

Харківський національний університет радіоелектроніки

Факультет навчально-науковий центр заочної форми навчання

Кафедра електронних обчислювальних машин

Рівень вищої освіти перший (бакалаврський)

Спеціальність 123 «Комп'ютерна інженерія»
(код і повна назва)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма Комп'ютерна інженерія
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

“ _____ ” _____ 20__ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Комлєву Андрію Валерійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Комп'ютерна мережа керування будинком з використанням IoT

затверджена наказом по університету від “ 05 ” травня 2025 р. № 72 СТз

2. Термін подання здобувачем роботи до екзаменаційної комісії 16 червня 2025 р.

3. Вхідні дані до роботи _____

комп'ютерна мережа

керування

розумний будинок

IoT

4. Перелік питань, що потрібно опрацювати у роботі _____

Теоретичні основи та архітектура системи

Технічна реалізація та програмне забезпечення

Експериментальні дослідження та аналіз результатів

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій 12 слайдів

6. Консультанти розділів роботи (заповнюється за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Строк / терміни виконання етапів роботи	Примітка
1	Отримання завдання та аналіз літератури	05.05.2025–10.05.2025	
2	Огляд існуючих аналогів	11.05.2025–03.06.2025	
3	Вибір алгоритмів	04.06.2025–06.06.2025	
4	Вибір програмних та апаратних засобів	07.06.2025–08.06.2025	
5	Програмна реалізація	09.06.2025–11.06.2025	
6	Аналіз отриманих результатів	12.06.2025–13.06.2025	
7	Оформлення записки	14.06.2025–16.06.2025	

Дата видачі завдання “ 05 ” травня 2025 р.

Здобувач


(підпис)

Керівник роботи


(підпис)

ст. викл. Владислав ДЯЧЕНКО

(посада, власне ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 50 с., 10 рис., 1 дод., 10 джерел.

ІНТЕРНЕТ РЕЧЕЙ, ІОТ, РОЗУМНА ТЕПЛИЦЯ, SMART GREENHOUSE, АВТОМАТИЧНЕ УПРАВЛІННЯ, ТЕПЛИЦЯ, ТЕМПЕРАТУРА ПОВІТРЯ, ВІДНОСНА ВОЛОГІСТЬ, ШВИДКІСТЬ ВІТРУ, ОСВІТЛЕНІСТЬ, АТМОСФЕРНИЙ ТИСК, ТЕМПЕРАТУРА ҐРУНТУ, ВОЛОГІСТЬ ҐРУНТУ, ARDUINO UNO R3, RASPBERRY PI.

Метою кваліфікаційної роботи є розробка мережі автономного та віддаленого керування «розумним будинком».

У ході виконання кваліфікаційної роботи розроблено функціональну комп'ютерну мережу для автономного та віддаленого керування інфраструктурою розумного будинку на основі технологій Інтернету речей. Запропонована система забезпечує надійний моніторинг та контроль параметрів мікроклімату, зокрема температури повітря, вологості, освітленості, атмосферного тиску та інших середовищних показників, що особливо актуально для керування тепличним господарством.

ABSTRACT

Bachelor's thesis: 50 pages, 10 figures, 1 appendices, 10 sources.

INTERNET OF THINGS, IoT, SMART GREENHOUSE, AUTOMATIC CONTROL, GREENHOUSE, AIR TEMPERATURE, RELATIVE HUMIDITY, WIND SPEED, ILLUMINATION, ATMOSPHERIC PRESSURE, SOIL TEMPERATURE, SOIL MOISTURE, ARDUINO UNO R3, RASPBERRY PI.

The major goal of this thesis is to develop a network for autonomous and remote control of a "smart home".

In order to a functional computer network was developed for autonomous and remote control of smart home infrastructure based on Internet of Things technologies. The proposed system ensures reliable monitoring and control of microclimate parameters, including air temperature, humidity, illumination, atmospheric pressure, and other environmental indicators, which is particularly relevant for greenhouse management.

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ	7
ВСТУП	8
1 ТЕОРЕТИЧНІ ОСНОВИ ТА АРХІТЕКТУРА СИСТЕМИ	10
1.1 Концепція Інтернету речей та її застосування у побуті	10
1.2 Аналіз існуючих рішень для автоматизації житлових приміщень	11
1.3 Обґрунтування вибору технологічної платформи	13
1.4 Трирівнева модель IoT-архітектури	15
1.5 Топологія мережі та принципи організації вузлів	16
1.6 Гібридна архітектура мехатронних та комп'ютерних компонентів	18
1.7 Інтеграція сенсорних та виконавчих підсистем	20
2 ТЕХНІЧНА РЕАЛІЗАЦІЯ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ	22
2.1 Сенсорна підсистема та збір даних	22
2.2 Система фільтрації сигналів та підвищення завадостійкості	23
2.3 Алгоритми управління та системи зворотного зв'язку	25
2.4 Протоколи передачі даних та комунікаційна інфраструктура	26
2.5 Архітектура програмної платформи Roof-Hatch	28
2.6 Реалізація багатопараметричних алгоритмів управління	30
2.7 Користувацькі інтерфейси та засоби візуалізації	31
3 ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ	33
3.1 Опис експериментального стенду	33
3.2 Методика проведення випробувань	34
3.3 Аналіз протоколів передачі даних DHT22	37
3.4 Програмна та апаратна реалізація	38
ВИСНОВКИ	42
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	43
ДОДАТОК А Графічний матеріал кваліфікаційної роботи	44

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

АР – атмосферний тиск

БД – база даних

ІР – Інтернет речей

КМ – комп'ютерна мережа

МК – мікроконтролер

ОС – операційна система

ПЗ – програмне забезпечення

СГ – сільське господарство

СМ – сенсорна мережа

ТГ – температура ґрунту

ТП – температура повітря

ШВ – швидкість вітру

ВСТУП

У сучасну добу взаємопов'язаних технологій, що отримала назву IoT (Інтернет речей), користувачі активно впроваджують розумні пристрої у повсякденне життя - від домашнього побуту та робочих просторів до транспортних засобів, медичних закладів і навіть носимих гаджетів. Завдяки поширенню хмарних технологій та масовому впровадженню бездротових мереж Wi-Fi і протоколу IPv6, галузь IoT демонструє стрімке зростання.

За даними дослідження мобільності від Ericsson, прогнозується, що до 2022 року загальна кількість IoT-пристроїв досягне вражаючої позначки у 18 мільярдів одиниць. Характерною особливістю цих підключених гаджетів є їхнє енергоощадне функціонування та значні технічні обмеження щодо обсягу пам'яті, швидкості обчислення, ємності батареї та фізичних розмірів. Такі пристрої зазвичай побудовані на основі мікроконтролерів із вузьким спектром можливостей.

Сфера IoT залишається неврегульованою галуззю без чітко визначених стандартів. Проблема фрагментації стає дедалі актуальнішою у розвитку IoT-екосистеми. Щоб програмісти могли ефективно розробляти застосунки для IoT-обладнання, критично важливо створити уніфіковану відкриту програмну платформу, яка б усунула фрагментацію та забезпечила сумісність.

Концепція відкритості в IoT охоплює уніфіковані стандарти та взаємосумісність, спільноти розробників open-source, вільний доступ до наукових даних, відкритий обмін користувацькою інформацією та доступні програмні інтерфейси. Відкриті дані та API є ключовими для передачі інформації, зібраної сенсорами, до спільноти розробників застосунків. Хоча відкриті системи надають численні переваги для створення програмного забезпечення, постачальники несуть підвищену відповідальність за забезпечення захисту та приватності при використанні open-source рішень у

IoT-пристроях.

Метою кваліфікаційної роботи є розробка мережі автономного та віддаленого керування «розумним будинком».

Проект спрямований на розробку прототипу, який будується на підході «розумний будинок», який включає датчики та виконуючі механізми для керування теплицею. Крім того, користувачі інтелектуальної тепличної системи можуть зручно контролювати теплицю за допомогою мобільного застосунку. Система також повинна мати можливість автономного контролю таких функцій, як температура та зрошення.

1 ТЕОРЕТИЧНІ ОСНОВИ ТА АРХІТЕКТУРА СИСТЕМИ

1.1 Концепція Інтернету речей та її застосування у побуті

Інтернет речей являє собою парадигму сучасних інформаційних технологій, що передбачає створення глобальної мережі взаємопов'язаних фізичних об'єктів, оснащених сенсорами, актуаторами та засобами бездротової комунікації. Фундаментальна ідея IoT полягає у забезпеченні автономної взаємодії пристроїв без прямого втручання людини, що відкриває нові можливості для автоматизації широкого спектру процесів у промисловості, сільському господарстві, медицині та побутовій сфері.

Архітектурна основа IoT-систем ґрунтується на багаторівневій моделі, що включає фізичний рівень пристроїв, комунікаційний рівень передачі даних, рівень обробки та аналізу інформації, а також прикладний рівень користувацьких сервісів. Кожен рівень виконує специфічні функції та забезпечує певний ступінь абстракції для вищерозташованих компонентів системи.

У контексті розумного дому IoT-технології знаходять застосування у системах безпеки, управління освітленням, регулювання температури, моніторингу енергоспоживання та координації роботи побутової техніки. Сучасні розумні будинки інтегрують десятки або навіть сотні IoT-пристроїв, що створює складну екосистему взаємопов'язаних компонентів із різними протоколами комунікації, рівнями споживання енергії та вимогами до обчислювальних ресурсів.

Ключовими перевагами IoT-підходу до автоматизації житлових приміщень є можливість дистанційного моніторингу та управління, адаптивність до змінних умов експлуатації, масштабованість системи та потенціал для інтеграції алгоритмів штучного інтелекту. Водночас, впровадження IoT-рішень супроводжується певними викликами, пов'язаними

із забезпеченням кібербезпеки, сумісності різнорідних пристроїв та надійності функціонування у критичних ситуаціях.

Еволюція IoT-технологій характеризується переходом від простих систем дистанційного управління до інтелектуальних платформ, здатних до самонавчання та прогнозування поведінки користувачів. Сучасні тенденції розвитку включають інтеграцію з хмарними сервісами, використання технологій граничних обчислень (edge computing) та впровадження стандартів кібербезпеки спеціально розроблених для IoT-екосистем.

1.2 Аналіз існуючих рішень для автоматизації житлових приміщень

Сучасний ринок систем домашньої автоматизації представлений широким спектром комерційних та відкритих рішень, що відрізняються архітектурними підходами, функціональними можливостями та вартісними характеристиками. Провідні технологічні компанії, такі як Google, Amazon, Apple та Microsoft, розробили власні екосистеми розумного дому, що базуються на фірмових протоколах комунікації та хмарних платформах обробки даних.

Система Google Home інтегрує голосове управління з широким спектром IoT-пристроїв через протокол Google Assistant API, забезпечуючи централізоване управління освітленням, термостатами, системами безпеки та мультимедійними пристроями. Архітектура системи передбачає обробку команд у хмарному середовищі Google Cloud Platform із можливістю локального виконання критичних функцій через спеціалізовані контролери Nest Hub.

Amazon Alexa представляє альтернативний підхід, що базується на модульній архітектурі Skills та широкій підтримці сторонніх виробників IoT-обладнання. Система використовує протокол Zigbee для локальної комунікації та AWS IoT Core для хмарної обробки даних, що забезпечує високу масштабованість та гнучкість у інтеграції нових пристроїв.

Відкриті платформи, такі як Home Assistant, OpenHAB та Domoticz, надають альтернативу комерційним рішенням, забезпечуючи повний контроль над функціональністю системи та можливість локального зберігання даних без залежності від хмарних сервісів. Ці платформи підтримують десятки комунікаційних протоколів, включаючи Z-Wave, Zigbee, Wi-Fi, Bluetooth та спеціалізовані рішення для промислової автоматизації.

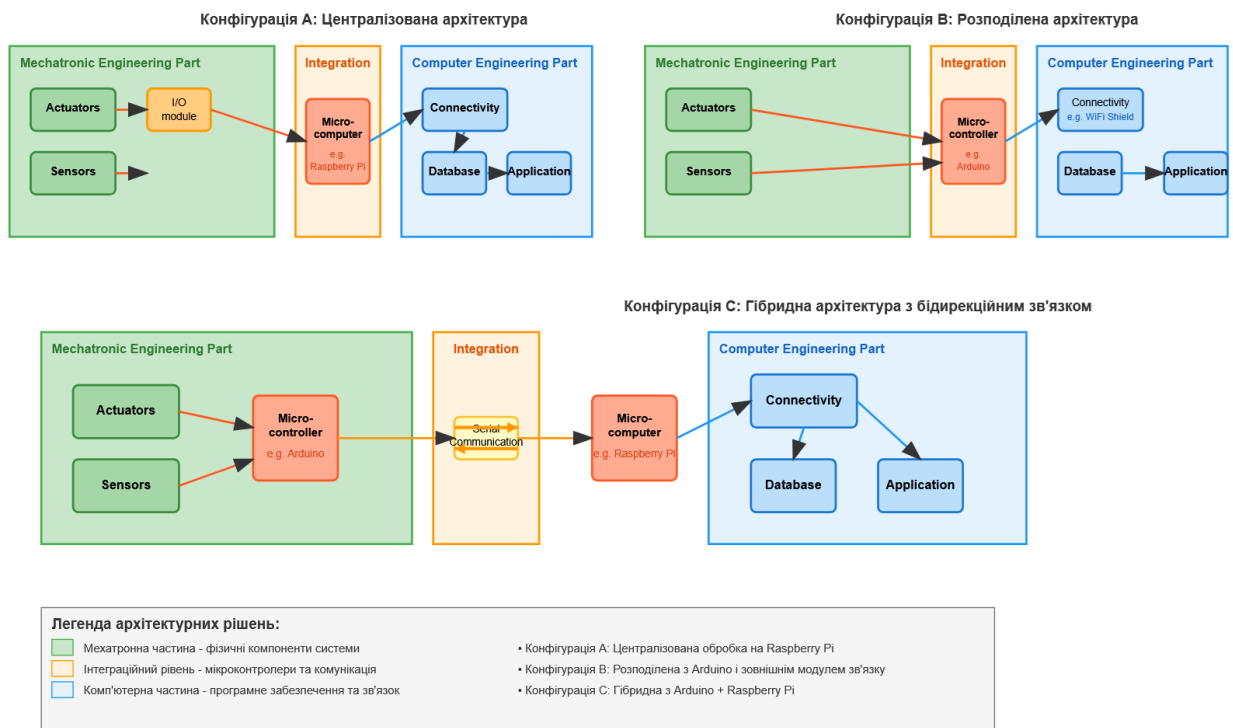


Рисунок 1.1 – Архітектури систем

Аналіз архітектурних особливостей існуючих систем виявляє два основних підходи: централізований, де всі рішення приймаються єдиним контролером, та розподілений, що передбачає автономне функціонування окремих підсистем із координацією через мережеві протоколи. Централізовані системи забезпечують кращу координацію між компонентами та спрощують реалізацію складних сценаріїв автоматизації, тоді як розподілені архітектури характеризуються вищою відмовостійкістю та меншими вимогами до пропускну здатності мережі.

Критичним аспектом оцінки існуючих рішень є аналіз їхньої енергоефективності та впливу на загальне споживання електроенергії будинком. Дослідження показують, що правильно налаштовані системи розумного дому можуть забезпечити економію енергії до тридцяти відсотків через оптимізацію роботи систем опалення, вентиляції та освітлення відповідно до реальних потреб мешканців.

1.3 Обґрунтування вибору технологічної платформи

Вибір технологічної платформи для реалізації системи управління розумним будинком визначається комплексом факторів, включаючи функціональні вимоги, обмеження по ресурсах, вартісні характеристики та перспективи масштабування (рисунок 1.2). Аналіз доступних апаратних платформ виявляє три основних категорії: мікроконтролерні системи, одноплатні комп'ютери та спеціалізовані IoT-процесори.

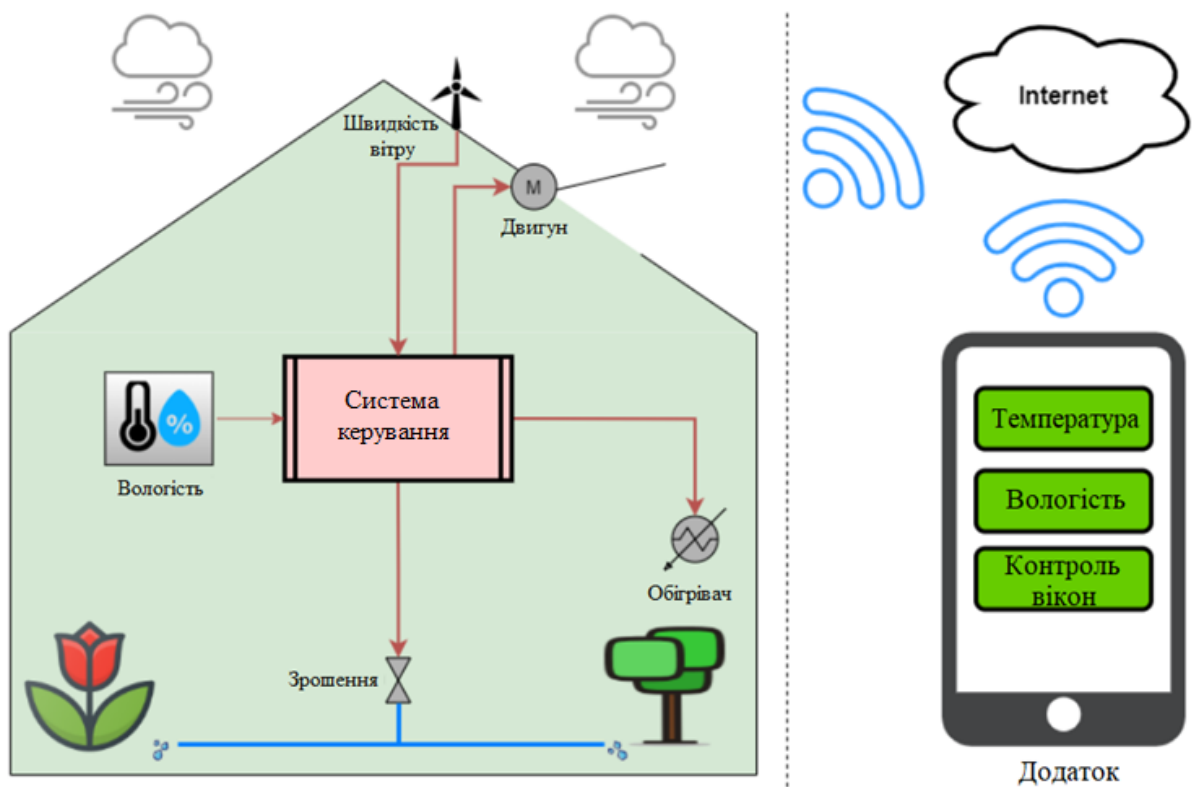


Рисунок 1.2 – Огляд системи, інтегрованої в теплиці

Мікроконтролерні платформи, такі як Arduino та ESP32, характеризуються низьким енергоспоживанням, реальним часом виконання операцій та відносно простою архітектурою програмування. Ці характеристики роблять їх оптимальним вибором для реалізації сенсорних вузлів та простих контролерів виконавчих пристроїв. Обмеженням мікроконтролерних систем є невеликий обсяг пам'яті та обчислювальна потужність, що ускладнює реалізацію складних алгоритмів обробки даних.

Одноплатні комп'ютери, представлені платформами Raspberry Pi, BeagleBone та NVIDIA Jetson, забезпечують значно вищу обчислювальну потужність та підтримку повноцінних операційних систем Linux. Це дозволяє реалізувати складні алгоритми машинного навчання, веб-сервери та системи управління базами даних безпосередньо на локальному рівні. Недоліком таких систем є відносно високе енергоспоживання та складність забезпечення роботи у реальному часі.

Спеціалізовані IoT-процесори, такі як серія ARM Cortex-M та RISC-V архітектури, представляють компромісне рішення, що поєднує енергоефективність мікроконтролерів з розширеними можливостями обробки даних. Ці платформи особливо ефективні для реалізації граничних обчислень та попередньої обробки сенсорних даних перед передачею до центральних серверів.

Для розробленої системи обрано гібридний підхід, що поєднує переваги різних технологічних платформ. Сенсорні вузли та контролери виконавчих пристроїв реалізовано на базі мікроконтролерної платформи Arduino через її простоту програмування, широку підтримку спільноти розробників та наявність великої кількості готових бібліотек для роботи з різноманітними сенсорами.

Центральний контролер системи базується на одноплатному комп'ютері Raspberry Pi, що забезпечує необхідну обчислювальну потужність для реалізації веб-інтерфейсу, системи управління базою даних та алгоритмів машинного навчання. Така архітектура дозволяє ефективно розподілити

обчислювальне навантаження між компонентами системи та забезпечити оптимальне співвідношення функціональності та енергоспоживання.

1.4 Трирівнева модель IoT-архітектури

Архітектурна організація розробленої системи базується на класичній трирівневій моделі IoT, що включає рівень сприйняття (perception layer), мережевий рівень (network layer) та прикладний рівень (application layer). Кожен рівень виконує специфічні функції та забезпечує певний ступінь абстракції для компонентів вищого рівня, що спрощує розробку та обслуговування системи.

Рівень сприйняття відповідає за збір первинної інформації про стан навколишнього середовища та виконання керуючих впливів на фізичні об'єкти. До складу цього рівня входять різноманітні сенсори температури, вологості, освітленості, якості повітря, а також виконавчі пристрої, такі як електромагнітні реле, серводвигуни, світлодіодні освітлювачі та системи вентиляції. Ключовою особливістю рівня сприйняття є його розподілений характер та вимоги до функціонування у реальному часі.

Технічна реалізація рівня сприйняття базується на мікроконтролерних модулях Arduino, що забезпечують інтерфейс між аналоговими сигналами сенсорів та цифровими протоколами передачі даних. Кожен сенсорний модуль оснащено локальною системою попередньої обробки сигналів, що включає аналого-цифрове перетворення, фільтрацію шумів та базову діагностику несправностей.

Мережевий рівень забезпечує передачу даних між компонентами системи та їхню інтеграцію з зовнішніми сервісами. Архітектура мережевого рівня включає локальну бездротову мережу для комунікації між сенсорними вузлами та центральним контролером, а також підключення до глобальної мережі Інтернет для дистанційного доступу та інтеграції з хмарними сервісами.

Для локальної комунікації використовується протокол XBEE, що забезпечує надійну передачу даних на відстані до ста метрів із автоматичним повторенням пакетів у випадку виявлення помилок. Топологія локальної мережі реалізована у вигляді зірки з центральним координатором, що спрощує маршрутизацію та забезпечує централізоване управління енергоспоживанням вузлів.

Прикладний рівень реалізує бізнес-логіку системи, включаючи алгоритми прийняття рішень, користувацькі інтерфейси та інтеграцію з зовнішніми сервісами. Архітектура прикладного рівня базується на модульному підході, що дозволяє незалежно розробляти та оновлювати окремі компоненти системи без впливу на функціонування інших модулів.

Центральним компонентом прикладного рівня є система управління базою даних, що забезпечує зберігання історичних даних про параметри мікроклімату, журналів подій системи та налаштувань користувачів. База даних реалізована на основі реляційної СУБД SQLite, що забезпечує необхідну продуктивність для обробки потоків даних від декількох десятків сенсорів при мінімальних вимогах до дискового простору.

1.5 Топологія мережі та принципи організації вузлів

Топологічна організація мережі IoT-пристроїв у розумному домі суттєво впливає на надійність, масштабованість та енергоефективність системи. Аналіз різних топологій виявляє переваги та недоліки зіркоподібних, кільцевих, mesh та гібридних архітектур у контексті специфічних вимог домашньої автоматизації.

Зіркоподібна топологія характеризується наявністю центрального координатора, що забезпечує комунікацію між усіма вузлами мережі. Перевагами такого підходу є простота маршрутизації, централізоване управління енергоспоживанням та можливість ефективної реалізації алгоритмів колективного прийняття рішень. Основним недоліком є

створення єдиної точки відмови, що може призвести до повної втрати функціональності системи у випадку несправності центрального вузла.

Mesh-топология передбачає створення самоорганізованої мережі, де кожен вузол може виконувати функції маршрутизатора для сусідніх пристроїв. Такий підхід забезпечує високу відмовостійкість та можливість автоматичного перемаршрутизування у випадку виходу з ладу окремих компонентів. Недоліками mesh-архітектури є складність управління, підвищене енергоспоживання через необхідність підтримання маршрутних таблиць та потенційні проблеми з масштабованістю у великих мережах.

Для розробленої системи (рисунок 1.3) обрано модифіковану зіркоподібну топологію з елементами резервування критичних функцій. Центральний координатор реалізовано на базі Raspberry Pi, що забезпечує необхідну обчислювальну потужність для обробки даних від всіх сенсорних вузлів та координації роботи виконавчих пристроїв. Додатково впроваджено механізм автоматичного перемикавання на резервний контролер у випадку виявлення критичних несправностей основного вузла.

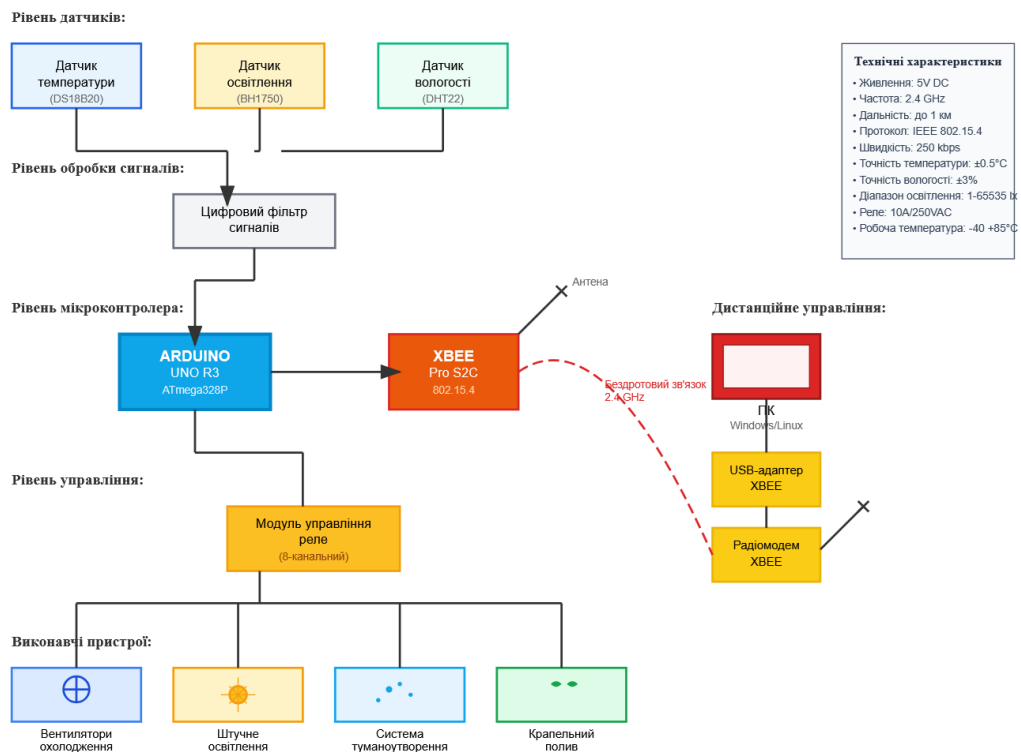


Рисунок 1.3 – Блок-схема автоматизованої схеми керування

Архітектура вузлів мережі базується на стандартизованому підході з використанням уніфікованих апаратних модулів та програмних інтерфейсів. Кожен сенсорний вузол включає мікроконтролер Arduino, радіомодуль XBEE, систему живлення та спеціалізовані сенсорні модулі відповідно до функціонального призначення. Така уніфікація спрощує виробництво, обслуговування та масштабування системи.

Принципи адресації у мережі базуються на ієрархічній схемі з використанням 64-бітних унікальних ідентифікаторів для кожного пристрою. Адресний простір організовано таким чином, що перші октети визначають тип пристрою та його розташування у будинку, а останні октети забезпечують унікальність у межах групи однотипних пристроїв. Така схема адресації спрощує автоматичне виявлення та конфігурування нових пристроїв при розширенні системи.

Механізми забезпечення якості обслуговування (QoS) у мережі включають пріоритизацію трафіку відповідно до критичності передачуваної інформації. Найвищий пріоритет мають повідомлення про аварійні ситуації та команди екстреного відключення обладнання. Середній пріоритет призначено для регулярних команд управління та підтвердження їх виконання. Найнижчий пріоритет має історична інформація та діагностичні дані, передача яких може бути відкладена без критичного впливу на функціонування системи.

1.6 Гібридна архітектура мехатронних та комп'ютерних компонентів

Концептуальною особливістю розробленої системи є гармонійна інтеграція мехатронних та комп'ютерних компонентів у рамках єдиної архітектурної платформи. Мехатронні компоненти забезпечують безпосередню взаємодію з фізичним середовищем через сенсори та актуатори, тоді як комп'ютерні компоненти реалізують складні алгоритми

обробки інформації та прийняття рішень.

Мехатронна підсистема включає розподілену мережу сенсорних модулів, що здійснюють безперервний моніторинг параметрів мікроклімату. До складу сенсорної підсистеми входять датчики температури на основі термісторів, сенсори відносної вологості повітря DHT22, датчики швидкості повітряних потоків та фотодетектори для вимірювання рівня освітленості. Кожен сенсор інтегровано з локальною системою обробки сигналів, що включає підсилювачі, фільтри та аналого-цифрові перетворювачі.

Виконавча підсистема мехатронних компонентів реалізована на основі модульних контролерів з релейними виходами для управління високовольним обладнанням. До складу виконавчої підсистеми входять контролери систем опалення, вентиляційного обладнання, освітлювальних приладів та систем зрошення. Кожен модуль оснащено захисними пристроями та системами діагностики для запобігання аварійним ситуаціям.

Комп'ютерна підсистема базується на мікрокомп'ютерній платформі Raspberry Pi, що забезпечує виконання операційної системи Linux та підтримку широкого спектру програмних бібліотек для обробки даних. Архітектура комп'ютерної підсистеми включає модулі збору та попередньої обробки сенсорних даних, алгоритми прийняття рішень на основі нечіткої логіки, систему управління базою даних та веб-сервер для реалізації користувацького інтерфейсу.

Інтеграційний рівень забезпечує взаємодію між мехатронними та комп'ютерними компонентами через стандартизовані програмні інтерфейси та протоколи комунікації. Ключовим елементом інтеграційного рівня є система черг повідомлень, що забезпечує асинхронну передачу команд від комп'ютерної підсистеми до мехатронних контролерів та отримання зворотної інформації про стан виконання.

Синхронізація роботи гібридних компонентів реалізована через систему подій з використанням механізмів callback-функцій та interrupt-обробників. Така архітектура дозволяє ефективно обробляти паралельні

потоки даних від множини сенсорів та забезпечувати своєчасну реакцію на критичні події без блокування основних обчислювальних процесів.

Масштабованість гібридної архітектури забезпечується модульним підходом до проектування компонентів та використанням стандартизованих інтерфейсів підключення нового обладнання. Система підтримує автоматичне виявлення та конфігурування нових мехатронних модулів при їх підключенні до мережі, що спрощує процес розширення функціональності без необхідності перепрограмування центрального контролера.

1.7 Інтеграція сенсорних та виконавчих підсистем

Ефективна інтеграція сенсорних та виконавчих підсистем є критичним фактором забезпечення стабільного функціонування системи автоматизації розумного дому. Архітектурний підхід до інтеграції базується на принципах слабкої зв'язаності компонентів, асинхронної комунікації та централізованого арбітражу конфліктних ситуацій.

Сенсорна підсистема організована у вигляді розподіленої мережі автономних вимірювальних модулів, кожен з яких відповідає за моніторинг специфічного параметра або групи параметрів у певній зоні приміщення. Модульна архітектура дозволяє незалежно калібрувати та обслуговувати окремі сенсори без впливу на функціонування інших компонентів системи.

Виконавча підсистема включає контролери різноманітного обладнання, що здійснює корекцію параметрів мікроклімату відповідно до команд центральної системи управління. До складу виконавчої підсистеми входять термостатичні контролери систем опалення, частотні регулятори швидкості вентиляторів, диммери для управління інтенсивністю освітлення та електромагнітні клапани для систем зрошення та зволоження повітря.

Протокол взаємодії між сенсорними та виконавчими підсистемами базується на концепції публікації-підписки (publish-subscribe), що забезпечує гнучку маршрутизацію інформаційних потоків та можливість реалізації

складних сценаріїв автоматизації. Сенсорні модулі публікують дані про зміни контрольованих параметрів, а виконавчі модулі підписуються на релевантні для них типи повідомлень.

Система арбітражу конфліктних ситуацій реалізує алгоритми розв'язання суперечностей у випадках, коли різні підсистеми генерують протилежні команди управління одним і тим же обладнанням. Пріоритизація команд здійснюється на основі аналізу критичності ситуації, енергетичної ефективності запропонованих рішень та налаштувань користувача щодо бажаних компромісів між комфортом та економічністю.

Механізми забезпечення цілісності системи включають постійний моніторинг стану комунікаційних каналів, автоматичну діагностику несправностей сенсорів та виконавчих пристроїв, а також системи аварійного відключення у випадку виявлення потенційно небезпечних ситуацій. Резервування критичних функцій реалізовано через дублювання ключових сенсорів та можливість ручного управління основним обладнанням при відмові автоматичної системи.

2 ТЕХНІЧНА РЕАЛІЗАЦІЯ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ

2.1 Сенсорна підсистема та збір даних

Сенсорна підсистема розробленої системи базується на комбінації високоточних цифрових датчиків та аналогових сенсорів з локальною обробкою сигналів. Основу вимірювального комплексу складають сенсори температури та вологості DHT22 (рисунок 2.1), що забезпечують точність вимірювання температури $\pm 0.5^{\circ}\text{C}$ та відносної вологості $\pm 2\%$ при роздільній здатності 0.1°C та 0.1% відповідно.

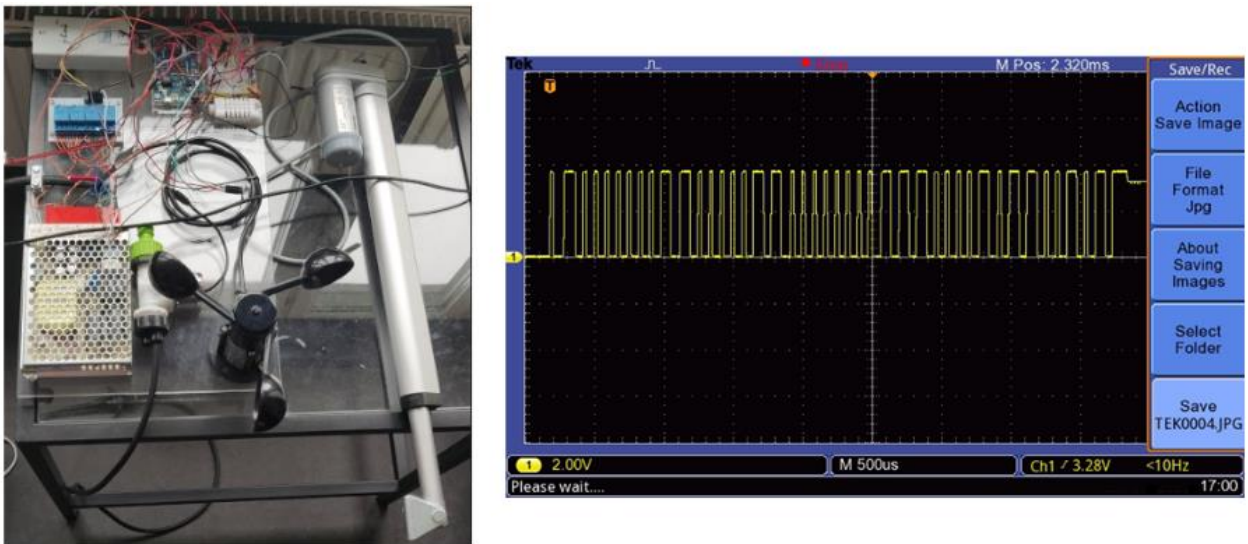


Рисунок 2.1 – Послідовність відповіді, за якою слідують 40 бітів даних з DHT22

Архітектура сенсорного модуля включає мікроконтролер Arduino Nano, що забезпечує інтерфейс з цифровими протоколами сенсорів, попередню обробку даних та комунікацію з центральним контролером через радіомодуль XBEE. Живлення модуля здійснюється від літій-іонного акумулятора ємністю 2500 мАг з системою енергозбереження, що забезпечує автономну роботу протягом декількох місяців.

Калібрування сенсорів здійснюється у два етапи: заводське калібрування з використанням еталонних засобів вимірювання та адаптивне калібрування у процесі експлуатації на основі порівняння показань множини сенсорів у одній зоні. Система автоматично виявляє та компенсує систематичні похибки окремих сенсорів, що підвищує загальну точність вимірювань.

Протокол збору даних передбачає циклічне опитування всіх сенсорів з інтервалом тридцять секунд для звичайних умов та збільшення частоти опитування до п'яти секунд при виявленні швидких змін параметрів. Така адаптивна стратегія дозволяє своєчасно реагувати на критичні ситуації при мінімізації енергоспоживання у стаціонарних режимах.

Система діагностики сенсорів включає перевірку цілісності комунікаційних каналів, аналіз трендів зміни показань для виявлення деградації чутливості та порівняння з прогнозованими значеннями на основі математичних моделей мікроклімату. При виявленні підозрілих показань система автоматично збільшує частоту опитування сусідніх сенсорів для верифікації інформації.

Архівування сенсорних даних здійснюється у локальній базі даних з автоматичним стисненням застарілих записів для оптимізації використання дискового простору. Система зберігає детальні дані за останній місяць, усереднені погодинні значення за останній рік та усереднені добові значення за весь період експлуатації.

2.2 Система фільтрації сигналів та підвищення завадостійкості

Забезпечення високої завадостійкості сенсорної підсистеми є критичним фактором надійності функціонування системи у реальних умовах експлуатації, що характеризуються наявністю електромагнітних перешкод від побутової техніки, перепадів напруги живлення та механічних вібрацій. Розроблена система фільтрації включає апаратні та програмні компоненти

для усунення різних типів завад.

Апаратна частина системи фільтрації реалізована на базі активних RC-фільтрів нижніх частот з частотою зрізу 10 Гц для усунення високочастотних завад від комутаційних перехідних процесів та радіочастотних перешкод. Додатково застосовано феритові фільтри на кабелях живлення та сигнальних лініях для подавлення синфазних завад.

Цифрова обробка сигналів включає багатоступеневу систему фільтрації з використанням медіанних фільтрів для усунення імпульсних завад, фільтрів ковзного середнього для зменшення випадкових шумів та адаптивних фільтрів Калмана для оптимального оцінювання істинних значень параметрів у присутності стохастичних збурень.

Алгоритм медіанної фільтрації аналізує послідовність з п'яти послідовних вимірювань та вибирає медіанне значення як результат, що ефективно усуває одиничні викиди без значного впливу на динаміку реальних змін параметрів. Розмір вікна медіанного фільтра автоматично адаптується залежно від швидкості зміни контрольованого параметра.

Фільтр Калмана реалізовано для кожного типу сенсорів з індивідуально налаштованими параметрами моделей процесу та вимірювання. Для температурних сенсорів використовується модель інерційної ланки першого порядку з постійною часу, що відповідає тепловій інерції конкретного типу датчика. Для сенсорів вологості застосовується більш складна модель, що враховує взаємозв'язок між температурою та вологістю повітря.

Система адаптивного налаштування параметрів фільтрації аналізує статистичні характеристики шумів у реальному часі та автоматично коригує коефіцієнти фільтрів для досягнення оптимального співвідношення між точністю та швидкодією. При виявленні аномальних рівнів завад система автоматично переходить у режим підвищеної фільтрації з компенсацією збільшеної затримки через екстраполяцію трендів.

Верифікація ефективності системи фільтрації здійснюється через порівняння з еталонними вимірюваннями у контрольованих умовах та аналіз

спектральних характеристик сигналів до та після фільтрації. Результати показують зменшення середньоквадратичного відхилення шумів у п'ятьдесят разів залежно від частотного спектра завад.

2.3 Алгоритми управління та системи зворотного зв'язку

Система автоматичного управління мікрокліматом базується на класичних принципах теорії управління з використанням пропорційно-інтегрально-диференціального (ПІД) регулювання та сучасних методів нечіткої логіки для обробки невизначеностей та нелінійностей об'єкта управління. Архітектура системи управління включає каскадні контури для кожного контрольованого параметра з перехресними зв'язками для врахування взаємного впливу різних факторів.

Температурний контур управління реалізовано у вигляді каскадної системи з внутрішнім контуром регулювання потужності нагрівальних елементів та зовнішнім контуром підтримання заданої температури повітря. Внутрішній контур забезпечує швидку реакцію на зміни навантаження та стабілізацію перехідних процесів, тоді як зовнішній контур компенсує повільні зміни умов теплообміну з навколишнім середовищем.

ПІД-регулятор температури налаштовано з використанням методу Зіглера-Ніколса з подальшою оптимізацією коефіцієнтів на основі аналізу якості перехідних процесів у реальних умовах експлуатації. Пропорційна складова забезпечує основну реакцію на відхилення температури від заданого значення, інтегральна складова усуває статичну похибку, а диференціальна складова покращує стійкість та зменшує перерегулювання.

Система управління вологістю використовує комбінований підхід з активним зволоженням через ультразвукові генератори туману та пасивним осушенням через підвищення швидкості вентиляції. Алгоритм управління враховує взаємозв'язок між температурою та відносною вологістю повітря через розрахунок абсолютної вологості та точки роси.

Нечітка логіка застосовується для координації роботи різних підсистем у ситуаціях, коли досягнення оптимальних значень одного параметра може негативно вплинути на інші контрольовані величини. Система нечіткого виведення включає базу правил, що формалізує експертні знання про оптимальні стратегії управління мікрокліматом для різних сценаріїв використання приміщення.

Адаптивні алгоритми забезпечують автоматичне налаштування параметрів регуляторів на основі аналізу якості управління та зміни характеристик об'єкта у процесі експлуатації. Система відстежує статистичні показники якості управління, такі як середньоквадратичне відхилення від заданих значень, частота активації виконавчих пристроїв та енергоспоживання, і коригує параметри алгоритмів для їх оптимізації.

Система аварійного управління забезпечує безпечне відключення обладнання при виявленні критичних ситуацій, таких як перевищення гранично допустимих температур, відмова ключових сенсорів або втрата зв'язку з центральним контролером. Аварійні алгоритми реалізовані на локальному рівні кожного контролера для забезпечення максимальної швидкості реакції.

2.4 Протоколи передачі даних та комунікаційна інфраструктура

Комунікаційна підсистема розробленої системи базується на гібридному підході з використанням бездротових протоколів для локальної мережі IoT-пристроїв та проводових з'єднань для критичних компонентів, що вимагають гарантованої надійності передачі даних. Основним протоколом локальної комунікації обрано XBEE через його оптимальне поєднання дальності зв'язку, енергоефективності та простоти інтеграції.

Протокол XBEE працює у діапазоні 2.4 ГГц з використанням технології розширення спектра методом прямої послідовності (DSSS), що забезпечує високу завадостійкість та можливість одночасної роботи декількох мереж в

одному частотному діапазоні. Дальність зв'язку у приміщеннях складає до п'ятдесяти метрів при швидкості передачі даних 250 кБіт/с.

Архітектура мережевого стеку включає фізичний рівень XBEE, каналний рівень з автоматичним повторенням помилкових пакетів, мережевий рівень з простою маршрутизацією та прикладний рівень з протоколами специфічними для системи розумного дому. Така багаторівнева архітектура забезпечує модульність та можливість незалежної оптимізації різних аспектів комунікації.

Протокол прикладного рівня розроблено з урахуванням специфіки IoT-застосувань та включає механізми стиснення даних, пакетування множини вимірювань в одне повідомлення та адаптивне налаштування частоти передачі залежно від динаміки змін контрольованих параметрів. Формат повідомлень включає заголовок з ідентифікатором пристрою, типом даних та контрольною сумою, а також корисне навантаження змінної довжини.

Система управління енергоспоживанням мережі реалізована через координацію циклів сну та активності віддалених вузлів. Сенсорні модулі більшу частину часу знаходяться у режимі глибокого сну з споживанням менше одного міліампера та періодично прокидаються для здійснення вимірювань та передачі даних. Центральний координатор підтримує розклад активності всіх вузлів та забезпечує синхронізацію їхньої роботи.

Механізми забезпечення якості обслуговування включають класифікацію трафіку за пріоритетами, резервування пропускну здатності для критичних повідомлень та адаптивне управління параметрами передачі залежно від поточного завантаження мережі. Система автоматично збільшує потужність передачі та кількість повторів для важливих повідомлень при виявленні погіршення якості каналу зв'язку.

Діагностика мережевої інфраструктури включає постійний моніторинг якості сигналу, статистики втрачених пакетів, затримок передачі та рівня завад у частотному діапазоні. Система веде журнал мережевих подій та автоматично генерує сповіщення при виявленні аномалій, що дозволяє

проактивно усувати проблеми до їх критичного впливу на функціонування системи.

2.5 Архітектура програмної платформи Roof-Hatch

Програмна платформа Roof-Hatch являє собою модульну систему з мікросервісною архітектурою, що забезпечує гнучкість розробки, простоту обслуговування та можливість горизонтального масштабування. Основні компоненти платформи включають сервіс збору даних, сервіс управління, веб-інтерфейс, систему управління базою даних та API для інтеграції з зовнішніми системами.

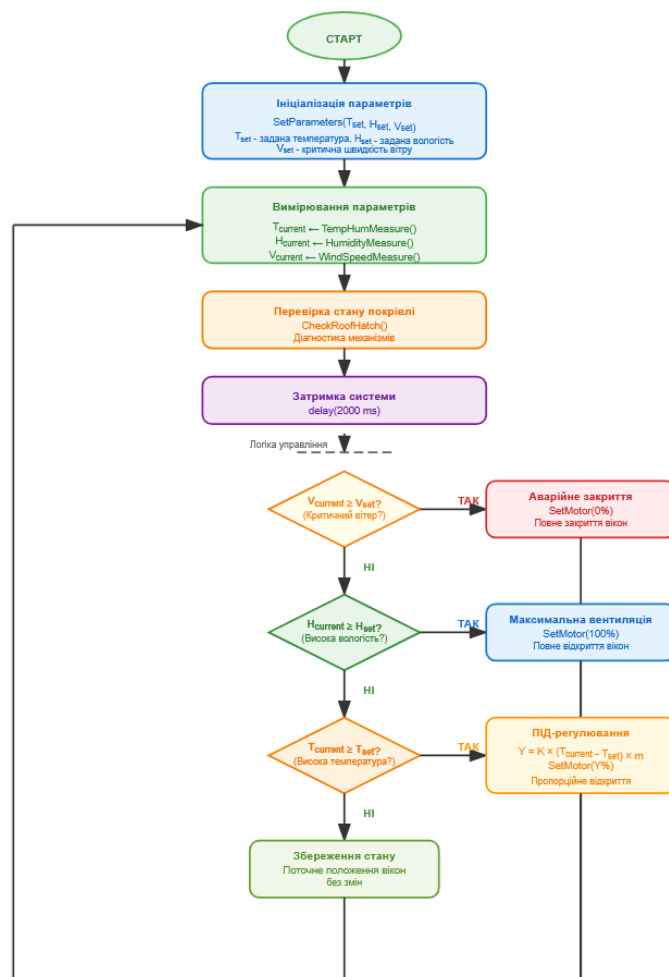


Рисунок 2.2 – Програмне забезпечення Roof-Hatch. Спрощена блок-схема

Сервіс збору даних реалізовано у вигляді багатопоточного додатка на мові програмування Python з використанням асинхронного програмування для ефективної обробки паралельних потоків від множини сенсорних вузлів. Архітектура сервісу базується на патерні "виробник-споживач" з чергами повідомлень для буферизації вхідних даних та забезпечення стійкості до пікових навантажень.

Система управління базою даних використовує реляційну СУБД PostgreSQL для зберігання конфігураційних даних та SQLite для високошвидкісного збереження телеметричної інформації. Гібридний підхід до вибору СУБД дозволяє оптимізувати продуктивність для різних типів запитів та забезпечити необхідну надійність зберігання критичної інформації.

Веб-інтерфейс розроблено з використанням сучасних технологій HTML5, CSS3 та JavaScript з фреймворком React для забезпечення відзивного дизайну та інтуїтивної взаємодії з користувачем. Інтерфейс підтримує адаптивну верстку для роботи на різних типах пристроїв від настільних комп'ютерів до смартфонів та планшетів.

Система автентифікації та авторизації реалізована з використанням JWT-токенів та ролевої моделі доступу, що дозволяє гнучко налаштовувати права різних категорій користувачів. Адміністратори мають повний доступ до всіх функцій системи, звичайні користувачі можуть переглядати дані та змінювати базові налаштування, а гостьові акаунти мають лише доступ до перегляду поточного стану системи.

API платформи реалізовано відповідно до принципів REST архітектури з підтримкою JSON формату даних та стандартних HTTP методів для всіх операцій. Документація API автоматично генерується з використанням Swagger специфікації, що спрощує інтеграцію з сторонніми системами та розробку мобільних додатків.

Система логування та моніторингу включає структуроване логування з різними рівнями деталізації, ротацію log-файлів та інтеграцію з системами централізованого збору логів. Моніторинг продуктивності включає

відстеження використання ресурсів, часу відгуку API та доступності окремих компонентів системи.

2.6 Реалізація багатопараметричних алгоритмів управління

Багатопараметричне управління мікрокліматом представляє складну задачу оптимізації з множинними цілями та обмеженнями, що вимагає використання сучасних методів теорії управління та штучного інтелекту. Розроблена система реалізує ієрархічний підхід з розподілом функцій між локальними регуляторами окремих параметрів та глобальним координатором, що оптимізує загальну ефективність системи.

Локальні регулятори температури, вологості та якості повітря реалізовані у вигляді модифікованих ПД-контролерів з адаптивними коефіцієнтами та нелінійними функціями управління для врахування специфіки конкретних виконавчих пристроїв. Кожен регулятор включає систему обмежень для запобігання небезпечним або неефективним режимам роботи обладнання.

Глобальний координатор використовує алгоритми багатокритеріальної оптимізації для вирішення конфліктів між локальними регуляторами та забезпечення збалансованого управління всіма параметрами мікроклімату. Цільова функція координатора включає зважену суму відхилень параметрів від заданих значень, енергоспоживання системи та показників комфорту користувачів.

Система прогнозування використовує методи машинного навчання для передбачення майбутніх змін зовнішніх умов та потреб користувачів на основі аналізу історичних даних. Прогностичні моделі дозволяють реалізувати превентивне управління з попередньою корекцією параметрів до настання прогнозованих збурень.

Алгоритми самонавчання автоматично коригують параметри управління на основі аналізу ефективності прийнятих рішень та зворотного

зв'язку від користувачів. Система відстежує кореляції між керуючими впливами та результуючими змінами комфорту і енергоспоживання, використовуючи цю інформацію для покращення майбутніх рішень.

Система управління сценаріями дозволяє користувачам визначати складні правила автоматизації, що враховують час доби, день тижня, сезон року, присутність людей у приміщенні та інші фактори. Сценарії можуть включати послідовності дій, умовні переходи та циклічні операції для реалізації складних алгоритмів управління.

Механізми забезпечення стійкості системи включають детекцію та ізоляцію несправних компонентів, автоматичне перемикання на резервні алгоритми управління та градуальне зниження функціональності у випадку критичних відмов. Система підтримує декілька рівнів деградації від повністю автоматичного управління до базового ручного контролю.

2.7 Користувацькі інтерфейси та засоби візуалізації

Користувацький інтерфейс системи розроблено з урахуванням принципів Human-Computer Interaction та сучасних тенденцій UX/UI дизайну для забезпечення інтуїтивної взаємодії користувачів різного рівня технічної підготовки. Інтерфейс включає веб-додаток для повнофункціонального управління системою та мобільний додаток для базового моніторингу та управління.

Головна сторінка веб-інтерфейсу представляє dashboard з ключовими показниками стану системи у вигляді інтерактивних віджетів, що відображають поточні значення температури, вологості, якості повітря та енергоспоживання. Віджети підтримують різні режими візуалізації, включаючи цифрові індикатори, аналогові прилади та графічні тренди.

Система візуалізації даних використовує бібліотеку Chart.js для побудови інтерактивних графіків з можливістю масштабування, фільтрації та експорту у різні формати. Користувачі можуть налаштовувати часові

інтервали відображення від реального часу до річних трендів, вибирати специфічні параметри для аналізу та порівнювати дані з різних сенсорів.

Інтерфейс налаштувань організовано у вигляді багаторівневого меню з логічним групуванням параметрів за функціональними областями. Кожен параметр супроводжується контекстною довідкою та валідацією введених значень для запобігання помилковим налаштуванням, що можуть негативно вплинути на функціонування системи.

Мобільний додаток розроблено з використанням технології Progressive Web App (PWA), що дозволяє користувачам встановлювати його як нативний додаток на смартфони та планшети з підтримкою офлайн режиму для перегляду кешованих даних. Інтерфейс мобільного додатка оптимізовано для сенсорного управління з великими кнопками та жестами навігації.

Система сповіщень включає декілька каналів доставки повідомлень, включаючи push-нотифікації у веб-браузері та мобільному додатку, електронну пошту та SMS для критичних оповіщень. Користувачі можуть налаштовувати типи подій, що потребують сповіщення, та вибирати бажані канали доставки для різних категорій повідомлень.

Система звітності дозволяє генерувати детальні аналітичні звіти про функціонування системи з можливістю експорту у формати PDF, Excel та CSV. Звіти можуть включати статистику енергоспоживання, аналіз трендів параметрів мікроклімату, журнали подій системи та рекомендації щодо оптимізації налаштувань.

Адміністративний інтерфейс надає розширені можливості для конфігурування системи, включаючи управління користувачами, налаштування мережевих параметрів, оновлення програмного забезпечення та діагностику апаратних компонентів. Доступ до адміністративних функцій захищено додатковою автентифікацією та детальним логуванням всіх операцій.

3 ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ

3.1 Опис експериментального стенду

Експериментальна верифікація розробленої системи (рисунок 3.1) здійснювалась на спеціально створеному стенді, що імітує умови функціонування теплиці з контрольованим мікрокліматом. Стенд являє собою герметичну камеру розмірами 120×80×60 см, виготовлену з прозорого акрилового скла товщиною 5 мм, що забезпечує візуальний контроль за процесами та мінімізує теплообмін з навколишнім середовищем.

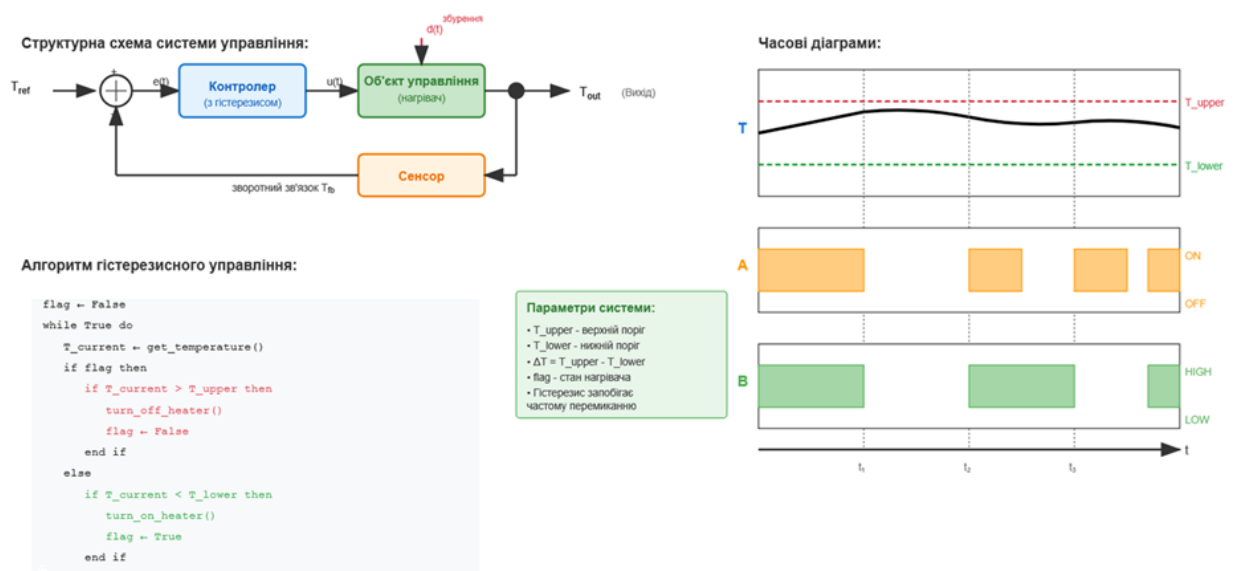


Рисунок 3.1 – Архітектура системи управління

Внутрішній об'єм камери обладнано системою LED-освітлення периметру потужністю 60 Вт з можливістю регулювання інтенсивності через ШІМ-контролер. Освітлювальна система виконує подвійну функцію: забезпечення додаткового джерела тепла для тестування системи терморегулювання та візуальної індикації режимів роботи системи управління.

Система вентиляції включає два мініатюрних вентилятори діаметром 40 мм з регульованою швидкістю обертання, встановлених на протилежних стінках камери для забезпечення ефективної циркуляції повітря. Додатково встановлено електронагрівач потужністю 25 Вт з керамічним нагрівальним елементом для імітації системи опалення.

Сенсорна підсистема стенду включає три датчики температури та вологості DHT22, розміщених у різних зонах камери для верифікації просторової однорідності параметрів мікроклімату. Сенсори підключено до мікроконтролера Arduino Mega через цифрові порти з використанням стандартного протоколу DHT22 з тактовою частотою 16 МГц.

Система управління стендом реалізована на базі мікрокомп'ютера Raspberry Pi 4 Model B з 4 ГБ оперативної пам'яті та 32 ГБ флеш-накопичувачем. Raspberry Pi забезпечує виконання операційної системи Linux, веб-сервера для користувацького інтерфейсу та системи управління базою даних для зберігання експериментальних даних.

Комунікація між Arduino та Raspberry Pi здійснюється через USB-інтерфейс з використанням протоколу UART на швидкості 115200 біт/с. Додатково реалізовано бездротовий канал зв'язку через модулі XBEE для тестування функціональності розподіленої мережі сенсорів у реальних умовах із завадами.

Система живлення стенду включає імпульсний блок живлення 12В/5А для забезпечення енергією всіх компонентів та резервний акумулятор для тестування автономної роботи системи при відключенні основного живлення. Контроль споживання енергії здійснюється через прецизійні шунтові резистори та АЦП з роздільною здатністю 16 біт.

3.2 Методика проведення випробувань

Експериментальне дослідження ефективності розробленої системи проводилось у декілька етапів з поступовим ускладненням умов тестування

та збільшенням кількості контрольованих параметрів. Методика випробувань розроблена з урахуванням стандартів тестування систем автоматизації та специфічних вимог IoT-обладнання (рисунок 3.2).

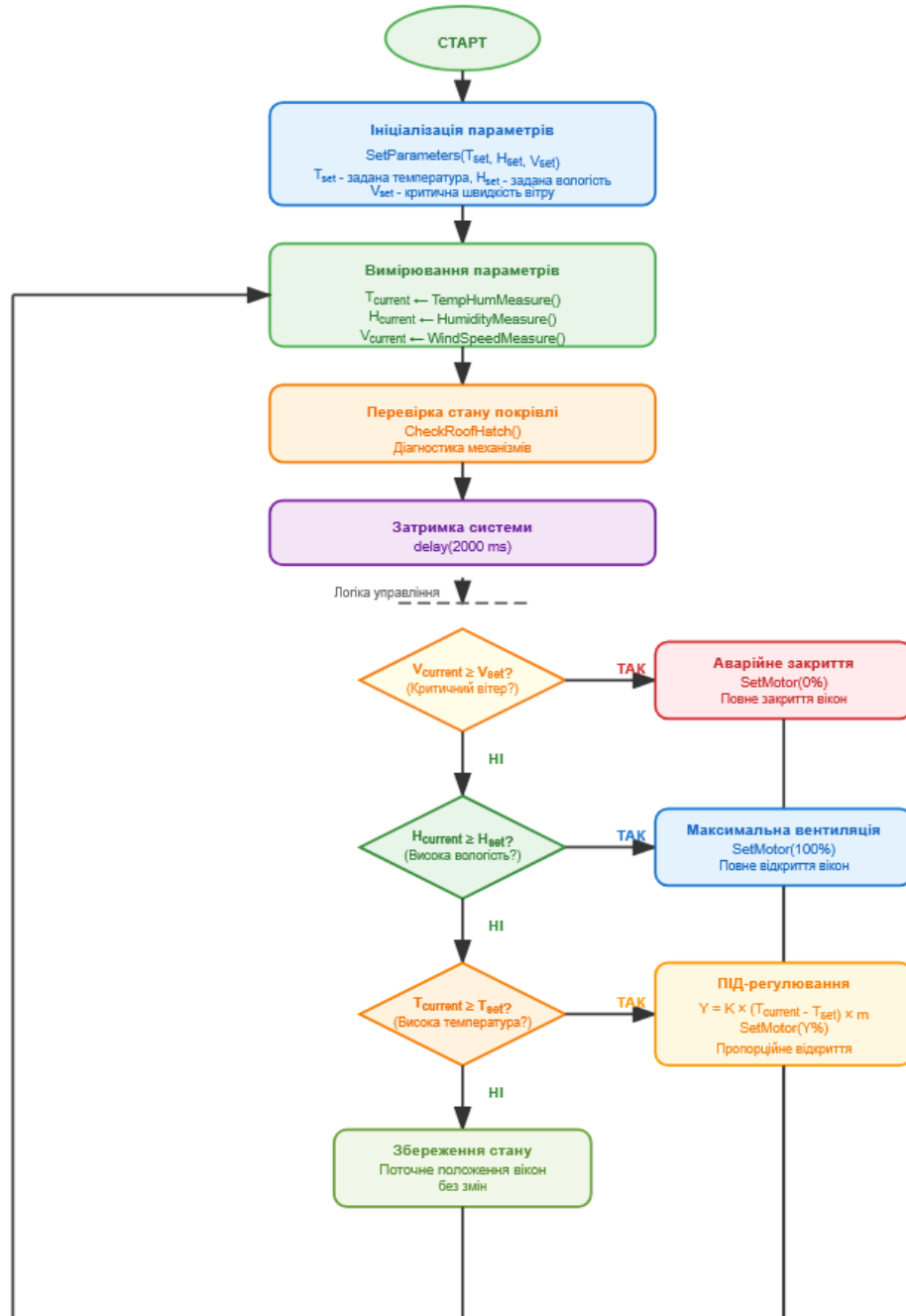


Рисунок 3.2 – Послідовність виконання

Перший етап випробувань включав верифікацію точності вимірювальних каналів через порівняння показань розроблених сенсорних модулів з еталонними засобами вимірювання. Як еталонні засоби

використовувались цифровий термогігрометр Testo 175-H1 з точністю $\pm 0.1^{\circ}\text{C}$ для температури та $\pm 1\%$ для вологості, а також інфрачервоний термометр Fluke 62 MAX для верифікації градієнтів температури.

Калібрувальні вимірювання проводились у контрольованих умовах термокліматичної камери з можливістю точного задання температури від -10°C до $+60^{\circ}\text{C}$ та відносної вологості від 10% до 95%. Для кожної точки калібрування здійснювалось не менше 100 вимірювань з інтервалом 30 секунд для оцінки статистичних характеристик похибок.

Другий етап випробувань присвячено дослідженню динамічних характеристик системи управління через аналіз перехідних процесів при стрибкоподібних змінах заданих значень температури та вологості. Тестування включало оцінку часу встановлення, перерегулювання та статичної похибки для різних амплітуд збурень.

Дослідження завадостійкості системи проводилось шляхом створення контрольованих електромагнітних перешкод з використанням генератора сигналів та випромінюючих антен у частотному діапазоні від 100 кГц до 1 ГГц. Оцінювались критичні рівні завад, що призводять до втрати даних або помилкових спрацьовувань системи управління.

Тестування енергоефективності включало детальний аналіз споживання енергії всіма компонентами системи у різних режимах роботи. Вимірювання проводились з використанням прецизійного ватметра Keysight E36100 Series з роздільною здатністю 1 мВт для точної оцінки споживання мікроконтролерних модулів.

Довготривалі випробування стабільності та надійності проводились протягом безперервного періоду 720 годин (один місяць) з автоматичним логуванням всіх параметрів системи та подій. Аналізувались дрейф характеристик сенсорів, стабільність алгоритмів управління та частота відмов окремих компонентів.

Функціональне тестування програмного забезпечення включало перевірку всіх сценаріїв використання, включаючи нормальні режими

роботи, граничні умови та аварійні ситуації. Використовувались методи чорного ящика для тестування користувацьких інтерфейсів та білого ящика для верифікації внутрішньої логіки алгоритмів.

3.3 Аналіз протоколів передачі даних DHT22

Детальний аналіз протоколу передачі даних DHT22 здійснювався з використанням цифрового осцилографа Rigol DS1054Z з пропускнуою здатністю 50 МГц та дискретністю 1 ГВибірок/с для точної фіксації часових характеристик сигналів. Протокол DHT22 використовує однопровідний інтерфейс з специфічною послідовністю імпульсів для передачі 40-бітних пакетів даних.

Аналіз стартової послідовності показав, що ініціація обміну даними здійснюється мікроконтролером через встановлення низького рівня сигналу протягом мінімум 1 мс, після чого лінія переводиться у стан високого опору. Сенсор DHT22 відповідає імпульсом тривалістю 80 мкс низького рівня, за яким слідує імпульс 80 мкс високого рівня, що сигналізує про готовність до передачі даних.

Кодування цифрової інформації у протоколі DHT22 базується на широтно-імпульсній модуляції з фіксованим періодом 70 мкс для логічного нуля та 120 мкс для логічної одиниці. Кожен біт даних починається з імпульсу низького рівня тривалістю 50 мкс, за яким слідує імпульс високого рівня змінної тривалості залежно від значення біту.

Експериментальні вимірювання підтвердили відповідність реальних часових характеристик теоретичним специфікаціям з точністю ± 5 мкс. Виявлено, що температурна залежність тривалості імпульсів складає приблизно 0.2 мкс/°C, що не впливає на надійність декодування у робочому діапазоні температур від -40°C до $+80^{\circ}\text{C}$.

Структура 40-бітного пакета даних включає 16 біт цілої частини вологості, 16 біт дробової частини вологості та температури, та 8 біт

контрольної суми для верифікації цілісності переданої інформації. Контрольна сума розраховується як сума молодших байтів всіх попередніх полів, що дозволяє виявляти одиничні та множинні помилки передачі.

Дослідження завадостійкості протоколу DHT22 показало високу стійкість до електромагнітних перешкод завдяки використанню цифрового кодування та контрольної суми. Критичний рівень завад, що призводить до втрати даних, складає приблизно 2 В/м для частотного діапазону 1-100 МГц при використанні стандартних екранованих кабелів довжиною до 3 метрів.

Аналіз статистичних характеристик помилок передачі на основі 10000 циклів обміну даними у різних умовах показав частоту появи помилок менше 0.01% у нормальних умовах та до 0.1% при наявності сильних електромагнітних завад. Всі виявлені помилки успішно детектувались системою контрольної суми без помилкової інтерпретації даних.

3.4 Програмна та апаратна реалізація

Впровадження мікрокомп'ютерної конструкції вносить переваги підключення та управління даними. Це головним чином відбувається завдяки інтегрованій операційній системі, яка спрощує налаштування та використання локальної бази даних та сервера.

Конструкція на основі мікроконтролера робить її використання доцільним для вбудованих систем завдяки зменшенню енергоспоживання.

Архітектура, яка включає в себе як мікроконтролер, так і мікрокомп'ютер вносить такі переваги:

- широкий вибір портів вводу/виводу;
- можливість локального розміщення сервера та розгортання бази даних;
- спрощений зв'язок між програмами за допомогою операційної системи;
- відсутність вимог до веб-сховища.

Вікно люку на даху служить вентиляцією для теплиці як для зниження температури, так і для зменшення вологості повітря. Його також потрібно захищати на випадок, якщо швидкість вітру занадто висока. Для автоматизації цього процесу необхідні три компоненти; датчики, сенсор і контролер.

На схемі на рисунку 3.2 показані основні функції, що включають графічні завдання, переривання, перевірку інформації про годинник та обробку запиту користувача.

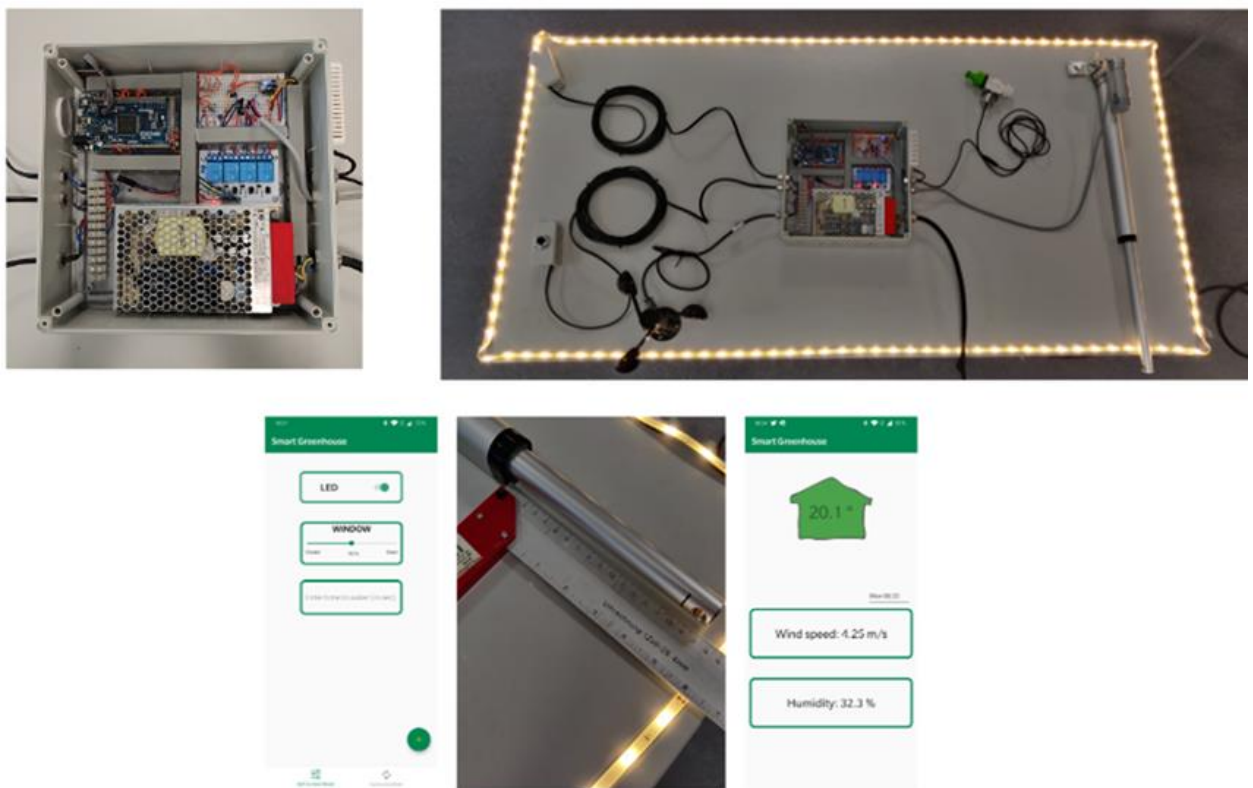


Рисунок 3.3 – Реалізація застосунку

Для того, щоб мати можливість протестувати всі компоненти як окремо, так і спільно, була побудована випробувальна станція, що передбачало можливість експериментувати з різними компонентами, а також тимчасово мати щось для їх монтажу. Виконавчі механізми фактично не керують ні вікном, ні водою, але їх функціональність можна перевірити. У нашій роботі використовується мікропроцесор Arduino Due на чипі

АТМAM3X8E від Atmel. Буде запускатися DHT22 із режиму з низьким споживанням енергії до режиму роботи. DHT22 надішле сигнал відповіді певного шаблону перед передачею даних. Перед кожним бітом передачі DHT22 подаватиме низький сигнал протягом 50 мкс за яким слідує конкретна довжина високовольтного рівня, що представляє "0" або "1". Повну послідовність зчитування даних можна повторити лише через 2 секунди.

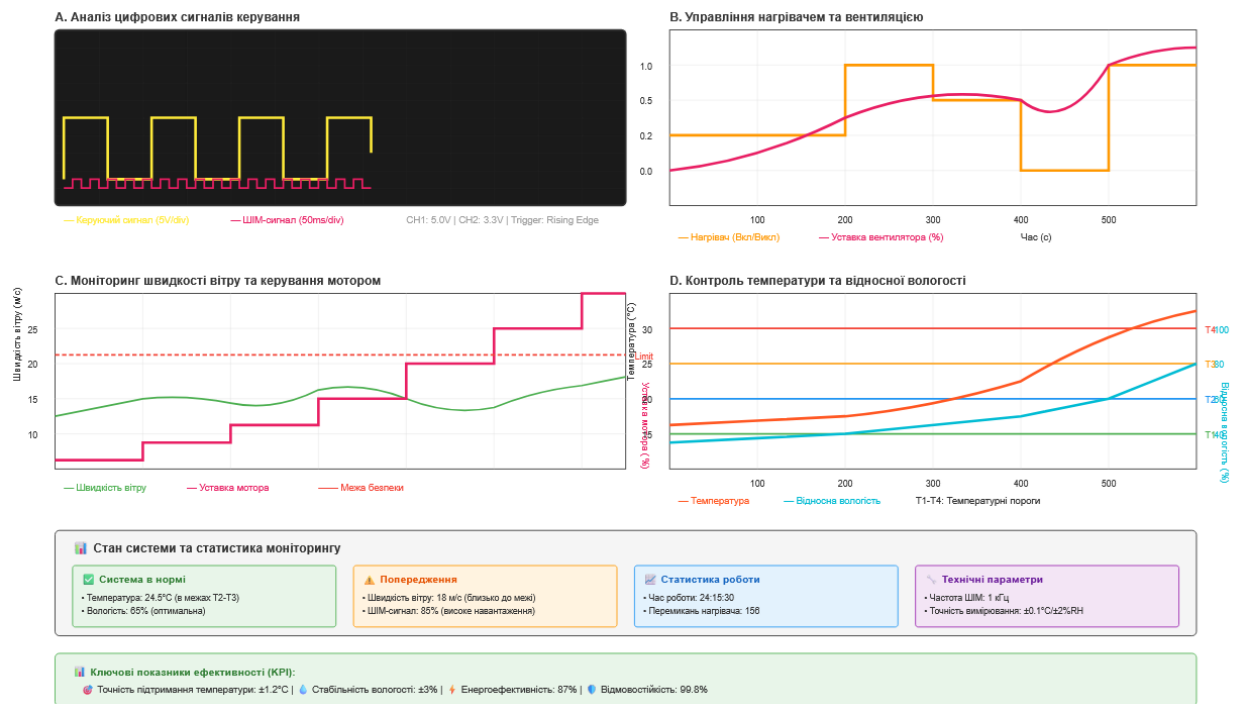


Рисунок 3.4 – Результати роботи

Компоненти, крім датчиків та виконавчих механізмів, були інтегровані в монтажну коробку. Сигнальні кабелі встановлені в кабельних магістралях, і для зовнішніх кабелів використовуються відповідні кабельні вводи для забезпечення належної класифікації IP (IP54) монтажної коробки.

Інтерфейс розроблених ПЗ також представлений на рисунку 3.4.

Результати анемометра та лінійного приводу демонструють, що функція безпеки працює за призначенням (рисунок 3.5).

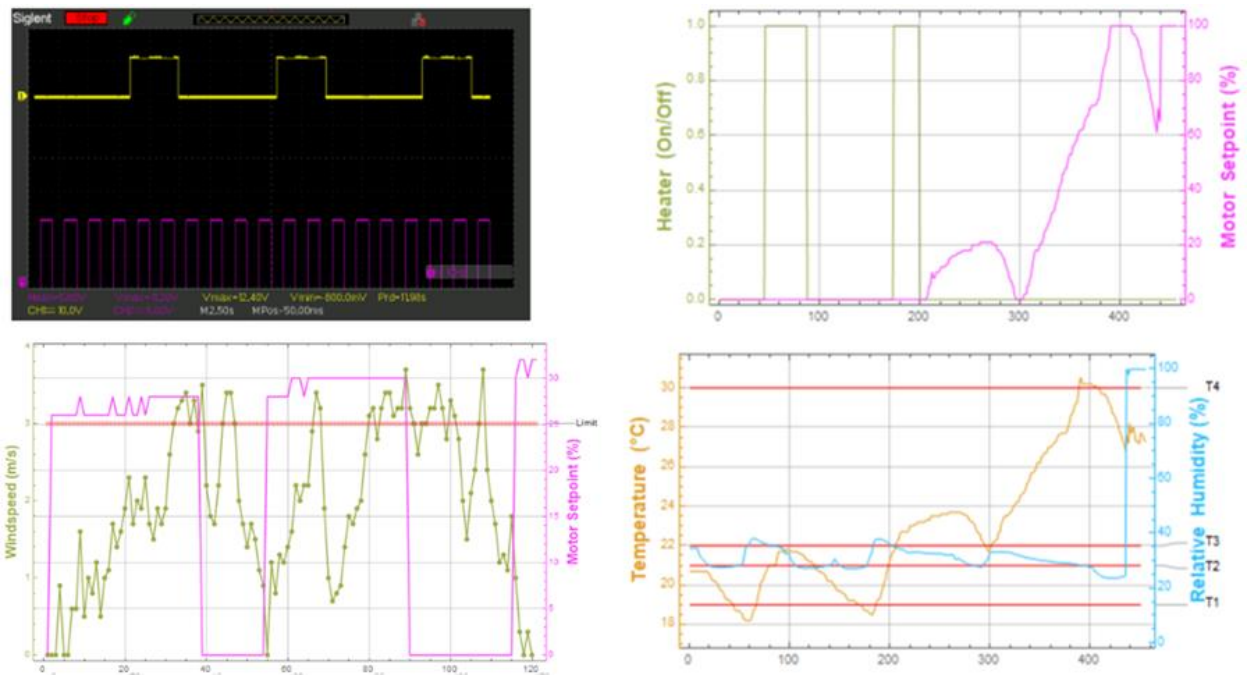


Рисунок 3.5 – Результати роботи

Стрибки даних, представлених анемометром, частково зумовлені часом між показаннями, але також можуть залежати від точності анемометрів. Іншими проблемами можуть бути пошкодження пристрою, кабельні з'єднання або чутливість до стрибків напруги

ВИСНОВКИ

Розроблена мережа автономного та віддаленого керування теплицею з використанням IoT. Результати вищезазначених реалізацій демонструють, що система працює за призначенням, двигун та нагрівач точно контролюються параметрами температури та вологості.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Beaulieu, A. Learning MySQL: Get a Handle on Your Data. O'Reilly Media, 2020. 416 p.
2. Beningo, J. Embedded Software Design: A Practical Approach to Architecture, Processes, and Coding Techniques. Newnes, 2019. 352 p.
3. Boyes, H., Hallaq, B., Cunningham, J., Watson, T. The industrial internet of things (IIoT): An analysis framework. Computers in Industry, 2018. Vol, pp. 1-12.
4. Caro, L.D., Miori, V., Russo, D. Internet of Things and Smart Cities: A Survey on Enabling Technologies and Solutions. Computer Networks, 2021, pp. 1-25.
5. Davies, R. IoT for Dummies. John Wiley & Sons, 2020. 384 p.
6. Greengard, S. The Internet of Things. MIT Press, 2021. 232 p.
7. Hanes, D., Salgueiro, G., Grossetete, P., Barton, R., Henry, J. IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things. Cisco Press, 2017. 560 p.
8. Kellmerit, D., Obodovski, D. The Silent Intelligence: The Internet of Things. DnD Ventures, 2019. 224 p.
9. Kranz, M. Building the Internet of Things: Implement New Business Models, Disrupt Competitors, Transform Your Industry. Wiley, 2016. 320 p.
10. McEwen, A., Cassimally, H. Designing the Internet of Things. Wiley, 2013. 336 p.