

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

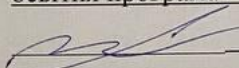
БАКАЛАВРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

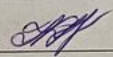
«IoT-пристрій контролю параметрів мікроклімату»

08-54.БКР.016.00.000 ПЗ

Виконав: студент 2 курсу, групи ІКІ-23мс
спеціальності 123 — «Комп'ютерна інженерія»
освітня програма — «Комп'ютерна інженерія».

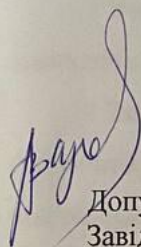
 Сорока Є.Р.

Керівник: к.т.н., доц. каф. ОТ

 Колесник І.С.

Рецензент: к.т.н., доц. каф. МБІС

 Карпинець В. В.

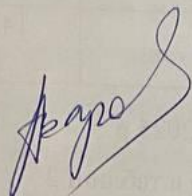


Допущено до захисту
Завідувач кафедри ОТ
д.т.н., проф. Азаров О.Д.

« 12 » 06 2025 р.

Вінниця ВНТУ – 2025 рік

Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра обчислювальної техніки
Освітньо-кваліфікаційний рівень бакалавр
Галузь знань — 12 Інформаційні технології
Спеціальність — 123 Комп'ютерна інженерія
Освітня програма — Комп'ютерна інженерія



ЗАТВЕРДЖУЮ

Завідувач кафедри ОТ проф., д.т.н.

Азаров О.Д.

« _____ » _____ 2025 р.

З А В Д А Н Н Я **НА БАКАЛАВРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ**

Сороці Євгенію Руслановичу

1 Тема роботи «IoT-пристрій контролю параметрів мікроклімату», керівник роботи Колесник Ірина Сергіївна, к.т.н., доцент, затверджені наказом вищого навчального закладу від 20.03.2025р. № 97.

2 Строк подання студентом роботи 13.05.2025 р.

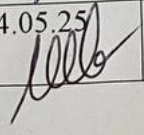
3 Вихідні дані до роботи: огляд доступних систем моніторингу кліматичних умов, характеристики програмно-апаратного комплексу, MQTT.

4 Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити): вступ, аналітичний огляд сучасних засобів контролю параметрів мікроклімату, обґрунтування технічного рішення та розробка апаратної платформи, програмна інтеграція IoT пристрою та його інтеграція з користувачем.

5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень): діаграма методів класів, Ethernet, модуль WIZnet W5500

6 Консультанти розділів роботи представлені в таблиці 1.

Таблиця 1 — Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв
1-3	к.т.н., доц. каф. ОТ Колесник І.С.	21.03.25 	13.05.25 
Нормоконтроль	ас. каф. ОТ Швець С.І.	14.05.25 	30.05.25

7 Дата видачі завдання 21.03.2025 р.

8 Календарний план наведено в таблиці 2.

Таблиця 2—Календарний план

№	Назва етапів виконання дипломної роботи	Строк виконання	Підпис
1	Постановка задачі роботи	21.03.25	вик.
2	Аналітичний огляд сучасних засобів контролю параметрів мікроклімату	21.03.25-30.03.25	вик.
3	Обґрунтування технічного рішення	21.03.25-30.03.25	вик.
4	Розробка апаратної платформи	31.03.25-13.04.25	вик.
5	Програмна інтеграція IoT пристрою та його інтеграція з користувачем.	14.04.25-27.04.25	вик.
6	Оформлення пояснювальної записки та ілюстративного матеріалу	28.04.25-11.05.25	вик.
7	Аналіз виконання роботи, висновки, додатки	28.04.25-11.05.25	вик.
8	Перевірка якості виконання бакалаврської роботи та усунення недоліків	13.05.25	вик.

Студент

Керівник роботи



Євгеній СОРОКА

к.т.н., доц. каф. ОТ Ірина КОЛЕСНИК

АНОТАЦІЯ

УДК 004.738.5

Сорока Є. Р. IoT-пристрій контролю параметрів мікроклімату. Пояснювальна записка містить 70 сторінки, 24 рисунки, 5 таблиць та 5 додатків. До списку використаних джерел входить 24 найменувань.

Бакалаврська кваліфікаційна робота присвячена розробці IoT-пристрою для автоматичного контролю параметрів мікроклімату з використанням сучасних сенсорів та мікроконтролерних технологій. У роботі розглянуто основи побудови систем моніторингу навколишнього середовища, визначено вимоги до точності та стабільності вимірювання основних кліматичних показників.

Проаналізовано актуальні методи збору та передавання даних в IoT-мережах, обґрунтовано вибір апаратної платформи та програмних рішень. У процесі реалізації створено прототип пристрою, здатного автономно вимірювати температуру, вологість та атмосферний тиск із подальшою передачею даних на сервер чи в хмару.

Розроблена система забезпечує безперервний моніторинг мікроклімату в реальному часі, що дозволяє інтегрувати її до більших інтелектуальних екосистем типу «розумного будинку» або екологічного моніторингу.

Ключові слова: мікроклімат, IoT, сенсори, температурний контроль, вологість, атмосферний тиск, мікроконтролер, моніторинг навколишнього середовища.

ABSTRACT

Soroka Ye. R. IoT Device for Microclimate Parameter Monitoring. The explanatory note contains 70 pages, 24 figures, 5 tables, and 5 appendices. The list of references includes 24 sources.

The bachelor's qualification work is devoted to the development of an IoT device for automatic monitoring of microclimate parameters using modern sensors and microcontroller technologies. The study covers the fundamentals of environmental monitoring systems, identifying requirements for the accuracy and stability of key climate parameter measurements.

Current methods of data collection and transmission in IoT networks are analyzed, and the choice of hardware platform and software solutions is substantiated. During implementation, a prototype of the device was developed, capable of autonomously measuring temperature, humidity, and atmospheric pressure with further data transmission to a server or cloud platform.

The developed system provides continuous real-time microclimate monitoring, allowing it to be integrated into larger intelligent ecosystems such as smart homes or environmental monitoring systems.

Keywords: microclimate, IoT, sensors, temperature control, humidity, atmospheric pressure, microcontroller, environmental monitoring.

ЗМІСТ

ВСТУП.....	4
1 АНАЛІТИЧНИЙ ОГЛЯД СУЧАСНИХ ЗАСОБІВ КОНТРОЛЮ ПАРАМЕТРІВ МІКРОКЛІМАТУ	6
1.1 Важливість забезпечення оптимального мікроклімату в приміщеннях.....	6
1.2 Огляд доступних систем моніторингу кліматичних умов	8
1.3 Порівняння існуючих рішень і обґрунтування вибору підходу	11
1.4 Формування вимог до програмно-апаратного комплексу	13
2 ОБҐРУНТУВАННЯ ТЕХНІЧНОГО РІШЕННЯ ТА РОЗРОБКА АПАРАТНОЇ ПЛАТФОРМИ.....	15
2.1 Основні кліматичні показники та методи їх вимірювання	15
2.2 Сфера застосування систем контролю параметрів навколишнього середовища.....	19
2.3 Вибір сенсорних компонентів: температури, вологості, тиску	20
2.4. Аналіз мікроконтролерної бази та засобів передавання даних	25
2.5 Інтеграція обраних технологій в апаратну архітектуру	32
2.6 Інтерфейси передачі даних в апаратній системі	32
2.6.1 Інтерфейс I ² C	32
2.6.2 Інтерфейс SPI.....	33
2.6.3 Одноканальний інтерфейс передачі даних (Single-Wire Interface) ...	34
2.6.4 Інтерфейс 1-Wire для датчиків температури DS18B20	34
2.7 Інтеграція обраних технологій в апаратну архітектуру	35
3 ПРОГРАМНА ІНТЕГРАЦІЯ ІОТ-ПРИСТРОЮ ТА ЙОГО ІНТЕРАКЦІЯ З КОРИСТУВАЧЕМ.....	41
3.1 Побудова архітектури програмного забезпечення	41
3.2 Реалізація обміну даними за допомогою MQTT.....	44
3.3 Налаштування середовища розробки та програмування контролера.....	47
3.4 Інтеграція з системами автоматизації типу MajorDoMo, Fibaro, OpenRemote	50

3.5 Алгоритм користування програмною системою моніторингу	51
ВИСНОВКИ	55
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	56
ДОДАТОК А Технічне завдання	59
ДОДАТОК Б Протокол перевірки навчальної (кваліфікаційної) роботи	62
ДОДАТОК В Графічна частина	63
ДОДАТОК Г Ілюстративна частина	65
ДОДАТОК Д Лістинг програмного забезпечення	67

ВСТУП

Актуальність. Мікроклімату у приміщеннях суттєво впливає на стан людини. До основних фізичних параметрів мікроклімату відносять температуру повітря, відносну вологість, атмосферний тиск, швидкість руху повітря та теплове випромінювання навколишніх об'єктів. Мікроклімат — це комплекс факторів, що визначають теплообмін організму з навколишнім середовищем [1].

Необхідність отримання актуальної інформації про параметри мікроклімату виникає при контролі умов праці, виробництва та зберігання продукції у приміщеннях. Високі температури спричиняють швидку втому, ризик теплового удару та можливі хвороби, тоді як низькі температури збільшують ймовірність простудних захворювань. Комбінація високої температури з підвищеною вологістю призводить до перегрівання організму, а низькі температура та вологість сприяють переохолодженню та пересиханню слизових оболонок.

Рух повітря прискорює теплообмін: при високій температурі це допомагає охолодженню, а при низькій — може викликати небажане переохолодження. Традиційні методи контролю мікрокліматичних параметрів за допомогою портативних чи настінних приладів із ручним фіксуванням результатів є малоефективними, трудомісткими і піддаються впливу людського фактора.

Метою роботи є створення програмно-апаратного модуля для автоматичного моніторингу параметрів мікроклімату будівлі та навколишнього середовища.

Задачі дослідження:

- проаналізувати існуючі програмні рішення для моніторингу параметрів мікроклімату та сформулювати вимоги для розробки нової системи;
- розробити архітектуру і функціональні можливості програмного забезпечення для моніторингу;
- вивчити інструментарій і бібліотеки для реалізації масштабованої та багатоканальної системи;

- створити програмно-апаратний комплекс для контролю параметрів мікроклімату;
- розробити тестові сценарії для перевірки роботи системи.

Об'єкт дослідження — система моніторингу параметрів мікроклімату будівлі та довкілля.

Предмет дослідження — методи та технології побудови програмно-апаратних рішень для автоматичного збору, передачі та збереження мікрокліматичних даних.

Практичне значення.

Розроблена система може бути впроваджена для контролю мікроклімату в промислових, виробничих та житлових приміщеннях, що сприятиме підвищенню комфорту і безпеки перебування людей, а також оптимізації умов зберігання продукції.

Методи дослідження.

У роботі застосовуються методи системного аналізу, об'єктно-орієнтованого проектування, моделювання процесів збору та передачі даних, а також експериментальна перевірка функціональності створеного модуля.

Наукова новизна.

Запропоновано комплексний підхід до побудови системи моніторингу параметрів мікроклімату, який поєднує сучасні апаратні компоненти, масштабоване програмне забезпечення та механізми автоматичного збору та обробки даних. Реалізовано прототип програмно-апаратного модуля, здатного безперервно відслідковувати ключові параметри мікроклімату і забезпечувати їх централізоване збереження.

1 АНАЛІТИЧНИЙ ОГЛЯД СУЧАСНИХ ЗАСОБІВ КОНТРОЛЮ ПАРАМЕТРІВ МІКРОКЛІМАТУ

1.1 Важливість забезпечення оптимального мікроклімату в приміщеннях

Мікроклімат приміщень — це один із ключових факторів, який визначає якість внутрішнього середовища, у якому постійно перебуває людина. Під мікрокліматом зазвичай розуміють сукупність параметрів повітряного середовища, що мають прямий вплив на тепловий стан організму, самопочуття, працездатність і навіть здоров'я людини. До таких параметрів відносять температуру повітря, відносну вологість, швидкість повітряного потоку, рівень теплового випромінювання та інші пов'язані характеристики.

Комфортне середовище всередині приміщень є важливим як у житлових, так і в комерційних та громадських будівлях: офісах, лікарнях, школах, навчальних аудиторіях, торгових залах тощо. Науково доведено, що тривале перебування в умовах надмірно сухого, вологого, холодного або перегрітого повітря викликає підвищене навантаження на організм людини, спричиняє зниження концентрації уваги, підвищену втому, загострення хронічних хвороб, а в окремих випадках — прямі загрози здоров'ю (особливо для дітей, літніх людей і осіб із серцево-судинними та респіраторними захворюваннями).

Санітарні норми та правила визначають допустимі та оптимальні значення параметрів мікроклімату для різних типів приміщень залежно від сезону. Наприклад, у житлових кімнатах взимку рекомендована температура повітря становить 20–22 °C, відносна вологість — 45–60 %, а швидкість повітря — до 0,2 м/с. У літній період ці значення можуть дещо змінюватися — наприклад, температура може зростати до 25 °C, а допустима вологість сягати до 65 %. Ці норми регламентуються державними стандартами (зокрема, ДСТУ EN ISO 7730, ДСанПіН 2.2.4.548-96 та ін.), а також вимогами проектної документації щодо енергоефективності та комфорту будівель.

До основних параметрів, що характеризують мікроклімат приміщень, відносяться:

— температура повітря — визначає теплове сприйняття людиною навколишнього середовища, підтримання постійної та комфортної температури зменшує теплове навантаження на організм;

— відносна вологість повітря — характеризує ступінь насиченості повітря водяною парою надто сухе повітря (менше 30 %) може спричинити пересушування слизових оболонок, а надмірно вологе (понад 70 %) — сприяти розвитку плісняви та грибків;

— швидкість руху повітря — впливає на ефективність теплообміну між тілом людини та навколишнім середовищем, надмірна циркуляція викликає ефект протягів, тоді як застій повітря сприяє підвищенню концентрації CO₂ та інших шкідливих речовин.

Для наочності у таблиці 1.1 наведено приклад рекомендованих параметрів мікроклімату для різних типів приміщень відповідно до сезону.

Ці нормативи мають враховуватись при проєктуванні, будівництві та експлуатації будівель, а також при встановленні систем опалення, вентиляції та кондиціонування повітря.

У зв'язку зі зростанням інтересу до концепції розумного будинку (Smart Home) та Інтернету речей (IoT), контроль параметрів мікроклімату стає не лише елементом комфорту, а й інструментом енергоефективності. Сучасні технології дозволяють у режимі реального часу відстежувати стан середовища в приміщенні, реагувати на зміни та автоматично регулювати кліматичні умови відповідно до встановлених сценаріїв або персональних налаштувань користувача.

Таким чином, підтримка оптимального мікроклімату в приміщеннях є надзвичайно важливою складовою як побутової, так і професійної діяльності людини. А впровадження автоматизованих систем контролю мікроклімату дозволяє підвищити якість життя, зменшити витрати на енергоспоживання та забезпечити відповідність санітарним вимогам.

Таблиця 1.1 — Оптимальні та допустимі норми мікроклімату залежно від типу приміщення та пори року

Тип приміщення	Температура повітря, °C	Відносна вологість, %	Швидкість руху повітря, м/с
Житлова кімната (зима/літо)	20–22 / 22–25	45–60 / 30–65	до 0.2 / до 0.3
Кухня	19–21	не нормується	до 0.2
Ванна	24–26	не нормується	до 0.2
Туалет	19–21	не нормується	до 0.2
Приміщення для навчання	20–22	45–60	до 0.2
Міжквартирний коридор	18–20	45–60	нормується
Комори (у холодну пору року)	16–18	нормується	нормується
Вестибюль	16–18	нормується	нормується

1.2 Огляд доступних систем моніторингу кліматичних умов

Проблема забезпечення належної якості повітря в житлових, офісних, промислових та спеціалізованих приміщеннях (таких як серверні, лабораторії, склади, медичні установи тощо) є однією з найбільш актуальних протягом останніх десятиліть. Це зумовлено підвищеними вимогами до умов праці, зберігання чутливої техніки та матеріалів, а також до створення комфортного та безпечного середовища для перебування людини. Зміни кліматичних умов, зростання щільності забудови, використання нових оздоблювальних матеріалів та систем герметизації ще більше ускладнили проблему підтримки оптимального мікроклімату в закритих просторах.

Одним з найважливіших аспектів створення сприятливого мікроклімату є контроль та регулювання температури, вологості повітря, а також забезпечення належної вентиляції та циркуляції повітряних потоків. Для реалізації таких завдань у сучасних умовах активно впроваджуються автоматизовані системи керування кліматом, які базуються на апаратно-програмних засобах, включаючи мікроконтролери, бездротові сенсори, інтелектуальні алгоритми обробки даних та інтерфейси взаємодії з користувачем.

Системи управління мікрокліматом використовуються для регулювання і підтримки необхідних параметрів температури, відносної вологості повітря, рівня вуглекислого газу, інтенсивності вентиляції та інших характеристик, що впливають на якість повітря. Вони можуть бути як централізованими, так і локальними, призначеними для роботи в межах одного або декількох суміжних приміщень.

Сучасні кліматичні системи, зокрема ті, що базуються на принципах автоматичного регулювання, включають в себе численні компоненти: сенсорні модулі, модулі обробки та зберігання даних, засоби візуалізації, виконавчі пристрої (нагрівачі, охолоджувачі, зволожувачі, вентиляційні блоки), а також засоби оповіщення користувача. Вони здійснюють постійний моніторинг параметрів повітря у приміщенні, аналізують отримані дані та автоматично здійснюють необхідні коригування, ґрунтуючись на показаннях датчиків температури, вологості та інших параметрів.

В межах даної роботи розглянуто приклад побудови системи контролю температури і вологості в приміщенні на основі бездротових технологій передачі даних. Зокрема, реалізовано прототип системи дистанційного моніторингу з використанням програмно-апаратної платформи Arduino на базі мікроконтролера ATmega328P [3]. Така платформа є однією з найпоширеніших в освітньому середовищі та аматорських розробках, оскільки поєднує низьку вартість, простоту програмування та гнучкість в розширенні функціональності.

В якості сенсора температури та вологості використовується мікросхема DHT21 (також відома як AM2301), яка забезпечує достатньо точні вимірювання

(± 0.5 °C для температури та ± 3 % для вологості) в широкому робочому діапазоні. Передача інформації з датчика на комп'ютер реалізується за допомогою методу кодоімпульсної модуляції, що дозволяє передавати цифрові сигнали без значних спотворень навіть у складних умовах.

Передавач радіоканалу для передачі вимірювальних даних працює на частоті 433 МГц і має вихідну потужність до 15 мВт на навантаженні 50 Ом, що дозволяє здійснювати обмін інформацією на відстані до декількох десятків метрів у приміщеннях.

Ця система дистанційного моніторингу орієнтована, зокрема, на контроль мікроклімату в серверних приміщеннях, де перегрів або надмірна вологість можуть призвести до відмови обладнання або втрати даних. Дані з сенсорів надсилаються на пульт системного адміністратора, де обробляються, візуалізуються та за потреби активується акустичне оповіщення при виявленні критичних значень параметрів.

Альтернативним підходом є реалізація системи контролю на основі мікроконтролера AVR ATmega8515 із сенсором DS18B20, який дозволяє зчитувати температурні значення з цифровим виходом по шині 1-Wire [4]. Принцип дії полягає у відображенні поточних показників температури на дисплеї та їх аналізі відповідно до заданого порогового значення. У разі виявлення відхилень автоматично вмикається виконавчий пристрій – кондиціонер або нагрівач.

Ще одним прикладом є локальна система контролю, реалізована на базі Arduino UNO з сенсором DHT11, що має ще кращі метрологічні характеристики, включаючи вищу точність та ширший діапазон вимірювань порівняно з DHT11 і DHT21. Підключення до комп'ютера здійснюється через USB-інтерфейс, після чого дані зчитуються та передаються в режимі реального часу з інтервалом близько 2 секунд. Система підтримує графічний інтерфейс, що дозволяє відслідковувати зміну температури та вологості з плином часу. При перевищенні допустимих значень активується звукове сповіщення, а також можливе

автоматичне увімкнення зовнішніх пристроїв — терморегуляторів, осушувачів чи зволожувачів повітря.

Окрему нішу займають системи моніторингу на базі бездротових сенсорних мереж (WSN), які дозволяють здійснювати дистанційний екологічний моніторинг. Такі пристрої дозволяють контролювати не лише температуру та вологість, а й рівень освітленості, вміст шкідливих речовин тощо. Конструкція зазвичай включає температурний сенсор, блок сигналізації, програмований модуль (мікроконтролер, пам'ять, радіоприймач), інтерфейс обміну з ПК і автономне джерело живлення [5].

Головними перевагами таких систем є низьке енергоспоживання, швидка обробка сигналів у реальному часі, універсальність, можливість автономної роботи та гнучкість у масштабуванні. Подібні пристрої здатні працювати в складі сенсорних мереж і здійснювати моніторинг у важкодоступних або небезпечних для людини місцях, таких як тунелі, підвали, промислові зони або зони екологічного контролю.

Таким чином, використання сучасних мікроконтролерних технологій та бездротових засобів зв'язку відкриває широкі можливості для побудови ефективних, надійних та економічних систем контролю мікроклімату, що мають широке практичне застосування в різних сферах діяльності.

1.3 Порівняння існуючих рішень і обґрунтування вибору підходу

Сучасним викликом у сфері контролю кліматичних умов є забезпечення оптимальних показників температури та вологості в різних типах приміщень, зокрема в приміщеннях третьої категорії, які відзначаються підвищеним рівнем теплового навантаження. Забезпечення індивідуального мікроклімату в таких умовах є надзвичайно актуальним, адже саме від цього залежить комфорт, безпека і продуктивність перебування людей у приміщеннях.

Вивчення існуючих технічних рішень у галузі моніторингу і регулювання кліматичних параметрів вказує на кілька суттєвих проблем, які обмежують їх ефективність. Одним із ключових недоліків більшості представлених пристроїв

є відсутність єдиної, універсальної програмно-апаратної платформи, яка б могла служити базою для побудови різноманітних додатків на основі сенсорних мереж. Така платформа мала б забезпечувати не лише збір і обробку даних з численних сенсорів, але й гнучке масштабування, інтеграцію з різними типами комунікаційних інтерфейсів та можливість кастомізації алгоритмів управління.

Відсутність подібної платформи особливо критична у випадках, коли потрібно реалізувати безперервний і стабільний моніторинг кліматичних показників у реальному часі. Наявні рішення часто є або занадто простими, що не дозволяє масштабувати систему або збирати комплексні дані, або надто складними, що ускладнює їх впровадження і подальше обслуговування.

Крім того, серед суттєвих недоліків, що впливають на практичне застосування, є складнощі з експлуатацією пристроїв, зокрема у контексті їх живлення. Багато сучасних сенсорних пристроїв розраховані на роботу від автономних джерел живлення — батарейок чи акумуляторів — що обмежує час їх безперервної роботи і підвищує витрати на технічне обслуговування (часту заміну або заряджання акумуляторів). У цьому сенсі найбільш оптимальним є використання зовнішнього джерела живлення, яке забезпечує стабільну роботу без ризику перерв у функціонуванні системи, що критично важливо для об'єктів із підвищеними вимогами до контролю параметрів мікроклімату.

Проведений аналіз сучасного стану проблеми показує, що попри велику кількість наявних продуктів і рішень, на сьогоднішній день не існує комплексної, ефективної системи, яка б гарантувала забезпечення комфортних температурних і вологісних умов з урахуванням індивідуальних особливостей кожного приміщення, зокрема для складних середовищ третьої групи. Це свідчить про необхідність розробки нових інтегрованих апаратно-програмних платформ, що базуються на відкритих стандартах і забезпечують високу гнучкість, надійність, зручність експлуатації та можливість масштабування.

Отже, для досягнення ефективного контролю мікроклімату потрібен підхід, який поєднує:

- універсальність — можливість підключення широкого спектру сенсорів і модулів;
- гнучкість — підтримка різних режимів роботи та обробки даних;
- надійність — стабільне живлення і безперервну роботу;
- простоту інтеграції — легкість підключення до існуючих систем управління;
- зручність експлуатації — мінімальні вимоги до обслуговуван.

Враховуючи викладене, стає очевидним, що найкращим рішенням є розробка системи на базі відкритої програмно-апаратної платформи, що дозволить ефективно вирішити наявні проблеми та забезпечити широкий спектр застосувань у різних умовах експлуатації.

1.4 Формування вимог до програмно-апаратного комплексу

Виходячи з аналізу потенційних сфер застосування системи для вимірювання параметрів навколишнього середовища, можна виділити комплекс основних вимог, яким має відповідати розроблювана програмно-апаратна система.

Мобільність — система має бути максимально мобільною, що сприятиме зручності її експлуатації та розширенню зони покриття. Мобільність може бути досягнута шляхом використання бездротових технологій передачі даних між вузлами системи та центральним комп'ютером, а також за рахунок компактних розмірів обладнання. Завдяки цьому користувачі зможуть легко переміщувати і швидко розгортати систему в різних приміщеннях або на відкритих територіях.

Масштабованість: важливо, щоб система могла без втрати продуктивності збільшувати кількість підключених функціональних вузлів (датчиків, контролерів тощо). Це дозволить розширювати систему в залежності від потреб конкретного об'єкта. Крім того, має бути забезпечена можливість її інтеграції з іншими існуючими системами, що підвищить універсальність рішення. Для цього необхідно передбачити легке додавання або вилучення окремих датчиків або модулів без суттєвого впливу на роботу всієї системи.

Модульність: розроблювана система має мати модульну структуру, що дозволить використовувати її як самостійний пристрій, так і як частину більш складних комплексів. Такий підхід забезпечить гнучкість застосування, а також спростить оновлення або масштабування системи у майбутньому.

Багатоканальність: система повинна забезпечувати одночасне і паралельне вимірювання кількох різних фізичних параметрів, що значно підвищить ефективність роботи та скоротить час на отримання повного набору даних. Для цього в системі передбачаються різноманітні датчики, здатні працювати паралельно, що дозволить отримувати комплексну картину мікроклімату без затримок.

Автоматизація формування звітності: важливо, щоб система автоматично збирала, обробляла та формувала звіти за результатами вимірювань. Протоколи мають бути зручними для користувача і представленими у вигляді таблиць, графіків або інших візуалізацій, що полегшують аналіз та прийняття рішень.

Виходячи з перерахованих вимог, був складений перелік основних фізичних величин, які повинна фіксувати система для забезпечення повноцінного контролю параметрів мікроклімату. Серед них найчастіше виділяють такі ключові показники:

- температура повітря;
- відносна вологість повітря;
- атмосферний тиск;
- рівень освітленості.

Ці параметри дозволяють комплексно оцінювати умови навколишнього середовища, що впливають на комфорт і безпеку перебування людей у приміщеннях, а також дають можливість своєчасно реагувати на зміни мікроклімату.

2 ОБҐРУНТУВАННЯ ТЕХНІЧНОГО РІШЕННЯ ТА РОЗРОБКА АПАРАТНОЇ ПЛАТФОРМИ

Розробка ефективної системи моніторингу мікрокліматичних параметрів вимагає комплексного підходу до вибору технічних рішень, що включає в себе аналіз характеристик середовища, визначення способів збору даних, вибір відповідних сенсорів, мікроконтролера та методів передавання інформації. У цьому розділі викладено обґрунтування вибраної архітектури апаратної частини системи, починаючи з вивчення фізичних величин, що підлягають вимірюванню, і закінчуючи інтеграцією всіх апаратних компонентів у єдину функціональну платформу.

2.1 Основні кліматичні показники та методи їх вимірювання

Температура — фізична величина, що характеризує рівень теплової енергії частинок речовини. Вона прямо пропорційна середній кінетичній енергії хаотичного руху молекул або атомів у тілі та відображає тепловий стан системи. Температурні показники зазвичай вимірюються в градусах Цельсія, Кельвіна або Фаренгейта.

Цей параметр відіграє ключову роль у забезпеченні комфортних умов життєдіяльності людини, регулюванні технологічних процесів, а також у гарантуванні надійної роботи різних приладів і систем автоматики.

Існує широкий спектр засобів для вимірювання температури, серед яких можна виділити термопари, терморезистори, а також напівпровідникові сенсори.

Термопари функціонують на основі ефекту Зеєбека — коли два провідники з різних металів з'єднуються і піддаються нагріванню, між їхніми контактами виникає електрорушійна сила, величина якої залежить від температури. Такі пристрої є термоелектричними перетворювачами і використовуються для вимірювання температур в широкому діапазоні, особливо в промисловості.

Терморезистори (RTD або NTC/PTC) – це пасивні сенсори, чутливі до змін температури завдяки залежності електричного опору провідника від

температури. Через них пропускається струм, і зміна опору дозволяє обчислити температуру. Варто враховувати вплив опору з'єднувальних проводів і паразитних напруг, які можуть викривляти результати вимірювання.

У сучасних електронних пристроях активно застосовуються напівпровідникові сенсори, зокрема на основі діодів, транзисторів або спеціалізованих мікросхем. Принцип їх дії базується на температурній залежності падіння напруги на р-п переході при прямому зміщенні. Така залежність наближена до лінійної, що значно спрощує обробку сигналу.

Напівпровідникові сенсори здатні працювати в діапазоні температур від -55°C до $+150^{\circ}\text{C}$ і завдяки своїм розмірам легко інтегруються в мікросхеми. Багато мікропроцесорних і мікроконтролерних платформ мають вбудовані температурні сенсори цього типу.

Особливої уваги заслуговують цифрові температурні датчики, що мають інтегрований аналого-цифровий перетворювач (АЦП) і передають дані за допомогою цифрових інтерфейсів, таких як 1-Wire, I²C або SPI. Вони є точними, компактними, доступними за ціною та не потребують додаткової лінеаризації або компенсації температури холодного спаю, як у випадку з термопарами. Завдяки цим перевагам вони широко застосовуються в побутовій і промисловій електроніці.

Вологість повітря — це фізична величина, що відображає кількість водяної пари, яка міститься в атмосферному повітрі. Відносна вологість — це співвідношення поточної кількості водяної пари до максимально можливої при тій же температурі, виражене у відсотках [7].

Для вимірювання вологості використовують спеціальні пристрої, які називаються гігрометрами або датчиками вологості.

За принципом дії гігрометри поділяють на кілька типів:

— ємнісні гігрометри функціонують за рахунок зміни ємності конденсатора, у якому діелектриком є повітря, оскільки діелектрична проникність повітря залежить від рівня вологості, зміни вологості спричиняють відповідні зміни ємності;

— резистивні датчики вологості складаються з пари електродів, між якими нанесено шар матеріалу, електричний опір якого чутливо змінюється зі зміною вологості повітря;

— термісторні датчики вологості працюють на основі пари термісторів елементів з температурозалежним опором, один з термісторів розміщується у відкритій камері, через яку проходить повітря, інший у закритій камері з сухим повітрям; якщо напруга між ними дорівнює нулю, температури, а отже і вологості, в обох камерах однакові, поява різниці напруги свідчить про відмінність у вологості, яку можна обчислити за цим показником;

— оптичні датчики вологості працюють на основі принципу точки роси, у таких пристроях світлодіод направляє промінь на дзеркальну поверхню, що охолоджується або нагрівається до досягнення точки роси, а момент появи конденсату (крапельологи) на дзеркалі світло розсіюється, і фотодетектор реєструє зміну інтенсивності; система зворотного зв'язку підтримує температуру дзеркала на рівні точки роси, що дозволяє точно визначати вологість при заданому атмосферному тиску.

Серед переваг ємнісних датчиків можна відзначити їх стійкість до високих температур, малу залежність характеристик від температури навколишнього середовища, здатність до повного відновлення після утворення конденсату, а також помірну чутливість до хімічно агресивних середовищ.

Атмосферний тиск — це фізична величина, що відображає силу, з якою повітря атмосфери тисне на об'єкти, розташовані на поверхні Землі та в повітряному просторі. Зазвичай атмосферний тиск вимірюють у паскалях (Па), барах (бар) або міліметрах ртутного стовпа (мм рт. ст.) [8].

Датчики абсолютного тиску використовуються для визначення тиску повітря, газів, парів або рідин відносно абсолютного вакууму, тобто з відліком від нульового тиску. Усі сенсори тиску працюють з певною референтною точкою. У випадку абсолютного тиску — це повний вакуум.

Окрім датчиків абсолютного тиску, існують також:

— датчики відносного тиску, які вимірюють тиск відносно атмосферного;

- вакуумні датчики, що фіксують тиск нижчий за атмосферний;
- диференційні датчики тиску, які визначають різницю між двома тисками.

У конструкції датчика абсолютного тиску ключовим елементом є сенсор, одна сторона якого герметично ізольована у вакуумній камері, а інша піддається дії зовнішнього тиску. За зміною деформації чутливого елемента електронна схема формує сигнал, що відповідає абсолютному значенню тиску.

Останнім часом значного поширення набули п'єзорезистивні датчики тиску. Вони бувають двох типів: для роботи в нейтральному (неагресивному) середовищі та в хімічно активному (агресивному) середовищі. Відмінність полягає у способі передачі тиску до сенсора:

- у неагресивному середовищі чутливий елемент безпосередньо контактує з вимірюваним середовищем або захищений силіконовим гелем;
- у випадку агресивного середовища використовується металева мембрана, зазвичай з нержавіючої сталі, яка передає тиск на сенсор, не вступаючи в хімічну реакцію з середовищем.

Принцип дії таких датчиків базується на ефекті п'єзорезистивного моста: при деформації мембрани під тиском змінюється опір елементів моста, що призводить до виникнення електричного сигналу, пропорційного вимірюваному тиску. Завдяки високій пружності монокристалічного кремнію забезпечується висока стабільність показників у широкому діапазоні тисків.

До переваг п'єзорезистивних датчиків належать:

- широкий діапазон вимірювання;
- висока чутливість;
- здатність паралельно реєструвати температуру для подальшого коригування тискових показників.

Освітленість — це фізична характеристика, яка описує інтенсивність освітлення певної точки поверхні. Вона визначається як відношення світлового потоку, що падає на поверхню, до площі цієї поверхні. Основною одиницею вимірювання освітленості є люкс (лк) [9].

Для контролю рівня освітлення використовуються спеціальні прилади — люксметри, оснащені чутливими до світла перетворювачами. Коли світловий потік досягає фоточутливого елемента, в його напівпровідниковій структурі відбувається вивільнення електронів. У результаті цього явища в фотоелементі виникає електричний струм, значення якого прямо пропорційне до інтенсивності освітлення.

2.2 Сфера застосування систем контролю параметрів навколишнього середовища

Системи моніторингу параметрів навколишнього середовища знаходять широке застосування в різних галузях. Вони відіграють важливу роль у промислових процесах, дослідницьких лабораторіях, житлових приміщеннях, а також на метеорологічних станціях, де необхідний постійний контроль за умовами довкілля.

Забезпечення стабільного мікроклімату є критично важливим для якості виробництва, збереження технологічних процесів і комфорту людей. Саме тому все частіше використовуються автоматизовані системи, які здійснюють безперервне вимірювання та аналіз середовищних показників.

Такі системи можуть бути інтегровані в екологічний моніторинг, технології «розумного дому», системи управління випробуванням вимірювального обладнання, а також у контроль параметрів повітряного середовища у приміщеннях різного призначення.

У контексті екологічного моніторингу здійснюється спостереження за станом атмосферного повітря, водних об'єктів, ґрунту та геологічного середовища з метою виявлення змін, прогнозування можливих загроз та своєчасного реагування для запобігання шкідливому впливу на здоров'я людей, флору, фауну та інфраструктуру.

Типова система моніторингу включає:

- виявлення та обстеження об'єкта спостереження;
- визначення його поточного стану;

- моделювання можливих змін;
- передачу інформації у зручному для аналізу вигляді.

Ще однією важливою сферою застосування є метеорологія — наука, що вивчає атмосферу та її властивості. Метеостанції, оснащені спеціалізованим обладнанням, здійснюють постійні вимірювання температури, вологості, тиску, швидкості та напрямку вітру, рівня опадів, видимості тощо. Ці дані необхідні для точного прогнозування погодних умов.

У межах інтелектуальних інженерних рішень все більше уваги приділяється впровадженню системи «розумний будинок». Це концепція автоматизації житлових будівель, яка охоплює контроль освітлення, опалення, вентиляції, кондиціонування, безпеки, а також керування побутовою технікою. Часто для віддаленого доступу до таких систем використовується Wi-Fi або інші бездротові технології.

Автоматизована будівля є прикладом комплексної системи, в якій різні інженерні підсистеми працюють узгоджено та адаптуються до змін у довкіллі.

В основі роботи таких систем лежать сенсори, що дозволяють зчитувати актуальні параметри середовища. Найбільш поширені серед них:

- датчики температури;
- датчики вологості;
- гігростати для стабілізації вологості;
- терморегулятори для управління тепловим режимом;
- термостати, які автоматично підтримують задану температуру.

Метою використання таких пристроїв є створення комфортного мікроклімату у приміщеннях при одночасному зниженні енерговитрат. У результаті система вимірювання мікрокліматичних показників є не лише засобом зручності, а й важливим компонентом енергоефективних і екологічно безпечних технологій управління будівлями.

2.3 Вибір сенсорних компонентів: температури, вологості, тиску

Для реалізації функцій моніторингу кліматичних параметрів у складі розроблюваної системи було обрано цифровий датчик DHT11, який широко використовується в проєктах автоматизації завдяки оптимальному поєднанню точності, надійності та доступної вартості.

Сенсор DHT11 побудований на основі ємнісного принципу вимірювання вологості та використовує вбудований термістор для фіксації температури навколишнього середовища. Основною перевагою цього модуля є наявність інтегрованого аналого-цифрового перетворювача (АЦП), що дозволяє подавати вже готовий цифровий сигнал без потреби у додаткових зовнішніх компонентах, що значно спрощує інтеграцію в мікроконтролерні системи.

Комунікація з мікроконтролером здійснюється за допомогою цифрового однопровідного інтерфейсу, який базується на шині з відкритим колектором. Для забезпечення коректної роботи цього типу зв'язку обов'язково використовується резистор підтяжки (pull-up resistor) номіналом від 5 до 10 кОм до позитивного джерела живлення. Такий тип інтерфейсу дозволяє економити ресурси мікроконтролера, що особливо актуально при роботі з обмеженою кількістю виводів.

Серед додаткових переваг DHT11 можна виділити його енергоефективність — у режимі очікування сенсор споживає мінімальний струм, що забезпечує довготривалу автономну роботу в системах, які живляться від батарей. При цьому час одного вимірювання становить близько двох секунд, що є прийнятним для задач неоперативного контролю мікроклімату.

Пристрій постачається в компактному пластиковому корпусі з вентиляційними отворами, які сприяють кращому контакту сенсора з навколишнім середовищем, що позитивно впливає на точність вимірювання. Наявність монтажних отворів або фіксуючих пазів у конструкції корпусу полегшує встановлення сенсора в різноманітні конструкції та корпуси.

На рисунку 2.1 зображено зовнішній вигляд модуля DHT11.

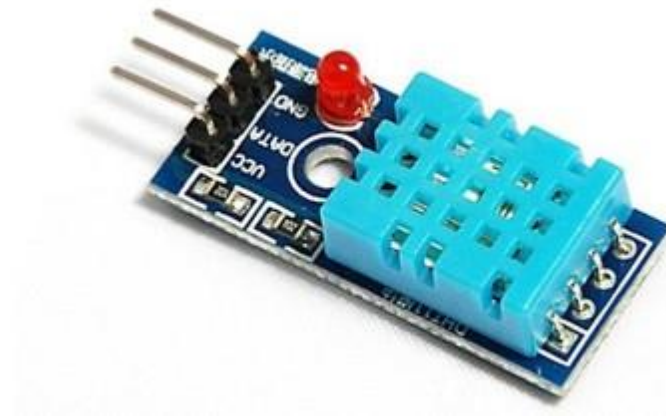


Рисунок 2.1 — Зовнішній вигляд датчика температури та вологості DHT11

Основні технічні параметри сенсора наведено у таблиці 2.1.

Таблиця 2.1 — Характеристики сенсора DHT11

Параметр	Значення
Робоча напруга живлення	3.0 В – 5.5 В
Діапазон вимірювання температури	від -40 °С до +80 °С
Точність вимірювання температури	±0.5 °С
Діапазон вимірювання відносної вологості	0% – 100%
Точність вимірювання вологості	±2% ... ±5%
Споживаний струм у активному режимі	1 мА – 1.5 мА
Струм у режимі очікування	40 мкА – 50 мкА
Кількість виводів	4
Мінімальний інтервал між вимірюваннями	2 секунди
Формат вихідних даних	16-біт цифрове значення

DHT11 добре підходить для проєктів, де необхідна балансована точність та мінімальна складність апаратної реалізації. Його використання є оптимальним варіантом як для автономних мобільних пристроїв, так і для стаціонарних систем моніторингу навколишнього середовища. У рамках даного проєкту він виконує

ключову роль у фіксації двох базових параметрів мікроклімату — температури та вологості.

DS18B20 — це компактний цифровий датчик температури, що забезпечує високу точність вимірювань за доступною вартістю. Його основною особливістю є використання однопровідного інтерфейсу 1-Wire, розробленого компанією Dallas Semiconductor, який дозволяє передавати дані та живлення по одному проводу. Цей протокол є широко розповсюдженим завдяки простоті реалізації та можливості підключення декількох датчиків до однієї лінії зв'язку, що знижує складність кабельної інфраструктури.

DS18B20 сумісний з багатьма мікроконтролерними платформами, зокрема Arduino, AVR, PIC і ARM. Для Arduino існують готові бібліотеки, які значно спрощують взаємодію з пристроєм, забезпечуючи швидке отримання температурних даних без потреби у складному програмуванні.

На рисунку 2.2 наведено зовнішній вигляд датчика температури DS18B20. Основні технічні параметри сенсора наведено у таблиці 2.2.

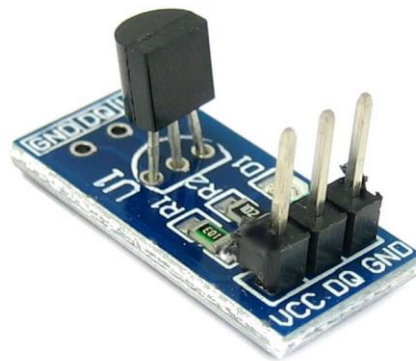


Рисунок 2.2 — Зовнішній вигляд датчика DS18B20

BMP280 — це високоточний цифровий датчик атмосферного тиску та температури, розроблений компанією Bosch Sensortec. Він призначений для широкого спектра застосувань, включаючи метеостанції, мобільні пристрої, системи моніторингу довкілля та проекти з мікроконтролерами. Основною особливістю BMP280 є його здатність вимірювати атмосферний тиск з високою

точністю, що дозволяє, зокрема, обчислювати висоту над рівнем моря з похибкою до кількох метрів.

Таблиця 2.2 — Технічні характеристики датчика DS18B20

Параметр	Значення
Діапазон вимірюваних температур	від −55 °C до +125 °C
Точність	±0,5 °C (в межах −10 °C ... +85 °C)
Час вимірювання	близько 94 мс
Напруга живлення	3 В – 5,5 В
Струм у стані очікування	~750 нА
Струм у робочому режимі	~1 мА
Інтерфейс зв’язку	1-Wire
Можливість адресації	до 64 пристроїв на одній шині
Формат даних	9–12 біт, програмовано

Датчик підтримує два основні інтерфейси зв’язку — I2C та SPI, що забезпечує гнучкість при інтеграції з різними мікроконтролерними платформами, зокрема Arduino, ESP32, STM32 та іншими. Існують готові бібліотеки для Arduino IDE, які спрощують процес зчитування даних та їх обробки, зокрема бібліотеки Adafruit BMP280 або SparkFun BMP280.

На рисунку 2.3 зображено зовнішній вигляд датчика BMP280. Основні технічні характеристики наведено у таблиці 2.3.

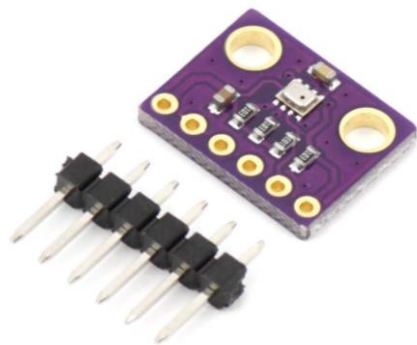


Рисунок 2.3 — Зовнішній вигляд датчика BMP280

Таблиця 2.3 — Технічні характеристики датчика BMP280

Параметр	Значення
Діапазон вимірювання тиску	від 300 до 1100 гПа
Точність вимірювання тиску	± 1 гПа
Діапазон вимірювання температури	від -40 °C до $+85$ °C
Точність вимірювання температури	± 1.0 °C
Робоча напруга	1.71 В – 3.6 В
Інтерфейс зв'язку	I2C, SPI
Роздільна здатність	до 0.01 гПа (для тиску)
Споживання енергії	~ 2.7 μ A (при 1 Гц)

2.4 Аналіз мікроконтролерної бази та засобів передавання даних

Arduino Nano — це компактна та потужна мікроконтролерна плата, побудована на основі мікросхеми ATmega328P, яка широко використовується у проектах вбудованих систем, Інтернету речей (IoT), автоматизації та моніторингу. Завдяки невеликим розмірам, енергоефективності та великій кількості входів-виходів Arduino Nano ідеально підходить для розробки компактних пристроїв, де важливо поєднання функціональності та мінімального споживання простору. На рисунку 2.4 зображено зовнішній вигляд мікроконтролерної плати Arduino Nano.

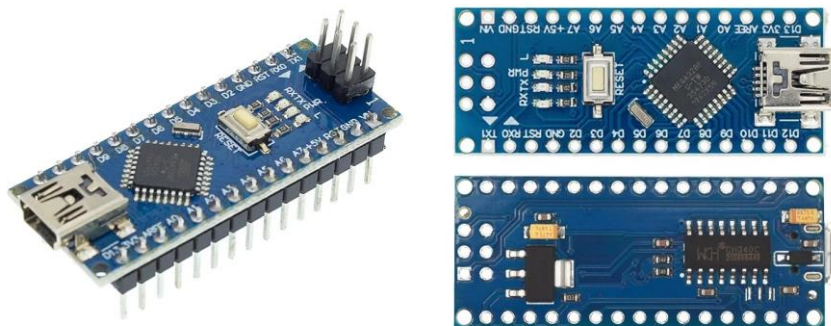


Рисунок 2.4 — Мікроконтролерна плата Arduino Nano

Нижче приведено основні характеристики Arduino Nano.

1. Мікроконтролер: ATmega328P — 8-бітовий AVR мікроконтролер зі RISC-архітектурою, що забезпечує високу продуктивність при низькому енергоспоживанні.

2. Тактування: 16 МГц кварцовий генератор — дозволяє працювати з достатньою швидкістю для більшості задач керування та обробки сигналів.

3. Пам'ять:

— Flash-пам'ять — 32 КБ (2 КБ використовуються завантажувачем) для зберігання програми;

— SRAM — 2 КБ для оперативних даних;

— EEPROM — 1 КБ для збереження даних, що зберігаються після вимкнення живлення.

4. Цифрові входи/виходи (GPIO): 14 портів, з яких 6 підтримують широтно-імпульсну модуляцію (ШИМ), що дозволяє керувати яскравістю світлодіодів, швидкістю двигунів та іншими пристроями.

5. Аналогові входи: 8 каналів АЦП (10-бітний аналого-цифровий перетворювач), що забезпечує високу точність вимірювань аналогових сигналів, таких як датчики температури, вологості, освітленості.

6. Комунікаційні інтерфейси:

— UART (послідовний порт) для обміну даними з ПК чи іншими пристроями.

— SPI (Serial Peripheral Interface) для швидкої передачі даних з периферійними модулями (екрани, пам'ять, датчики).

— I2C (Inter-Integrated Circuit) для підключення декількох пристроїв на двох проводах, популярний інтерфейс для сенсорів та дисплеїв.

7. Живлення:

— вхід VIN (6-12 В) для живлення від зовнішнього джерела;

— 5 В стабілізоване живлення для плати та підключених модулів;

— USB живлення від комп'ютера або зовнішніх блоків через mini- або micro-USB роз'єм.

8. Розміри: 45×18 мм — це дозволяє легко інтегрувати плату в компактні пристрої або на макетні плати.

9. Додаткові елементи:

- вбудований стабілізатор напруги, що забезпечує стабільність живлення;
- кнопка «Reset» для скидання програми;
- виводи для підключення зовнішніх кристалів, програматора через ICSP.

Переваги Arduino Nano.

1. Компактність: Значно менший розмір порівняно з Arduino Uno, що дає змогу встановлювати його у пристрої з обмеженим простором.

2. Гнучкість живлення: Можливість живлення від USB або зовнішнього джерела з широким діапазоном напруги.

3. Велика кількість аналогових входів: 8 аналогових каналів дозволяють підключати більше сенсорів без необхідності додаткових мультиплексорів.

4. Повна сумісність з Arduino IDE: Програмування та завантаження прошивок відбувається так само легко, як і на інших платах Arduino.

5. Велика підтримка спільноти: Широкий набір бібліотек і прикладів робить Arduino Nano популярним вибором для початківців і професіоналів.

6. Можливість монтажу на макетну плату (breadboard): DIP корпус дозволяє просто інтегрувати плату в прототипи без пайки.

Застосування Arduino Nano.

Arduino Nano часто використовується у таких сферах.

1. Вимірювальні пристрої: збір даних з датчиків температури, вологості, освітленості, тиску.

2. Системи «Розумний будинок»: керування освітленням, кліматом, безпекою.

3. Робототехніка: керування приводами, зчитування інформації з датчиків.

4. IoT пристрої: завдяки компактності та низькому споживанню енергії може слугувати вузлом для збору та передачі даних.

5. Автоматизація лабораторних установок і промислових процесів.

Важливі технічні особливості.

1. Енергоспоживання: Arduino Nano підтримує режими зниженого енергоспоживання, що дозволяє використовувати його у пристроях з автономним живленням.

2. Програмування: За допомогою USB-порту, плата підтримує стандартний bootloader Arduino, що виключає необхідність додаткового програматора.

3. Надійність: Використання перевіреної мікроконтролерної архітектури ATmega328P забезпечує стабільність роботи в широкому діапазоні умов.

4. Інтерфейси вводу-виводу: Широкі можливості для підключення різноманітних модулів і датчиків за допомогою цифрових і аналогових портів.

Arduino Nano є оптимальним вибором для розробки компактних, універсальних і енергоефективних систем управління та моніторингу. Його технічні характеристики і можливості інтерфейсів дозволяють реалізувати широкий спектр задач у сфері автоматизації, контролю параметрів навколишнього середовища, інтеграції в системи «Розумний будинок» та IoT.

Для організації передачі даних між мікроконтролерними пристроями та зовнішніми мережами використовуються різноманітні апаратні модулі, що забезпечують як провідне, так і бездротове з'єднання.

В якості основного провідного мережевого інтерфейсу для підключення мікроконтролера до Ethernet обрано модуль WIZnet W5500. Цей модуль призначений для забезпечення швидкої та стабільної комунікації в локальній мережі, а також для роботи з віддаленими серверами через стандартні інтернет-протоколи.

Модуль W5500 має вбудований TCP/IP стек, що значно спрощує реалізацію мережевого взаємодії без необхідності залучати додаткові ресурси мікроконтролера. Плата підтримує роботу як з протоколами передачі даних TCP

і UDP, так і з HTTP, що робить її універсальним рішенням для інтеграції у різні системи керування і моніторингу.

На рисунку 2.5 зображено зовнішній вигляд модуль WIZnet W5500 для підключення мікроконтролера до Ethernet.



Рисунок 2.5 — Модуль WIZnet W5500

Важливою особливістю модуля є підтримка одночасної роботи до восьми незалежних сокетів, що дозволяє обробляти декілька з'єднань паралельно. Розмір вбудованого буфера передачі та прийому даних складає 32 кБ, що забезпечує ефективну роботу навіть при значному обсязі мережевого трафіку.

До технічних особливостей модуля належать:

- живлення від 3.3 В або 5 В з підтримкою логічних рівнів 3.3 В (з толерантністю до 5 В);
- підтримка широкого спектру мережевих протоколів: TCP, UDP, ICMP, IPv4, ARP, IGMP, PPPoE;
- можливість одночасної роботи з 8 сокетом;
- вбудований 32 КБ буфер передачі даних.

Завдяки цим характеристикам модуль W5500 є оптимальним рішенням для реалізації надійної, швидкої та багатофункціональної Ethernet-комунікації в мікроконтролерних системах.

Для бездротової передачі даних широко використовується стандарт IEEE 802.11, більш відомий як Wi-Fi. Цей стандарт описує фізичний і каналний рівні

мережевої моделі OSI та забезпечує можливість створення бездротових локальних мереж із високою швидкістю передачі даних.

Під маркою Wi-Fi об'єднана група стандартів, що еволюціонували для забезпечення надійного зв'язку через радіоканали, включаючи популярні модифікації IEEE 802.11b, 802.11g та 802.11n (див.рис. 2.6).

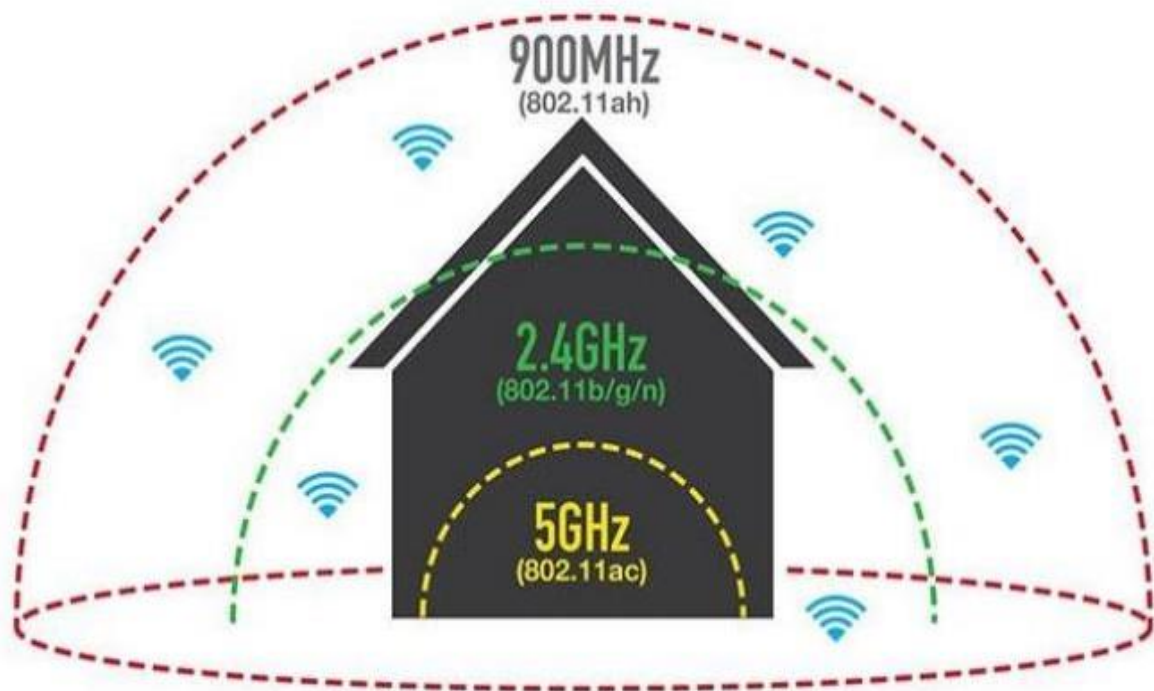


Рисунок 2.6 — Ethernet, модуль WIZnet W5500

Основні технічні параметри Wi-Fi включають:

- частотний діапазон роботи: 2.4 ГГц;
- ширина каналу передачі — 20 МГц;
- потужність передавача до +10 дБ;

Дальність покриття — до 150 метрів у відкритому просторі.

Wi-Fi мережа складається з двох типів пристроїв: клієнтів, оснащених бездротовими мережевими картами, та точок доступу, які забезпечують зв'язок між бездротовими і провідними мережами, виконуючи роль моста.

Завдяки широкому розповсюдженню та підтримці багатьма мікроконтролерними платформами Wi-Fi є популярним вибором для організації бездротового зв'язку в IoT пристроях, домашній автоматизації, системах

моніторингу та інших застосуваннях, де відсутність дротових з'єднань є критичною перевагою.

В сучасних мікроконтролерних системах для організації зв'язку використовуються різноманітні інтерфейси, що забезпечують ефективний обмін інформацією між пристроями та мережею. Крім Ethernet і Wi-Fi, широко застосовуються й інші протоколи передачі даних, які можна інтегрувати у систему в залежності від потреб проекту.

Окрім Ethernet і Wi-Fi, існують і інші протоколи, що можуть бути інтегровані до мікроконтролерних систем для різних сценаріїв використання:

Bluetooth та BLE (Bluetooth Low Energy) — забезпечують короткодіючий бездротовий зв'язок з низьким енергоспоживанням, що ідеально підходить для переносних пристроїв, сенсорів та мобільних додатків.

LoRa та LoRaWAN — технології для довготривалої бездротової передачі даних на великі відстані (до кількох кілометрів) з низькою швидкістю передачі, застосовуються в системах моніторингу міста, сільського господарства, екології.

ZigBee — протокол для створення мереж з багатьма пристроями з низьким енергоспоживанням, часто застосовується у системах розумного будинку.

Сучасні системи автоматизації та IoT пристрої передбачають передачу чутливих даних, тому питання безпеки є надзвичайно актуальним. Засоби передачі даних підтримують різні методи захисту інформації:

- шифрування (наприклад, TLS/SSL для Ethernet і Wi-Fi з'єднань);
- аутентифікація користувачів і пристроїв;
- використання безпечних протоколів передачі даних;
- захист від атак перехоплення або підміни пакетів.

Таким чином, грамотний вибір і налаштування засобів передавання даних — ключовий етап при розробці мікроконтролерних систем, що забезпечує їх ефективність, надійність і зручність експлуатації.

Основним завданням засобів передавання даних є забезпечення стабільної, надійної і безпечної комунікації між мікроконтролером та зовнішніми пристроями або серверами. Це дає змогу збирати інформацію з сенсорів,

передавати команди управління, отримувати оновлення і здійснювати моніторинг у режимі реального часу.

Важливими критеріями при виборі засобів зв'язку є:

- швидкість передачі даних: залежить від обраного протоколу та апаратної підтримки;
- дальність зв'язку: наприклад, Ethernet забезпечує локальне підключення, Wi-Fi — бездротове з покриттям до кількох сотень метрів;
- надійність та стійкість до перешкод: важливо в промислових чи побутових умовах;
- енергоспоживання: критичне для автономних пристроїв;
- сумісність із мікроконтролером і програмним забезпеченням.

2.6 Інтерфейси передачі даних в апаратній системі

Для зв'язку між мікроконтролером та периферійними пристроями в системі використовуються різні апаратні інтерфейси, які забезпечують надійну передачу даних. Основними інтерфейсами є Inter-Integrated Circuit (I²C), Serial Peripheral Interface (SPI), а також одноканальний інтерфейс передачі даних (Single-Wire Interface), що застосовуються у датчиках BMP280, DHT11/22, DS18B20 та модулі Ethernet WIZNet W5500.

2.6.1 Інтерфейс I²C

I²C (Inter-Integrated Circuit) — це дволінійна послідовна шина даних, розроблена для обміну інформацією між інтегральними схемами в електронних пристроях. Вона використовує дві основні лінії:

- SDA (Serial Data Line) — лінія передачі даних;
- SCL (Serial Clock Line) — лінія тактового сигналу.

Ці дві лінії є двонаправленими, і до них підключається кілька пристроїв. Кожен пристрій у мережі має унікальну адресу, що дозволяє вести обмін даними між ведучим (master) і веденим (slave) без конфліктів.

Робота шини I²C базується на протоколі "ведучий-ведений", де:

- ведучий формує тактовий сигнал SCL і ініціює передачу даних;
- ведений приймає або надсилає дані у відповідь на запит ведучого.

Передача починається з передачі адреси цільового пристрою. Якщо адреса співпадає з адресою веденого, він відповідає та приймає або передає дані. Тактовий сигнал завжди генерує ведучий, що дозволяє синхронізувати обмін даними.

Основні переваги I²C:

- мінімальна кількість ліній, що підвищує компактність системи;
- можливість підключення численних пристроїв на одну шину;
- простота апаратної реалізації та широке поширення.

В нашій системі датчик атмосферного тиску BMP280 використовує інтерфейс I²C для передачі показників до мікроконтролера Arduino через пари ліній SDA і SCL.

2.6.2 Інтерфейс SPI

SPI (Serial Peripheral Interface) — це високошвидкісний синхронний послідовний інтерфейс, який використовується для обміну даними між мікроконтролером та периферійними пристроями. Основними лініями SPI є:

- MOSI (Master Out Slave In) — лінія передачі даних від ведучого (master) до веденого (slave);
- MISO (Master In Slave Out) — лінія передачі даних від веденого до ведучого;
- SCK (Serial Clock) — тактовий сигнал, який формує ведучий;
- SS (Slave Select) — вибір конкретного веденого пристрою (активний низький сигнал).

Відмінні риси SPI:

- повно-дуплексний режим передачі, тобто одночасна передача та прийом даних;
- висока швидкість обміну інформацією;

— простий протокол без адресації — керування веденими здійснюється через лінію SS;

— можливість підключення кількох ведених пристроїв з використанням окремих ліній SS.

У нашій системі модуль Ethernet WIZNet W5500 підключений до мікроконтролера Arduino через SPI-інтерфейс. Мікроконтролер виступає ведучим, формує тактовий сигнал і керує обміном даних з Ethernet-модулем, що забезпечує передачу інформації через локальну мережу.

2.6.3 Одноканальний інтерфейс передачі даних (Single-Wire Interface)

Датчик DHT11/DHT22 використовує спрощений одноканальний інтерфейс передачі даних, який базується на єдиній лінії даних (SDA). Цей інтерфейс часто називають однопровідним (1-Wire) або одиночним каналом.

Особливості цього інтерфейсу:

- всі сигнали даних та керування проходять через одну лінію;
- для коректної роботи шини потрібен зовнішній підтягуючий резистор (типово 4,7 кОм), який підтримує лінію у високому стані у режимі очікування;
- мікроконтролер виступає ведучим і контролює обмін даними, ініціюючи звернення до датчика та отримуючи від нього інформацію;
- дані передаються у вигляді послідовності бітів, що формують 40-бітовий пакет, який включає значення температури, вологості та контрольну суму.

Обмін відбувається за жорстко визначеним протоколом: ведучий повинен дотримуватися строгої послідовності сигналів запиту і відповіді. Якщо послідовність порушується, датчик не відреагує, що унеможлиблює передачу даних.

Цей інтерфейс простий у реалізації, але вимагає точності в програмній частині для коректної синхронізації.

2.6.4 Інтерфейс 1-Wire для датчиків температури DS18B20

Датчики DS18B20 використовують інтерфейс 1-Wire, який дозволяє підключати кілька пристроїв до однієї лінії даних. Кожен датчик має унікальну 64-бітну адресу, що дає змогу розпізнавати і опитувати їх індивідуально.

Особливості інтерфейсу 1-Wire:

- один провід для передачі даних та живлення (можливий режим паразитного живлення);
- адресація та обмін даними відбуваються за допомогою спеціального протоколу;
- застосовується підтягуючий резистор для підтримки стабільного сигналу.

В системі підключення DS18B20 реалізовано на одній шіні з підключенням через резистор 4,7 кОм до живлення, що дозволяє ефективно опитувати кілька сенсорів температури по одній лінії.

Використання різних інтерфейсів передачі даних — I²C, SPI, одноканального Single-Wire та 1-Wire — забезпечує гнучкість, надійність і ефективність обміну інформацією в апаратній частині системи. Кожен інтерфейс вибирається залежно від вимог конкретного пристрою: I²C підходить для низькошвидкісних сенсорів з підтримкою кількох пристроїв, SPI забезпечує швидкий зв'язок з високошвидкісними модулями, а одноканальні інтерфейси ідеальні для простих датчиків з малою кількістю ліній підключення.

2.7 Інтеграція обраних технологій в апаратну архітектуру

Апаратна частина розробленої системи включає низку електронних компонентів, які взаємодіють у реальному часі задля реалізації поставленого завдання моніторингу та передавання даних. Основу архітектури становлять наступні ключові елементи:

- мікроконтролер Arduino;
- мережева плата WIZNet W5500 (Ethernet);
- датчик атмосферного тиску BMP280;
- комбінований сенсор температури та вологості DHT11;

- цифрові температурні датчики DS18B20.
- блок живлення.

Центральним елементом системи управління є мікроконтролер Arduino (рисунок 2.7), що поєднує переваги AVR-архітектури з відкритою екосистемою. Arduino забезпечує зчитування даних з сенсорів, обробку інформації та організацію передачі на сервер за допомогою Ethernet-з'єднання.

Усі апаратні компоненти системи з'єднані відповідно до стандартних рекомендацій виробника. Взаємозв'язки між мікроконтролером Arduino, датчиками та модулем Ethernet:

- BMP280 — підключений до пінів A4 (SDA) та A5 (SCL) через I2C;
- DHT11 — цифровий вихід з'єднано з піном D2;
- DS18B20 — підключено до D3 через резистор підтягування 4.7 кОм до Vcc;
- W5500 — підключено через SPI: MOSI (D11), MISO (D12), SCK (D13), CS (D10).

Для зв'язку з мережею Інтернет використовується спеціалізований модуль WIZNet W5500 (рисунок 2.8), що підтримує TCP/IP стек на апаратному рівні. Він з'єднується з мікроконтролером через SPI-шину, забезпечуючи стабільний та високошвидкісний обмін даними. Це дозволяє передавати інформацію на сервер або в хмарне середовище для подальшого аналізу чи збереження.

Вимірювання атмосферного тиску здійснюється за допомогою цифрового сенсора BMP280, який обмінюється даними з Arduino через інтерфейс I2C. Цей датчик має високу точність і стабільність вимірювань, що є важливим для метеорологічних і екологічних систем (рисунок 2.9).

Для контролю мікроклімату в приміщенні застосовується датчик DHT11, який підключено до цифрового входу Arduino. Цей сенсор забезпечує одночасне зчитування температури та відносної вологості повітря. Простота підключення та робота по одному піну роблять його популярним у побутових і навчальних проектах (рисунок 2.10).

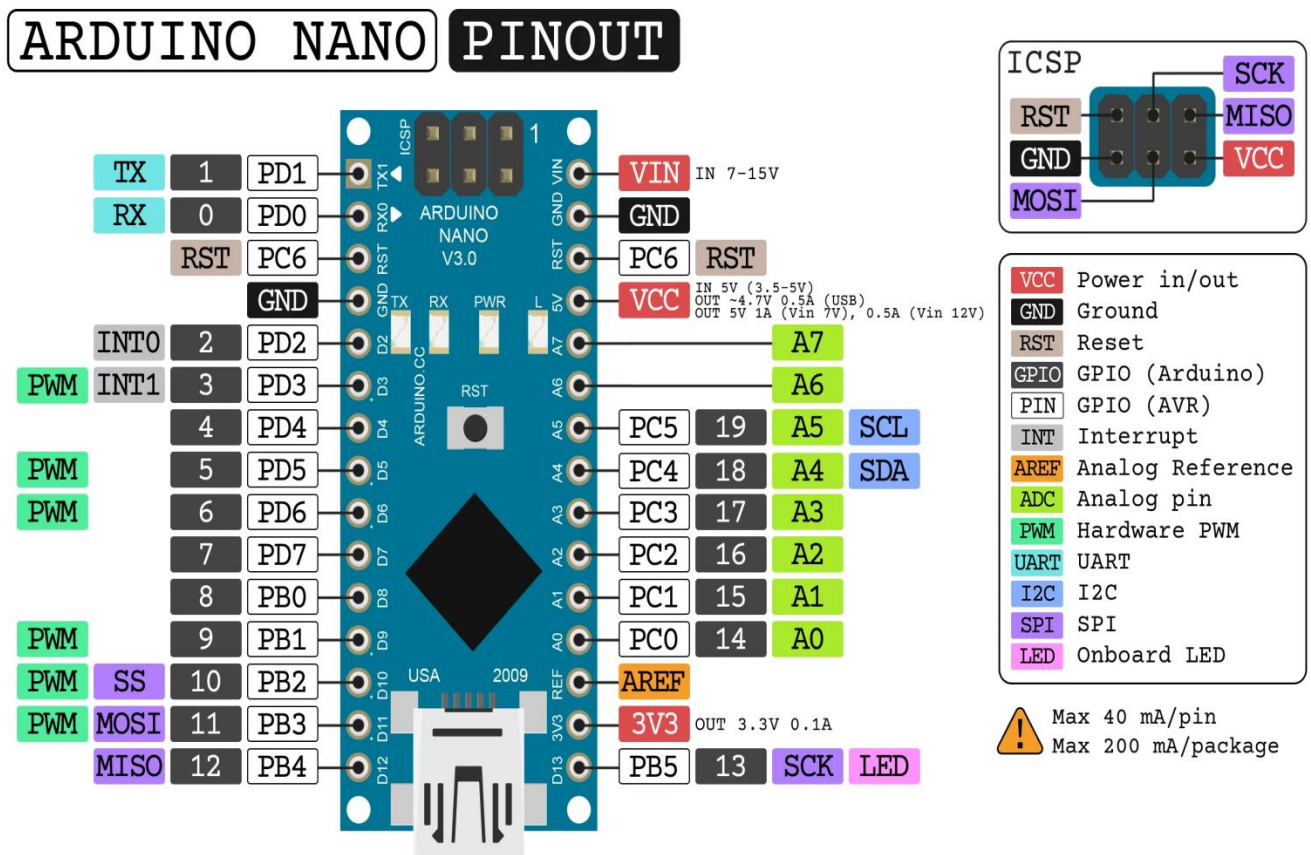


Рисунок 2.7 — Мікроконтролер Arduino

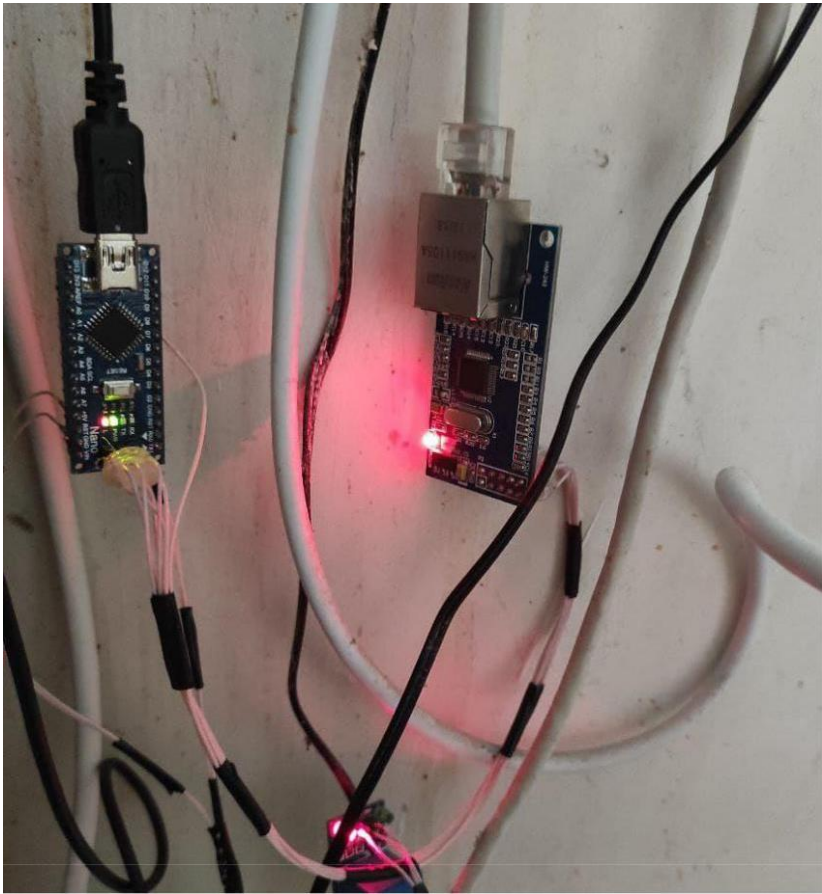


Рисунок 2.8 — Мікроконтролер, з'єднаний з Ethernet-модулем

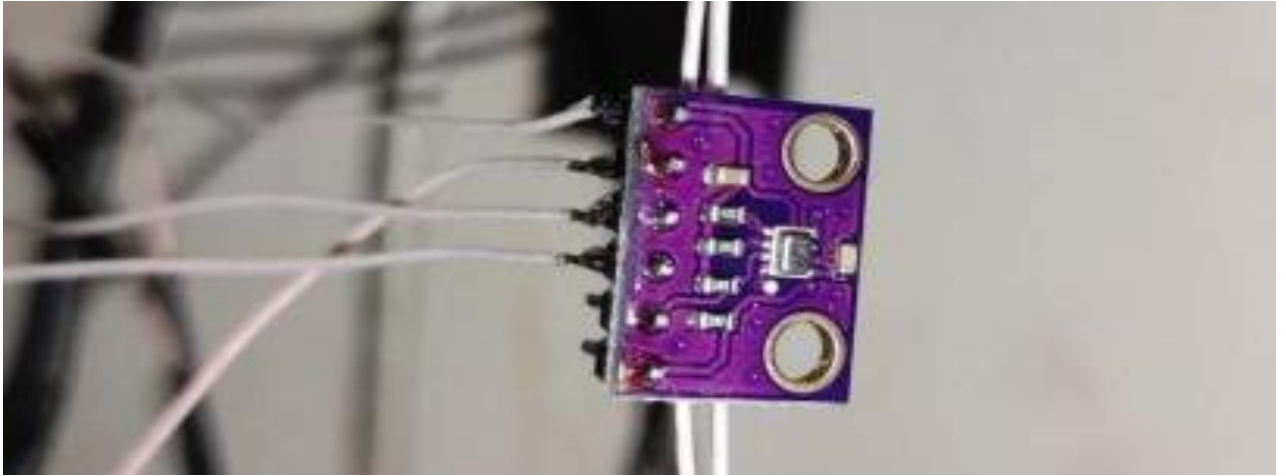


Рисунок 2.9 — Датчик атмосферного тиску BME280

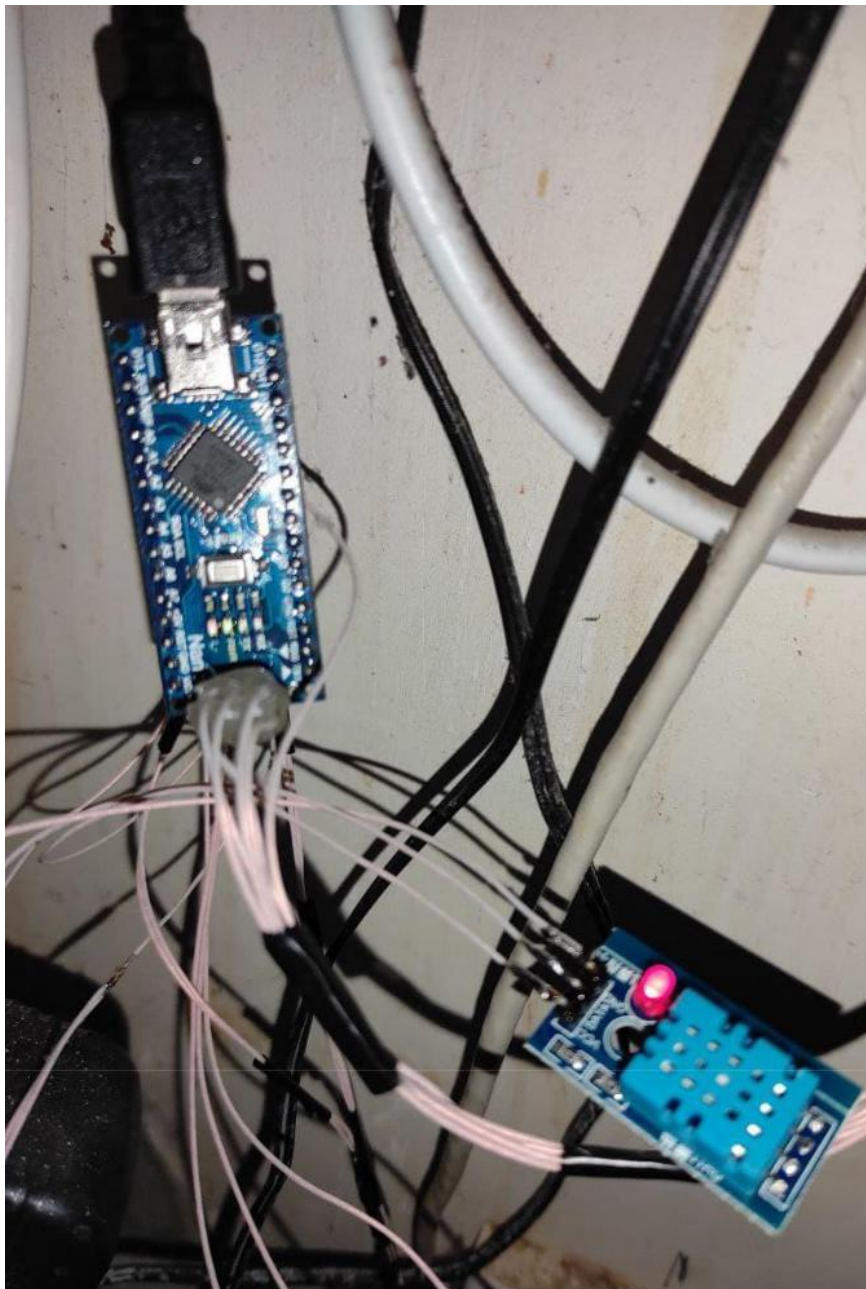


Рисунок 2.10 — Датчик температури й відносної вологості DHT11

Для точного вимірювання температури в кількох точках застосовуються цифрові датчики DS18B20 (рисунок 2.11), які працюють за протоколом 1-Wire. Усі сенсори підключено до однієї лінії, а їх ідентифікація в системі здійснюється за допомогою унікальних 64-бітних адрес. Це дозволяє легко масштабувати систему, додаючи нові сенсори без зміни схеми підключення.

Живлення системи забезпечується стабілізованим джерелом постійного струму з напругою 5 В (рисунок 2.12). Обраний блок сумісний з усіма компонентами системи за напругою та струмовими характеристиками, а також має компактні розміри для вбудованого монтажу. Він підключений до джерела безперебійного живлення (UPS), що дозволяє системі функціонувати навіть у випадках зникнення електроенергії в будинку.

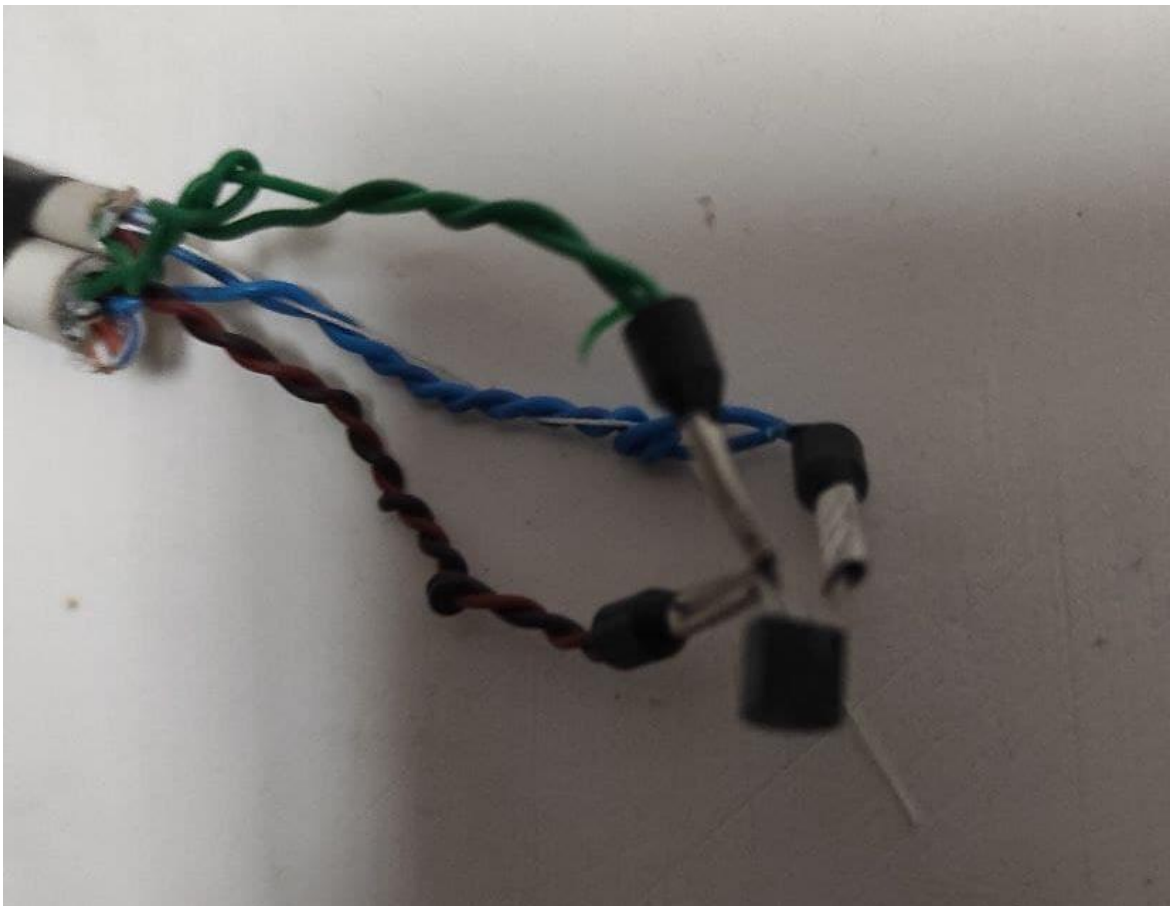


Рисунок 2.11 — Підключення цифрових датчиків температури DS18B20



Рисунок 2.12 — Блок живлення

У розділі обґрунтовано вибір компонентів апаратної частини системи моніторингу, зокрема мікроконтролера Arduino, сенсорів температури, вологості та тиску, а також Ethernet-модуля. Розглянуто типи інтерфейсів (I²C, SPI, 1-Wire, одноканальний) та їхню відповідність функціональним вимогам системи. Обрана архітектура є енергоефективною, доступною та придатною для інтеграції з програмною частиною.

3 ПРОГРАМНА ІНТЕГРАЦІЯ ІОТ-ПРИСТРОЮ ТА ЙОГО ІНТЕРАКЦІЯ З КОРИСТУВАЧЕМ

3.1 Побудова архітектури програмного забезпечення

У основі програмного рішення лежить принцип об'єктно-орієнтованого програмування. Весь функціонал розбито на незалежні класи-об'єкти, кожен із яких інкапсулює свої дані та поведінку.

Модуль мікроконтролера:

- збирає вимірювання з усіх датчиків (температури, вологості, тиску тощо);
- опрацьовує зовнішні команди, що надходять у вигляді JSON-повідомлень через MQTT-брокер;
- генерує керуючі сигнали для виконавчих пристроїв (реле, насосів, вентиляторів).

MQTT-брокер:

- виконує роль посередника для публікації/підписки;
- приймає дані від контролера та розсилає їх клієнтам (веб-інтерфейсу, додатку);
- пересилає команди від користувача на контролер.

Серверна частина:

- отримує та зберігає телеметрію в PostgreSQL;
- надає API для веб-інтерфейсу та інтеграції з іншими системами;
- слідкує за з'єднаннями та виконує відновлення після збоїв.

Клієнтський інтерфейс:

- візуалізує графіки і табличні звіти на основі даних із БД.
- дозволяє надсилати команди (нагрів, зволоження, сигналізацію) назад на контролер.

Діаграма прецедентів (Use Case) показана на рисунку 3.1.

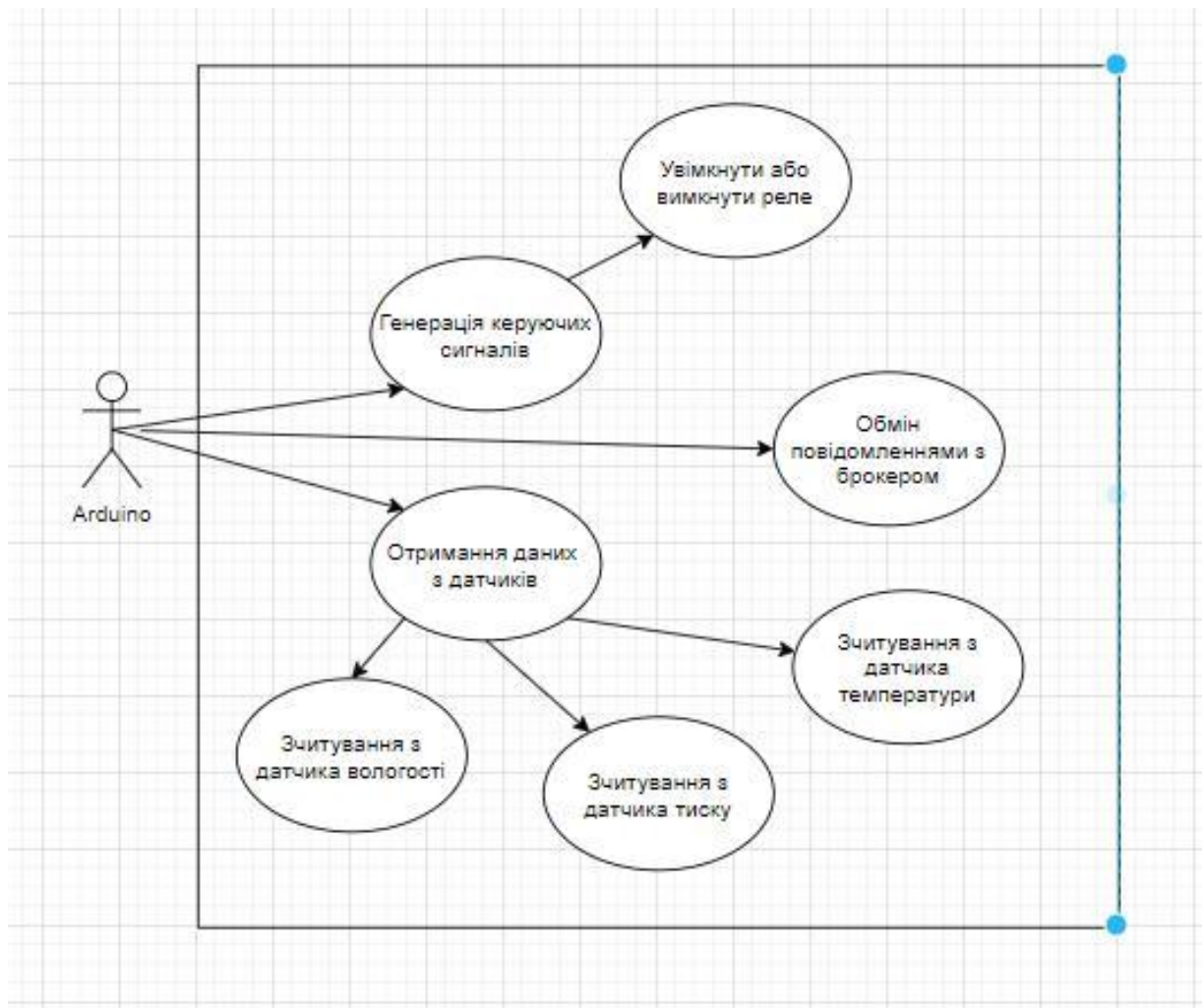


Рисунок 3.1 — Діаграма прецедентів (Use Case)

Збирання показань відбувається тауим чином: контролер періодично опитує всі підключені датчики.

Відправка даних: після зчитування формує MQTT-повідомлення і публікує в топік.

Отримання команд: контролер підписаний на топік команд, обробляє вхідні JSON-повідомлення.

Генерація керуючих сигналів: на основі команди перемикає виходи (реле, ШІМ).

Модель класів показана на рисунку 3.2.

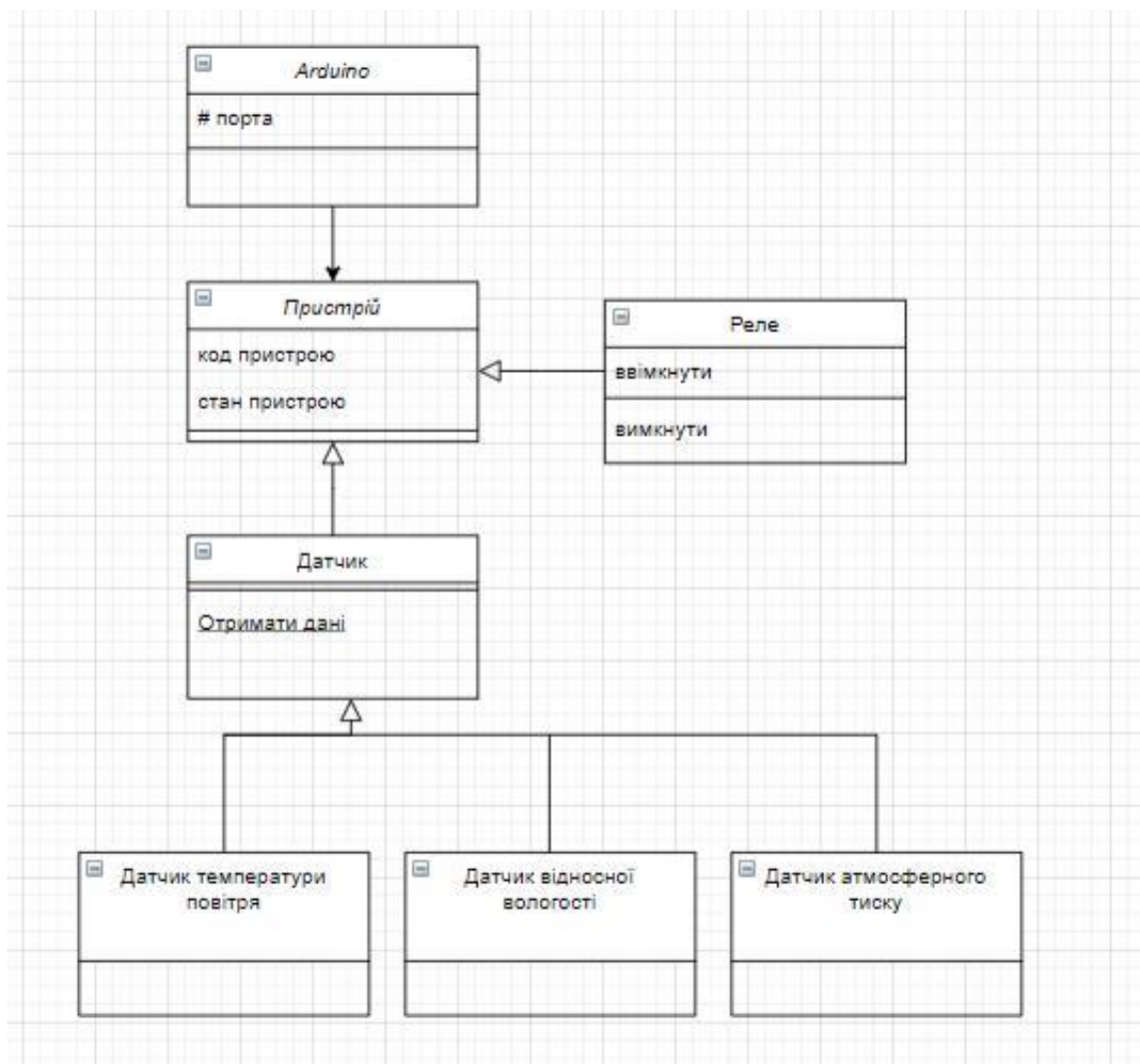


Рисунок 3.2 — Діаграма методів класів

Вона містить відповідні елементи:

Device (абстрактний)

— має метод read() для сенсорних пристроїв та toggle() для виконавчих.

Sensor : Device

— класи-нащадки: TempSensor, HumiditySensor, PressureSensor;

— реалізують read() та повертають числові значення.

Actuator : Device

— наприклад, Relay, FanController;

— методи on(), off(), setLevel(int).

MQTTClient

— відповідає за підписку/публікацію;

— при вхідному повідомленні викликає у відповідному об'єкті метод обробки.

DatabaseManager:

— інкапсулює роботу з PostgreSQL (пул з'єднань, запити INSERT/SELECT);

— забезпечує автоматичне відновлення після збоїв з'єднання та черговою обробкою накопичених запитів.

Обробка зовнішніх подій:

програмне переривання від MQTTClient спрацьовує на прийом нового повідомлення;

контролер тимчасово призупиняє циклічні вимірювання, парсить JSON та передає команду виконавцю.

Після виконання команди генерує валідаційний прапор (успішно/помилка) і миттєво відправляє зворотне підтвердження в брокер.

Цикл вимірювань відновлюється одразу після завершення обробки команди.

Зберігання даних у БД

Для кожного сенсора виділено окрему таблицю з полями:

- timestamp (час вимірювання);
- value (показник);
- status (стан пристрою: OK / ERROR).

DatabaseManager відкриває пул підключень (конфігурація: логін, пароль, хост, порт). Підтримує кілька одночасних клієнтських з'єднань. Автоматично відновлює зв'язок після перезавантаження СУБД та відправляє всі накопичені запити.

3.2 Реалізація обміну даними за допомогою MQTT

MQTT (Message Queuing Telemetry Transport) — легковажний протокол «видавець-підписник», спеціально розроблений для ресурс-обмежених IoT-

пристроїв. Він працює поверх TCP/IP, використовуючи за замовчуванням порт 1883 (для незашифрованого з'єднання) або 8883 (через SSL/TLS) (рисунок 3.3).

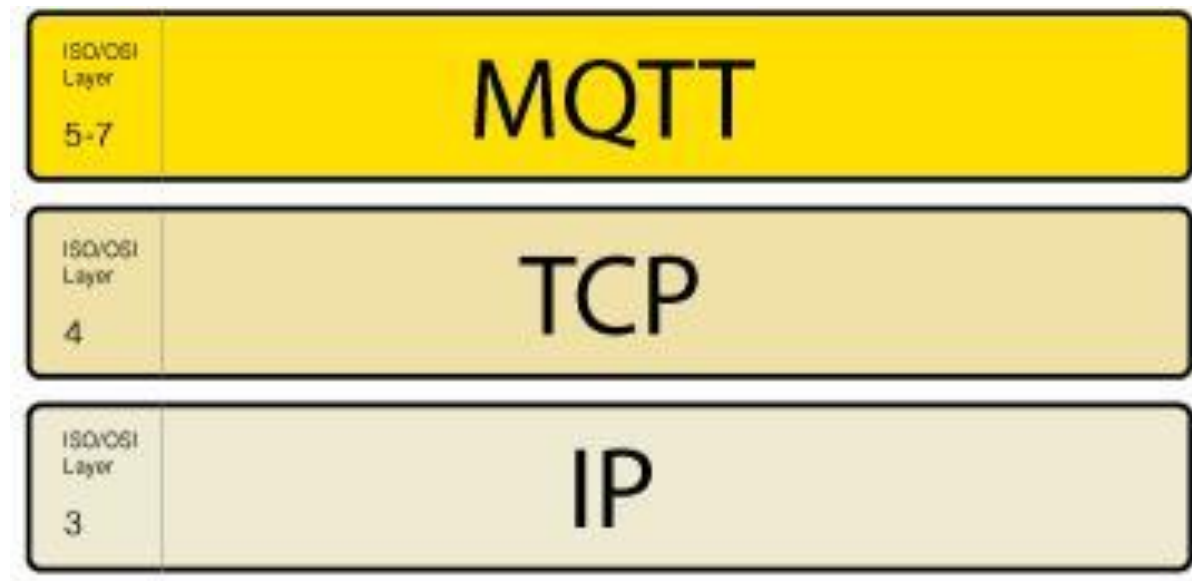


Рисунок 3.3 — Ієрархічна модель OSI

Нижчевказано ключові рольові модулі MQTT-системи

1. Broker — це ентральна станція, що приймає всі повідомлення від видавців (publisher) і розсилає їх відповідним підписникам (subscriber). Відповідає за аутентифікацію клієнтів, фільтрацію тем (topic) та реалізацію політик QoS.

2. Publisher — пристрій чи модуль, який публікує дані в певний топик, не відслідковує конкретних підписників і надсилає лише брокеру.

3. Subscriber — пристрій чи програма, що підписується на один або кілька топиків і отримує лише ті повідомлення, які відповідають його підписці.

Топіки й адресація:

— ієрархічна структура: теми розділяються слешем «/», наприклад home/kitchen/temperature;

— однорівневі (home/+/temperature) та багаторівневі (home/#) підстановки дозволяють гнучко визначати групи повідомлень.

Діаграма ієрархії топиків MQTT показана на рисунку 3.4.

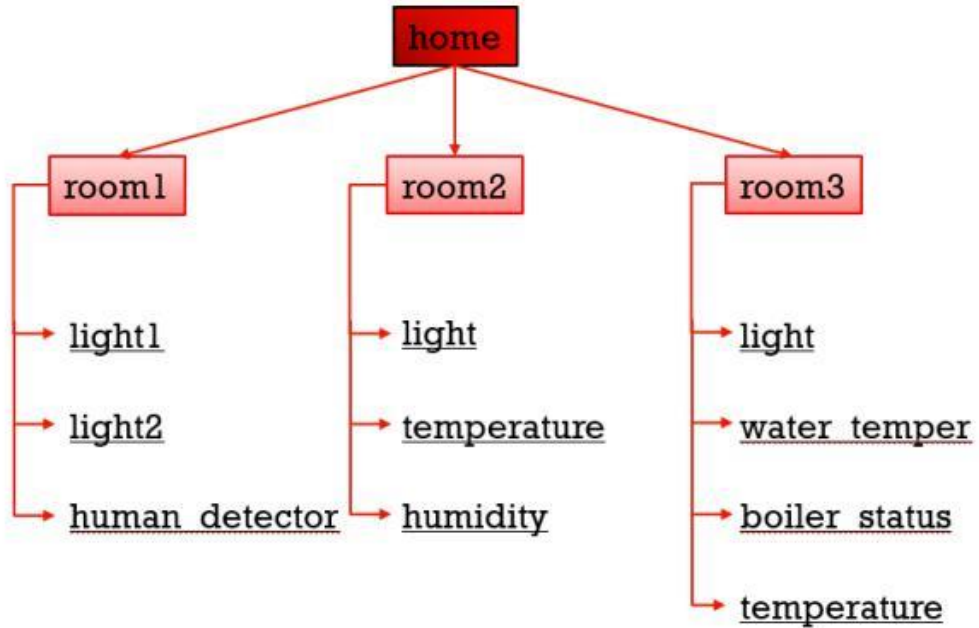


Рисунок 3.4 — Діаграма ієрархії топиків MQTT

Типи MQTT-повідомлень наведені в таблиці 3.1.

Таблиця 3.1 — Основні MQTT-команди

Повідомлення	Призначення
CONNECT	Встановити з’єднання з брокером
CONNACK	Підтвердження підключення
PUBLISH	Опублікувати повідомлення в топик
PUBACK/PUBREC/PUBREL/PUBCOMP	Кроки підтвердження доставки при QoS 1–2
SUBSCRIBE	Підписатися на один або декілька топиків
SUBACK	Підтвердження підписки
UNSUBSCRIBE	Відписатися від тем
UNSUBACK	Підтвердження відписки
DISCONNECT	Розірвати з’єднання

Рівні якості обслуговування (QoS):

- QoS 0 («At most once»): повідомлення надсилається один раз без підтвердження. Може бути втрачено, але ресурс-заощадливо;
- QoS 1 («At least once»): повідомлення гарантовано доставлене мінімум один раз. Видавець отримує підтвердження (PUBACK);
- QoS 2 («Exactly once»). Найбезпечніший режим: послідовність PUBREC → PUBREL → PUBCOMP гарантує унікальну доставку без дублікатів.

Захищені з'єднання

- Підтримка SSL/TLS на порті 8883 для шифрування трафіку та аутентифікації клієнтів.
- Опціональна авторизація за логіном/паролем або використання сертифікатів.

Переваги MQTT для IoT

1. Невеликий «хедер»: мінімальний накладний трафік у порівнянні з HTTP.
2. Низькі обчислювальні вимоги: підходить для слабких MCU з обмеженою пам'яттю.
3. Гнучка архітектура «видавець-підписник»: клієнти не знають один про одного, спілкуються лише через брокера.
4. Стійкість до неполадок мережі: підтримує повторну передачу та черги повідомлень.

Потік обміну

1. Publisher формує повідомлення з даними (наприклад JSON) і надсилає через PUBLISH до брокера, вказавши відповідний топик.
2. Broker отримує повідомлення, перевіряє, які підписники очікують дані за цим топиком, і розсилає їм копії повідомлення.
3. Subscriber приймає повідомлення і обробляє його в програмі (збереження в БД, оновлення UI, генерація сигналів тощо).

Такий підхід із MQTT гарантує оперативний і надійний обмін даними між мікроконтролером і серверною/клієнтською частиною IoT-системи, забезпечує гнучке масштабування та стійкість до збоїв мережевого зв'язку.

3.3 Налаштування середовища розробки та програмування контролера

Середовище розробки Arduino IDE складається з трьох основних областей: текстового редактора, консолі виводу повідомлень і серійного монітору. У редакторі програміст пише свій скетч на мові C/C++, де працює підсвічування синтаксису та автодоповнення базових команд. Після натискання кнопки «Verify» IDE переводить код у байт-код, а «Upload» прошиває його безпосередньо в пам'ять плати через USB-інтерфейс. Область виведення повідомлень показує результати компіляції — повідомлення про помилки чи попередження, а серійний монітор дозволяє переглядати дані, які Arduino передає через `Serial.print()`, і навіть відправляти текст назад у контролер під час виконання програми.

У головному меню Arduino IDE є розділи «File», «Edit», «Sketch», «Tools» та «Help». Через «Tools» обираються модель плати (наприклад, Arduino Nano чи ESP32), послідовний порт і завантажуються додаткові бібліотеки з офіційного менеджера. Інтерфейс працює на Windows, macOS та Linux, а його встановлення займає лічені хвилини. Окрім базових можливостей, у IDE є «Library Manager» для швидкого пошуку та підключення бібліотек датчиків, дисплеїв і протоколів зв'язку, а також «Board Manager» для додавання підтримки нестандартних плат і програматорів.

Arduino IDE поширюється під вільною ліцензією, має відкритий вихідний код і жодних ліцензійних обмежень. Це робить її безкоштовною для установлення та використання. Велика кількість готових прикладів у меню «Examples» дозволяє швидко розпочати роботу з новим сенсором або модулем. Бібліотечна екосистема налічує тисячі пакетів, від простих драйверів під DHT11 до складних стеків MQTT чи HTTP, що значно прискорює розробку.

На рисунку 3.5 зображено Середовище розробки Arduino IDE.



Рисунок 3.5 — Середовище розробки Arduino IDE

Щоб почати роботу з новим датчиком, спочатку потрібно через «Library Manager» встановити відповідний пакет (наприклад, «DHT sensor library» або «Adafruit BMP280»). Після цього у скетчі додається директива `#include`, і можна використовувати клас і методи бібліотеки без ручного налаштування низькорівневих протоколів. Підключившись до обраного COM-порту, розробник може вивантажити код у контролер та спостерігати за результатами в серійному моніторі.

Arduino IDE легко масштабується під складніші проєкти та інтегрується з іншими інструментами: наприклад, плагін PlatformIO для Visual Studio Code забезпечує розширене автодоповнення, відлагодження й інтеграцію в CI/CD.

Завдяки вбудованій підтримці USB-прошивки, розробник може швидко додати до існуючої мережі ще одну плату або перетворити її на вузол IoT, який використовує бібліотеку PubSubClient для обміну через MQTT.

У результаті Arduino IDE поєднує в собі простоту та доступність початкового рівня з достатньою гнучкістю для реалізації проєктів «розумного будинку», моніторингу мікроклімату або інших складних систем домашньої та промислової автоматизації.

3.4 Інтеграція з системами автоматизації типу MajorDoMo, Fibaro, OpenRemote

MajorDoMo — відкрита платформа для комплексного керування інженерними системами «розумного дому» та супутньої інформаційної інфраструктури. Вона легко встановлюється на звичайний ПК або домашній сервер і не потребує потужних апаратних ресурсів. Серед ключових сценаріїв застосування:

- мультимедіа: централізоване керування аудіо- та відеопристроями;
- мікроклімат: контроль температури, вологості та якості повітря;
- безпека: моніторинг сигналізації, відеоспостереження і датчиків руху;
- органайзер: планування подій, нагадування, звіти про стан систем.

Fibaro — бездротова система розумного будинку на базі протоколу Z-Wave. Вона автоматично сканує всі підключені датчики й відразу повідомляє користувача про будь-які події. Основні можливості Fibaro:

- групове керування: об'єднання пристроїв у логічні сценарії;
- клімат: включення/вимкнення опалення, вентиляції та кондиціонерів;
- освітлення: 50ім мери та реле для плавного регулювання яскравості;
- сповіщення: push-повідомлення, email та SMS про тривоги або зміну статусу.

OpenRemote виконує роль центрального сервера, що приймає команди з мобільних чи веб-клієнтів і передає їх далі іншим контролерам або сервісам. Його ключові складові:

- сервер “розумного дому”: сумісний з Windows, Linux, NAS (Synology тощо);
- онлайн IDE: вбудоване середовище для створення та тестування сценаріїв автоматизації;
- мобільний застосунок: динамічно завантажує інтерфейс і конфігурацію з сервера на смартфон користувача.

Підтримувані протоколи та обладнання включають: 1-Wire, KNX, Z-Wave, EnOcean, Bluetooth, VLC, MythTV, Lutron, Denon AVR, X10, Insteon, панелі AMX, FreeBox, XBMC та інші, що робить OpenRemote надзвичайно гнучким і масштабованим рішенням для інтеграції різноманітних пристроїв у єдину екосистему.

3.5 Алгоритм користування програмною системою моніторингу

Для обміну даними між контролером та веб-інтерфейсом використовується унікальний токен. Саме він слугує ключем для ідентифікації користувача в системі. Приклад формату токена показаний на рисунку 3.6.



Channel Token:token_ErBbYiKNAaT3hrdD

Рисунок 3.6 — Приклад токена

Щоб додати новий датчик у систему, достатньо вказати його MQTT-топiк, заповнити опис та вибрати тип даних (температура, вологість тощо). Натиснення кнопки +Resource відкриває форму додавання топiка (рис. 3.7).

Configured resources

hum

Resource name

+ Resource

humidity

Resource Description

number

any

Cancel

Save

Рисунок 3.7 — Додавання нового топіка

Після реєстрації датчика можна створити віджет для його візуалізації. Кнопка Add Widget викликає діалог, де задаються: назва, тип віджету (графік, індикатор, перемикач тощо), джерело даних і розмір панелі (рис. 3.8).

Add a Widget

Widget Type

Basic Value

Size

Medium (33% wide)

Title

Widget title

Channel

Arduino2

Resource

led

Cancel

Done

Рисунок 3.8 — Додавання віджету

Для побудови лінійного графіка обирається відповідний тип віджету та задається колір кривої. Приклад показу температури наведено на рисунку 3.9.

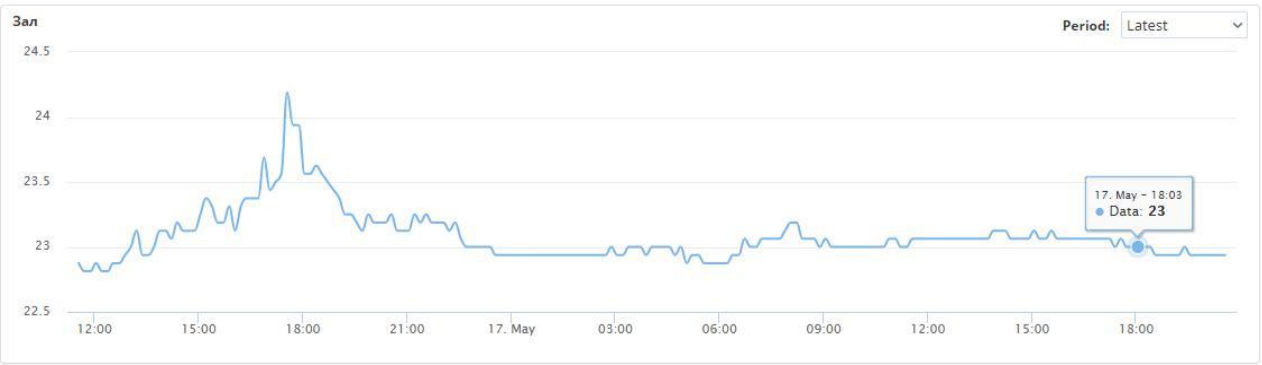


Рисунок 3.9 — Лінійний графік температури

Якщо потрібно порівняти кілька показників на одному полотні, застосовується віджет із підтримкою декількох ліній. Кожній лінії присвоюється власне ім'я й колір; результат можна побачити на рисунку 3.10.

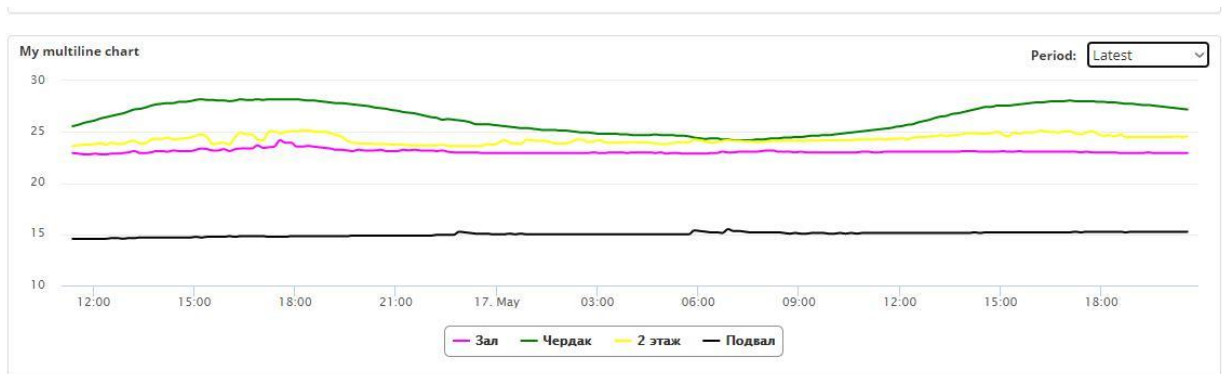


Рисунок 3.10 — Багатолінійний графік

Для керування зовнішніми пристроями (наприклад, реле) існує спеціальний віджет-реле, що відображає стан контакту та дозволяє вмикати або вимикати навантаження (рис. 3.11).



Рисунок 3.11 — Віджет для керування реле

Щоб швидко поглянути на останні вимірювання, використовуються компактні індикатори — маленькі плашки з поточним значенням кожного сенсора (рис. 3.12).

Для аналізу даних за конкретний період передбачено фільтр за датою: користувач обирає інтервал на календарі, і система показує лише ті записи, що потрапляють у вибраний проміжок. При великому часовому охопленні значення автоматично агрегуються у середньодобові показники (рис. 3.13).

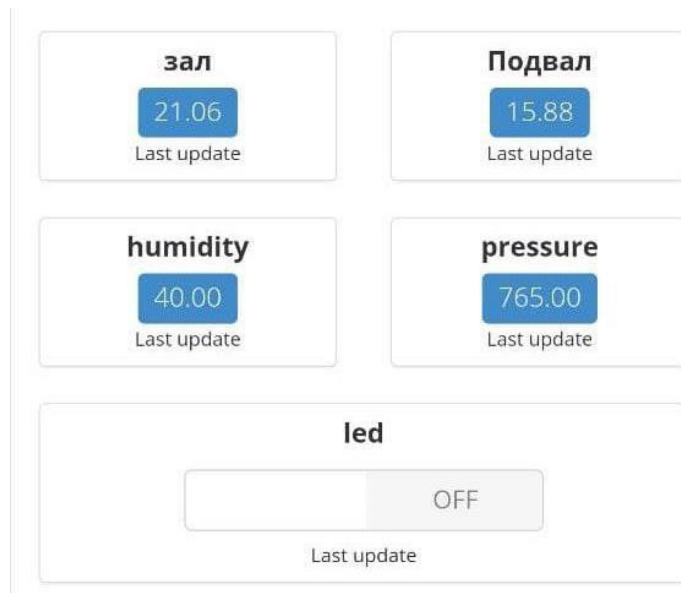


Рисунок 3.12 — Індикатори поточних значень



Рисунок 3.13 — Вибір інтервалу відображення

Розроблено інтуїтивно зрозумілий інтерфейс, що дозволяє керувати реєстрацією датчиків, створювати та налаштовувати віджети. Інтерфейс забезпечує гнучку візуалізацію показників мікроклімату у вигляді графіків, індикаторів та перемикачів, з підтримкою фільтрації за часом.

ВИСНОВКИ

У рамках виконаної роботи створено комплексний програмно-апаратний пристрій для безперервного моніторингу параметрів мікроклімату, що автономно вимірює температуру, вологість і атмосферний тиск, передає дані на віддалений сервер та інтегрується з існуючими системами «розумного будинку». Завдяки застосуванню моделі «видавець–підписник» на основі MQTT, об'єктно-орієнтованої архітектури ПЗ і перевірених апаратних рішень забезпечено гнучкість, масштабованість і надійність системи.

За результатами дослідження сформульовано такі висновки.

1. Проведений огляд існуючих IoT-рішень для моніторингу мікроклімату дав змогу виокремити ключові вимоги до нової системи: мобільність, модульність, багатоканальність та автоматизоване формування звітів.

2. Обґрунтовано вибір апаратної платформи на базі Arduino Nano, сенсорів DHT11, BMP280, DS18B20 і Ethernet-модуля W5500—які забезпечують точність вимірювань та стабільність передачі даних.

3. Розроблено об'єктно-орієнтовану архітектуру ПЗ із виділенням модулів для зчитування сенсорів, обміну через MQTT, зберігання в PostgreSQL та взаємодії з клієнтським інтерфейсом.

4. Реалізовано інтерфейс користувача з можливістю динамічного додавання датчиків, побудови лінійних і багатолінійних графіків, управління реле та відображення поточних значень.

5. Інтеграція із системами MajorDoMo, Fibaro та OpenRemote підтвердила універсальність рішення та його здатність працювати як у локальних, так і в мережових середовищах «розумного будинку».

Запропонована система готова до впровадження в проектах «розумного будинку», метеостанціях та промислових рішеннях, які вимагають високої надійності та гнучкості моніторингу мікроклімату.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Chodorow K. MongoDB: The Definitive Guide. 2nd ed. – Sebastopol, CA: O'Reilly Media, 2013. – 393 с.
2. Harrison B.L., Consolvo S., Choudhury T. Using multi-modal sensing for human activity modeling in the real world // Handbook of Ambient Intelligence and Smart Environments. – New York: Springer, 2010. – P. 463–478.
3. Michael D., Parrish A., Berg B. Introduction to Tornado. – Sebastopol, CA: O'Reilly Media, 2012. – 138 с.
4. Preece S.J., Goulermas J.Y., Kenney L.P., Howard D. A comparison of feature extraction methods for the classification of dynamic activities from accelerometer data // IEEE Transactions on Biomedical Engineering. – 2010. – Vol. 57, No 6. – P. 871–879.
5. Jiadi Y., Yingying C., Xiangyu X. Sensing vehicle conditions for detecting driving behaviors // SpringerBriefs in Electrical and Computer Engineering. – Cham: Springer, 2010. – P. 75–80.
6. Stoichkov R. Android smartphone application for driving style recognition // Multimodal Information Processing Group, Munich, 2013. – P. 3–7.
7. Fazeen M., Gozick B., Dantu R., Bhukhiya M., González M. Safe driving using mobile phones // IEEE Transactions on Intelligent Transportation Systems. – 2012. – Vol. 13, No 3. – P. 1068–1078.
8. Derick J., Trivedi M. Driving style recognition using a smartphone as a sensor platform // Proceedings of the 14th International IEEE Conference on Intelligent Transportation Systems (ITSC), Washington, DC, USA, 2011. – P. 1609–1615.
9. Ferrer S., Ruiz T. Travel behavior characterization using raw accelerometer data collected from smartphones // Transportation Research Part C: Emerging Technologies. – 2014. – Vol. 44. – P. 295–305.
10. Zhongyang C., Jiadi Y., Yanmin Z., Yingying C., Minglu L. D3: Abnormal driving behaviors detection and identification using smartphone sensors // Proceedings

of IEEE International Conference on Sensing, Communication, and Networking (SECON), 2015. – P. 1–6.

11. Chen D., Cho K., Han S., Jin Z., Shin K. Invisible sensing of vehicle steering with smartphones // Department of Electrical Engineering and Computer Science, University of Michigan, 2015. – P. 13–20.

12. Dominguez A., Bailey J., Djermanović D. Saving data on Android: Learning Room, Firebase and SQLite with Kotlin // Razeware LLC, 2019. – 306 с.

13. Saiprasert C., Thajchayapong S., Pholprasit T., Tanprasert C. Driver behaviour profiling using smartphone sensory data in a V2I environment // Information Communication and Computing Research Unit, National Electronics and Computer Technology Center, Thailand, 2016. – P. 1–6.

14. Banks A., Gupta R. MQTT Version 3.1.1 Plus Errata 01. OASIS Standard, 2015. – 58 с.

15. Simon Monk. Programming Arduino: Getting Started with Sketches. 2nd ed. – Sebastopol, CA: O'Reilly Media, 2016. – 216 с.

16. Frank D. Astler. Arduino Cookbook: Recipes to Begin, Expand, and Enhance Your Projects. 3rd ed. – Sebastopol, CA: O'Reilly Media, 2014. – 560 с.

17. Adafruit Industries. Adafruit BMP280 I2C or SPI Barometric Pressure & Altitude Sensor Library. – GitHub repository. URL: https://github.com/adafruit/Adafruit_BMP280_Library (дата звернення: 10.05.2025).

18. Majewski J., Hultgren K. Arduino Tutoring Guide. – Shrewsbury, UK: Packt Publishing, 2017. – 312 с.

19. Arduino. Arduino Reference. – офіційний сайт Arduino. URL: <https://www.arduino.cc/reference/en/> (дата звернення: 10.05.2025).

20. Benoit Blanchon. ArduinoJson 5: World's Best JSON Library for Embedded C++. – GitHub repository. URL: <https://github.com/bblanchon/ArduinoJson> (дата звернення: 10.05.2025).

21. Miles Burton. Reliability of IoT Devices: Design and Testing. – New York: Springer, 2019. – 210 с.

22. Marco Schwartz. Internet of Things with ESP8266. – Birmingham: Packt Publishing, 2018. – 286 с.
23. Dallas Semiconductor. DS18B20 Programmable Resolution 1-Wire® Digital Thermometer Datasheet. – Dallas: Maxim Integrated, 2014. – 12 с.
24. DHT11 Datasheet. DHT11 Humidity & Temperature Sensor. – Aosong Electronics. URL: <https://cdn-shop.adafruit.com/datasheets/DHT11.pdf> (дата звернення: 10.05.2025).

ДОДАТОК А

Технічне завдання

Міністерство освіти та науки України

Вінницький національний технічний університет

Факультет інформаційних технологій та комп'ютерної інженерії

ЗАТВЕРДЖУЮ

Завідувач кафедри ОТ

д.т.н., проф.

_____ О. Д. Азаров

“__” _____ 2025 р.

ТЕХНІЧНЕ ЗАВДАННЯ

на виконання бакалаврської кваліфікаційної роботи

«IoT-пристрій контролю параметрів мікроклімату»

08-54.БКР.016. 00.000 ПЗ

Науковий керівник к.т.н., доц. каф. ОТ

_____ Колесник І.С..

Студент групи 1КІ-23мс

_____ Сорока Є.Р.

Вінниця 2025

1 Підстава для виконання бакалаврської кваліфікаційної роботи (БКР)

1.1 Актуальність.

Необхідність отримання актуальної інформації про параметри мікроклімату виникає при контролі умов праці, виробництва та зберігання продукції у приміщеннях

2 Мета і призначення БКР

2.1 Мета роботи полягає у створення програмно-апаратного модуля для автоматичного моніторингу параметрів мікроклімату будівлі та навколишнього середовища.

2.2 Призначення розробки — виконання бакалаврської кваліфікаційної роботи із подальшим впровадженням та розвитком продукту.

3 Вихідні дані для виконання БКР

Огляд доступних систем моніторингу кліматичних умов, характеристики програмно-апаратного комплексу, MQTT

4 Етапи БКР та очікувані результати приведені у таблиці А.1.

Таблиця А.1 — Етапи виконання роботи

№	Назва етапів виконання дипломної роботи	Строк виконання	Підпис
1	Постановка задачі роботи	21.03.25	
2	Аналітичний огляд сучасних засобів контролю параметрів мікроклімату	21.03.25-30.03.25	
3	Обґрунтування технічного рішення	21.03.25-30.03.25	
4	Розробка апаратної платформи	31.03.25-13.04.25	
5	Програмна інтеграція IoT пристрою	14.04.25-27.04.25	
6	Оформлення пояснювальної записки	28.04.25-11.05.25	

7	Аналіз виконання роботи, висновки, додатки	28.04.25- 11.05.25	
8	Перевірка якості виконання бакалаврської роботи та усунення недоліків	13.05.25	

5 Матеріали, що подаються до захисту БКР: пояснювальна записка БКР, графічні і ілюстративні матеріали, протокол попереднього захисту БКР на кафедрі, відзив наукового керівника, відзив рецензента, протоколи складання державних екзаменів, анотації до БКР українською та іноземною мовами.

6 Порядок контролю виконання та захисту БКР

Виконання етапів розрахункової та графічної документації БКР контролюється науковим керівником згідно зі встановленими термінами. Захист БКР відбувається на засіданні Державної екзаменаційної комісії, затвердженою наказом ректора.

7 Вимоги до оформлення БКР

7.1 При оформлювання БКР використовуються:

— ДСТУ 3008: 2015 «Звіти в сфері науки і техніки. Структура та правила оформлювання»;

— ДСТУ 8302: 2015 «Бібліографічні посилання. Загальні положення та правила складання»;

— міждержавний ГОСТ 2.104-2006 «Єдина система конструкторської документації. Основні написи»;

— Методичні вказівки до виконання бакалаврських кваліфікаційних робіт зі спеціальності 123 «Комп'ютерна інженерія» (освітня програма «Комп'ютерна інженерія»). Кафедра обчислювальної техніки ВНТУ 2022;

— документами, на які посилаються у вище вказаних.

7.2 Порядок виконання БКР викладено в «Положення про кваліфікаційні роботи на першому (бакалаврському) рівні вищої освіти СУЯ ВНТУ-03.02.02-П.001.01:21»

ДОДАТОК Б

ПРОТОКОЛ ПЕРЕВІРКИ НАВЧАЛЬНОЇ (КВАЛІФІКАЦІЙНОЇ) РОБОТИ

Назва роботи:

IoT-пристрій контролю параметрів мікроклімату

Тип роботи: Бакалаврська кваліфікаційна робота

(БДР,МКР)

Підрозділ кафедра обчислювальної техніки

(кафедра,факультет)

Показники звіту подібності Strike Plagiarism

Оригінальність 99% Схожість 1%

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 виконання автором. 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)

Захарченко С.М.
(прізвище,ініціали)

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

Автор роботи

(підпис)

Сорока Є.Р.

(прізвище,ініціали)

Керівник роботи

(підпис)

Колесник І.С.

(прізвище,ініціали)

ДОДАТОК В

Графічна частина

ІоТ-ПРИСТРІЙ КОНТРОЛЮ ПАРАМЕТРІВ МІКРОКЛІМАТУ

ДОДАТОК В
Графічна частина
Діаграма методів класів

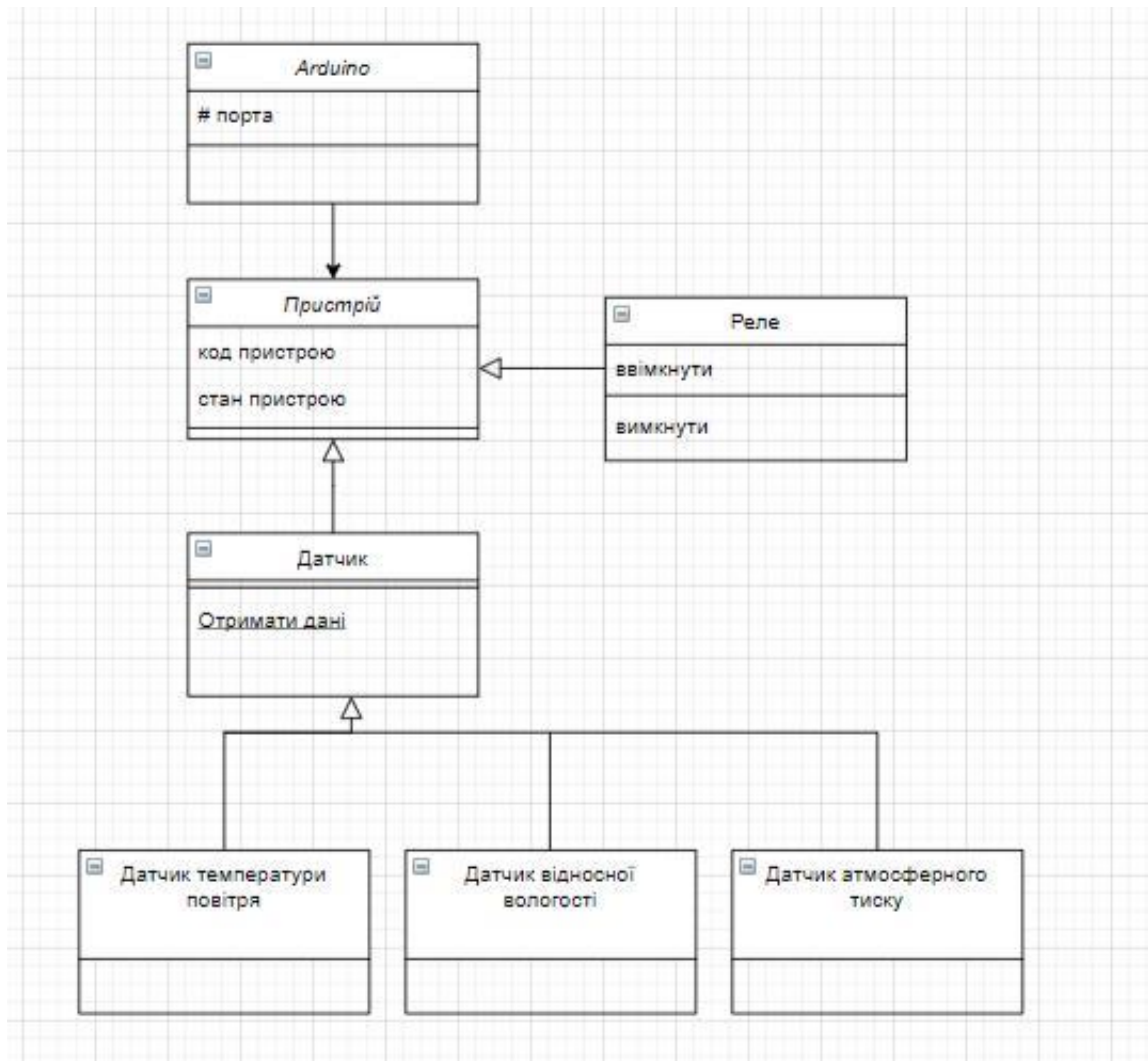


Рисунок В1 — Діаграма методів класів

ДОДАТОК Г

Ілюстративна частина

ІоТ-ПРИСТРІЙ КОНТРОЛЮ ПАРАМЕТРІВ МІКРОКЛІМАТУ

ДОДАТОК Г

Ілюстративна частина

Ethernet, модуль WIZnet W5500

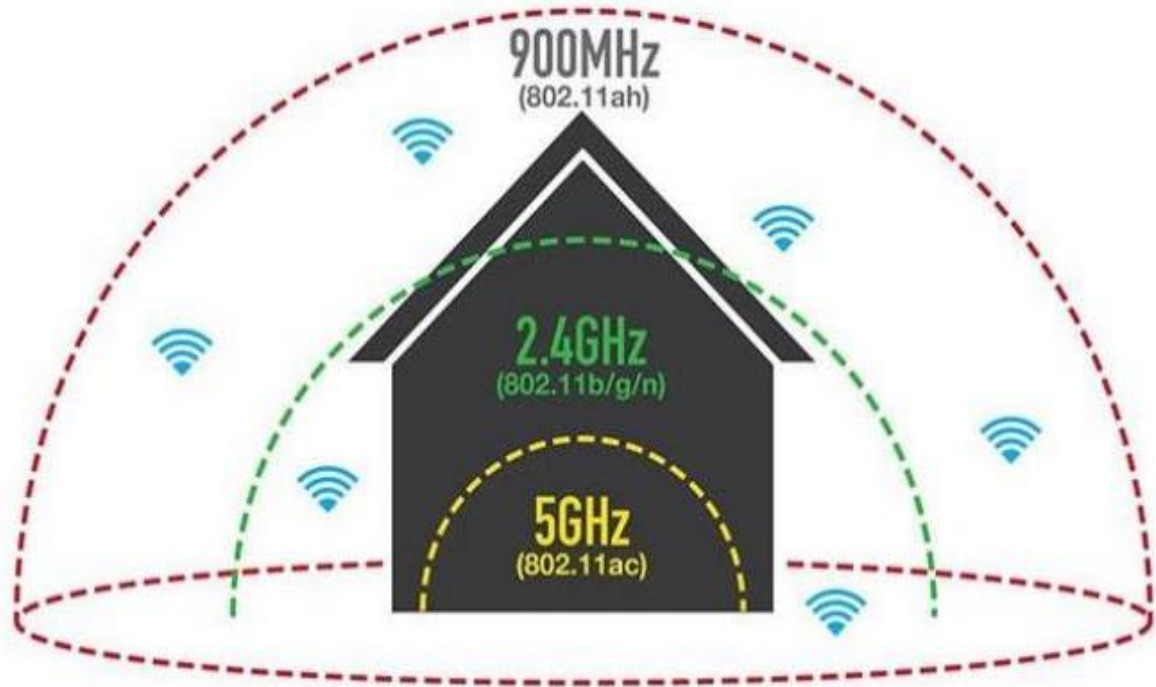


Рисунок Г1 — Ethernet, модуль WIZnet W5500

ДОДАТОК Д

Лістинг програмного забезпечення

```

// monitoring.ino

#include <SPI.h>
#include <Wire.h>
#include <OneWire.h>
#include <DallasTemperature.h>
#include <Ethernet.h>
#include <PubSubClient.h>
#include <ArduinoJson.h>
#include <Adafruit_BMP280.h>
#include <dht.h>

#define ONE_WIRE_BUS 5
#define DHT11_PIN 7

#define MQTT_BROKER "mqtt.beebotte.com"
#define TOKEN      "token_4g41ZmAfLWgAI29d"
#define CHANNEL    "Arduino2"

EthernetClient  ethClient;
PubSubClient   client(ethClient);
Adafruit_BMP280 bmp280;
dht            DHT;
OneWire        oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);

const int powerLed1 = 3;
const int powerLed2 = 4;

byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
IPAddress ip(192, 168, 1, 50);
IPAddress gateway(192, 168, 1, 1);
IPAddress subnet(255, 255, 255, 0);

DeviceAddress sensor1 = { 0x28, 0xB0, 0xA7, 0x01, 0x3B, 0x19, 0x01, 0xEB
};
DeviceAddress sensor2 = { 0x28, 0xB0, 0xB9, 0xE5, 0x3A, 0x19, 0x01, 0xD9
};
DeviceAddress sensor3 = { 0x28, 0x6E, 0xC8, 0x5A, 0x3B, 0x19, 0x01, 0x0E
};

```

```

DeviceAddress sensor4 = { 0x28, 0xB9, 0x79, 0xC1, 0x3E, 0x19, 0x01, 0x8D
};
DeviceAddress sensor5 = { 0x28, 0xE6, 0x9F, 0xEF, 0x3A, 0x19, 0x01, 0x3A
};

long lastMsg          = 0;
long lastReconnectAttempt = 0;

float temp1, temp2, temp3, temp4, temp5;

void setup() {
  Serial.begin(9600);
  sensors.begin();
  bmp280.begin();
  pinMode(powerLed1, OUTPUT);
  pinMode(powerLed2, OUTPUT);
  digitalWrite(powerLed1, LOW);
  digitalWrite(powerLed2, HIGH);

  client.setServer(MQTT_BROKER, 1883);
  client.setCallback(onMessage);

  if (Ethernet.begin(mac) == 0) {
    Ethernet.begin(mac, ip, gateway, subnet);
  }
  delay(1000);
  lastReconnectAttempt = 0;
}

void loop() {
  if (!client.connected()) {
    reconnect();
  }
  client.loop();
  if (millis() - lastMsg > 600000) {
    lastMsg = millis();
    readSensorData();
  }
}

void onMessage(char* topic, byte* payload, unsigned int length) {
  StaticJsonBuffer<128> jsonIn;
  JsonObject& root = jsonIn.parseObject(payload, length);
  if (!root.success()) {
    Serial.println("JSON parse failed");
  }
}

```

```

    return;
}
bool data = root["data"];
digitalWrite(powerLed1, data ? HIGH : LOW);
digitalWrite(powerLed2, data ? LOW : HIGH);
}

void readSensorData() {
    sensors.requestTemperatures();
    temp1 = sensors.getTempC(sensor1);
    temp2 = sensors.getTempC(sensor2);
    temp3 = sensors.getTempC(sensor3);
    temp4 = sensors.getTempC(sensor4);
    temp5 = sensors.getTempC(sensor5);

    float pressure = bmp280.readPressure() * 0.00750062; // hPa
    int chk = DHT.read11(DHT11_PIN);
    float humidity = DHT.humidity;

    publish("test", 22.0, true);
    publish("res", temp1, true);
    publish("first", temp2, true);
    publish("second", temp3, true);
    publish("third", temp4, true);
    publish("fourth", temp5, true);
    publish("pres", pressure, true);
    publish("hum", humidity, true);
}

void publish(const char* resource, float data, bool persist) {
    StaticJsonBuffer<128> jsonOut;
    JsonObject& root = jsonOut.createObject();
    root["channel"] = CHANNEL;
    root["resource"] = resource;
    if (persist) root["write"] = true;
    root["data"] = data;

    char buffer[128];
    root.printTo(buffer, sizeof(buffer));

    char topic[64];
    snprintf(topic, sizeof(topic), "%s/%s", CHANNEL, resource);
    client.publish(topic, buffer);
}

```

```

const          char          chars[]          =
"abcdefghijklmnpqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ123456789
0";

char id[17];

const char* generateID() {
    randomSeed(analogRead(A0));
    for (int i = 0; i < 16; i++) {
        id[i] = chars[random(strlen(chars))];
    }
    id[16] = '\0';
    return id;
}

void reconnect() {
    while (!client.connected()) {
        if (client.connect(generateID(), TOKEN, "")) {
            client.subscribe("Arduino2/led");
        } else {
            delay(5000);
        }
    }
}

```