    } else if(currentTemp > TEMP_THRESHOLD_UPPER){
      digitalWrite(IRL, LOW);
      digitalWrite(IRL_LED, LOW);
    }
  }
  } else {
    digitalWrite(IRL, LOW);
```



```
digitalWrite(IRL_LED, LOW);
}

//bluetooth loop setup
unsigned long currentMillis = millis();

if (currentMillis - previousMillis >= interval) {
  previousMillis = currentMillis;
  temperatureString = String(currentTemp) + "C ";
  SerialBT.print("My temperature: ");
  SerialBT.println(temperatureString);
  if (node_writings != "") {
    SerialBT.print("Node: ");
    SerialBT.println(node_writings);
    SerialBT.print("Temperature: ");
    SerialBT.println(temp_writings);
  }
}

// Read received messages
if (SerialBT.available()){
  char incomingChar = SerialBT.read();
  if (incomingChar != '\n'){
    message += String(incomingChar);
  }
  else{
    message = "";
  }
  Serial.write(incomingChar);
}
```

Додаток Г – Лістинг програмної частини інтерфейсу користувача у мобільному застосунку
(обов'язковий)

```

var body: some View {
    VStack{
        HStack{
            Button("scan"){ presented.toggle() }.buttonStyle(appButton(bodyColor: .cyan,
paddingHorizontal: 20, paddingVertical: 7)).padding()
            Spacer()
            if isConnected {
                Button("disconnect"){ bluetooth.disconnect()
}.buttonStyle(appButton(bodyColor: .red, paddingHorizontal: 20, paddingVertical: 7)).padding() }}
            if isConnected {
                VStack {
                    Button("UP") {
                        editing = false
                        string = String(Int(response.subdata(in: 2..<3).first ?? 0) + 1)
                        bluetooth.send(Array(string.utf8))
                    }.buttonStyle(appButton(bodyColor: .orange, paddingHorizontal: 30,
paddingVertical: 10))
                        .padding()
                    Text(String("Temperature: \(response.subdata(in: 2..<3).first ?? 0)"))
                        .appTextViewStyle(bodyColor:
temperatureColor(Double(response.subdata(in: 2..<3).first ?? 0)), paddingHorizontal: 25,
paddingVertical: 25)
                    Button("DOWN") {
                        editing = false
                        string = String(Int(response.subdata(in: 2..<3).first ?? 0) - 1)
                        bluetooth.send(Array(string.utf8))
                    }.buttonStyle(appButton(bodyColor: .blue, paddingHorizontal: 30,
paddingVertical: 10))
                        .padding()
                }.padding(50)
                let temp1 = response.subdata(in: 0..<1).first ?? 0
                let temp2 = response.subdata(in: 1..<2).first ?? 0
                let temperature = Double(temp1) + Double(temp2) / 100.0
                Text(String(format: "Current temperature: %.2f C", temperature))
                    .appTextViewStyle(bodyColor: .green, paddingHorizontal: 25,
paddingVertical: 25)
                Text("\(Date().timeIntervalSince1970)")
                Text("RSSI: \(rssi) dB")
                    .foregroundColor(Color.rssiColor(for: rssi))
                    .padding()
            }
            Spacer()
        }.sheet(isPresented: $presented){ ScanView(bluetooth: bluetooth, presented: $presented, list:
$list, isConnected: $isConnected) }
        .onAppear{ bluetooth.delegate = self }
    }
}

```

Додаток Д – Лістинг програмної частини керування BLE у мобільному
застосунку
(обов'язковий)

```

func connect(_ peripheral: CBPeripheral) {
    if current != nil {
        guard let current = current else { return }
        manager?.cancelPeripheralConnection(current)
        manager?.connect(peripheral, options: nil)
    } else { manager?.connect(peripheral, options: nil) } }

func disconnect() {
    guard let current = current else { return }
    manager?.cancelPeripheralConnection(current)
}

func startScanning() {
    peripherals.removeAll()
    manager?.scanForPeripherals(withServices: nil, options: nil)
}

func stopScanning() {
    peripherals.removeAll()
    manager?.stopScan()
}

func send(_ value: [UInt8]) {
    guard let characteristic = writeCharacteristic else { return }
    current?.writeValue(Data(value), for: characteristic, type: .withResponse)
}

enum State { case unknown, resetting, unsupported, unauthorized, poweredOff, poweredOn,
error, connected, disconnected }

struct Device: Identifiable {
    let id: Int
    let rssi: Int
    let uuid: String
    let peripheral: CBPeripheral }

```

**ПРОТОКОЛ
ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ
НА НАЯВНІСТЬ ТЕКСТОВИХ ЗАПОЗИЧЕНЬ**

Назва роботи: «Комплексна система децентралізованого керування опаленням»

Тип роботи: Магістерська кваліфікаційна робота
(БДР, МКР)

Підрозділ АПТ, ФІТА
(кафедра, факультет)

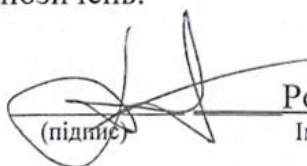
Показники звіту подібності Unicheck

Оригінальність 98.9% Схожість 1.9%

Аналіз звіту подібності (відмітити потрібне)

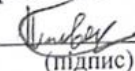
- Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.
- Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її автора. Роботу направити на розгляд експертної комісії кафедри.
- Виявлені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

Особа, відповідальна за перевірку

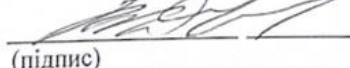

(підпис)

Роман МАСЛІЙ
Ім'я ПРІЗВИЩЕ

Ознайомлені з повним звітом подібності, який був згенерований системою Unicheck щодо роботи.

Автор роботи 
(підпис)

Артур ПИЛЯВЕЦЬ
Ім'я ПРІЗВИЩЕ

Керівник роботи 
(підпис)

Володимир ДУБОВОЙ
Ім'я ПРІЗВИЩЕ

ВІДГУК
керівника магістерської кваліфікаційної роботи

студента Пилявця Артура Ігоровича
(прізвище, ім'я, по батькові)
на тему Комплексна система децентралізованого керування опаленням

Тема магістерської кваліфікаційної роботи є актуальною в контексті зростання інтересу до сучасних систем опалення та енергоефективних технологій.

В роботі, студент чітко визначив мету та поставив завдання дослідження, включивши в аналіз як аспекти теоретичного характеру, так і практичні аспекти розробки системи керування. У роботі проведено аналіз існуючих систем опалення, висвітлено методи та технології, що використовуються в цьому контексті, та детально розглянуто структуру та функціональні можливості розробленої системи.

Студент продемонстрував достатньо глибокі знання фахової літератури, врахувавши актуальні тенденції у галузі та вміло використавши їх у роботі. Проявив самостійність у виборі підходів до досягнення результатів, а також здатність до аналізу та вибору правильних інженерних рішень.

Програмне забезпечення, розроблене в ході дослідження, було піддане тестуванню, яке підтвердило коректність розробки. Важливим є також вміння автора розпізнавати та виправляти можливі недоліки.

Слід відзначити, що студент Артур Пилявець є співавтором стартап-проекту, який відповідає темі його магістерської кваліфікаційної роботи і став фіналістом конкурсу Сікорський-челендж.

В процесі роботи спостерігалися певні відхилення від графіку виконання.

Магістерська кваліфікаційна робота відповідає спеціальності 151 – Автоматизація та комп'ютерно-інтегровані технології, заслуговує на оцінку «А», а її автор, Пилявець Артур Ігорович, заслуговує на присудження освітньої кваліфікації «Магістр» за спеціальністю «Автоматизація та комп'ютерно-інтегровані технології», освітня програма «Інформаційні системи і Інтернет речей».

Керівник магістерської кваліфікаційної
роботи

Професор каф. КСУ, д.т.н., професор **Володимир ДУБОВОЙ**



ВІДГУК
опонента магістерської кваліфікаційної роботи

студента Пилявця Артура Ігоровича
 (прізвище, ім'я, по батькові)
 на тему Комплексна система децентралізованого керування опаленням

Магістерська робота "Комплексна система децентралізованого керування опаленням" викликала певний інтерес своєю тематикою та обраною методологією дослідження. Автор систематично працював над аналізом та розробкою, впорядковуючи інформацію та деталізуючи структуру роботи.

У роботі чітко визначено завдання та мету дослідження, а також здійснено аналіз існуючих систем опалення та методів їх управління. Подано концепцію нової системи, що включає в себе як апаратне, так і програмне забезпечення. Важливим є врахування сучасних тенденцій та використання актуальних технологій у розробці.

Студент творчо розвинув і практично реалізував розробки, які виконувалися на кафедрі комп'ютерних систем управління протягом останніх років. Зокрема, запропонована на кафедрі система Mobile Smart Heaters розвинута у напрямку більш гнучкої конфігурації програмного і апаратного забезпечення.

Проте в ході обговорення роботи було виявлено деякі аспекти, які можна розглядати як потенційні області вдосконалення. Наприклад, відсутність зв'язку і координації між мобільними і стаціонарними засобами комплексної системи опалення будинку.

У цілому, магістерська кваліфікаційна робота відповідає спеціальності 151 – Автоматизація та комп'ютерно-інтегровані технології, заслуговує на оцінку «А», а її автор, Пилявець Артур Ігорович, заслуговує на присудження освітньої кваліфікації «Магістр» за спеціальністю «Автоматизація та комп'ютерно-інтегровані технології», освітня програма «Інформаційні системи і Інтернет речей».

Опонент магістерської кваліфікаційної роботи

Доцент кафедри комп'ютерних наук, к.т.н.

Руслан Белзецький

