

КВАЛІФІКАЦІЙНА РОБОТА

бакалавр
Освітній рівень


Мікропроцесорна система контролю мікроклімату розумного саду
Назва теми

КвРКІ 210480.21.04.32 ПЗ
Шифр

Галузь знань 12 «Інформаційні технології»
Шифр, назва

Спеціальність 123 «Комп'ютерна інженерія»
Шифр, назва

Освітня програма «Комп'ютерна інженерія та програмування»
Назва

Виконав: студент IV курсу, група K12-21-4  Назар БАБІЧЕНКО
Підпис Ініціали, прізвище

Керівник  Світлана САЧЕНКО
Підпис, дата Ініціали, прізвище

Нормоконтролер  Тетяна КИСІЛЬ
Підпис, дата Ініціали, прізвище

До захисту допускаю:
Зав. кафедри комп'ютерної
інженерії та інформаційних
систем

 Ольга ПАВЛОВА
Підпис Ініціали, прізвище

« 9 » травня 2025 р.

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет Інформаційних технологій

Кафедра Комп'ютерної інженерії та інформаційних систем

Освітній рівень: бакалавр

Галузь знань: 12 Інформаційні технології

Спеціальність: 123 Комп'ютерна інженерія

Освітня програма: «Комп'ютерна інженерія та програмування»

ЗАТВЕРДЖУЮ

Зав. кафедри Ольга Павлова



" 10 " 01 2025 р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА

Бабієнко Назару Віталійовичу

Прізвище, ім'я та батькові студента

1. Тема проекту (роботи) Мікропроцесорна система контролю мікроклімату розумного саду

Керівник проекту (роботи) САМЕНКО Світлана, к.с.н., доцент

Прізвище, ім'я та по батькові науковця, місце роботи

Затверджена наказом ректора університету від 07.02.2025 р. № 23

2. Строк подання студентом проекту (роботи) на кафедру 01.06.2025 р.

3. Вихідні дані до проекту (роботи) Завдання на кваліфікаційну роботу

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

Мікропроцесорна система контролю мікроклімату розумного саду та постановка задачі щодо її розробки

Проектування апаратної частини мікропроцесорної системи контролю мікроклімату розумного саду

Програмна реалізація та тестування мікропроцесорної системи контролю мікроклімату розумного саду

5. Перелік графічного матеріалу (із зазначенням обов'язкових креслень)

Архітектура програмного забезпечення системи контролю мікроклімату

Структурна схема мікропроцесорної системи на базі ESP32

Схема з'єднання компонентів на макетній платі

Блок-схема роботи основного алгоритму системи

Інтерфейс користувача в мобільному застосунку Flutter

Загальний вигляд пристрою в захисному корпусі

6. Консультанти розділів дипломного проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Тетяна КИСІЛЬ, доцент кафедри КПС		
Антиплагіат	Андрій НІЧЕПОРУК, доцент кафедри КПС		

7. Дата видачі завдання « 10 » 01 2025 р.

КАЛЕНДАРНИЙ ПЛАН

№з/п	Назва етапів (розділів) дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Примітка
1	Вибір напрямку дослідження та узгодження тематики кваліфікаційної роботи з керівником	10.01.2025	виконано
2	Ознайомлення з предметною областю; формулювання мети та задач дослідження; визначення об'єкта та предмета дослідження	01.02.2025	виконано
3	Робота над розділом 1 – аналіз існуючих систем моніторингу мікроклімату, постановка задачі та обґрунтування актуальності	01.03.2025	виконано
4	Робота над розділом 2 – вибір апаратних та програмних засобів, розробка архітектури системи контролю мікроклімату	01.04.2025	виконано
5	Робота над розділом 3 – програмно-апаратна реалізація системи на базі ESP32, тестування, моделювання	29.04.2025	виконано
6	Оформлення пояснювальної записки згідно вимог	25.05.2025	виконано
7	Попередній захист ВКР	26.05.2025	виконано
8	Захист ВКР на засіданні ЕК	Червень 2025 року	

Студент

Керівник роботи

Підпис

Назар БАБІЧКО
Ініціали, прізвище
Світлана САЧЕНКО

№ р я д к а	ф о р м а т	Позначення	Найменування	К і л - л и с т і в	№ ек з	П р и м і т к а
			<u>Текстові документи</u>			
1		КвРКІ 210480.21.04.32 ПЗ	Пояснювальна записка	59		
			<u>Графічні матеріали</u>			
2		КвРКІ 210480.21.04.32 Е8	Архітектура ПЗ проєкту	1		
3		КвРКІ 210480.21.04.32 Е8	Схема з'єднання компонентів на макетній платі	1		
4		КвРКІ 210480.21.04.32 Е8	Блок-схема роботи основного алгоритму системи	1		

КвРКІ 210480.21.04.32 ВП				
Зм	Арж	№ докум	Підпис	Дата
Розробив		Бабієнко	<i>Handwritten signature</i>	
Перевір.		Саченко	<i>Handwritten signature</i>	
Н.контр.		Кисіль	<i>Handwritten signature</i>	10.06.14
Зп.		Павлова	<i>Handwritten signature</i>	06.06.15
Відомість проєкту			Літера	Аркуш
			У	1
			ХНУ, КІ2-21-4	

АНОТАЦІЯ

Тема кваліфікаційної роботи: «Мікропроцесорна система контролю мікроклімату розумного саду».

Автор роботи: Назар БАБІЄНКО.

Керівник роботи: Світлана САЧЕНКО

Пояснювальна записка: 59 с., 12 рис., 10 табл., 4 дод., 50 джерел.

Графічна частина: 3 креслення.

ESP32, ARDUINO IDE, РОЗУМНИЙ САД, МІКРОКЛІМАТ, СЕНСОРИ, АВТОМАТИЗАЦІЯ, ІoT.

Метою дипломної роботи є розробка ефективної мікропроцесорної системи контролю параметрів мікроклімату в умовах «розумного саду» з використанням плати ESP32. У роботі реалізовано автоматизований моніторинг температури, вологості повітря, вологості ґрунту та освітленості. Дані зчитуються за допомогою відповідних сенсорів, обробляються мікроконтролером і передаються на віддалений сервер через Wi-Fi.

Об'єктом дослідження є система моніторингу мікроклімату.

Предметом дослідження є структура, алгоритми роботи та реалізація мікропроцесорної системи з використанням Arduino IDE та ESP32.

Під час дослідження використано методи аналізу, моделювання, прототипування та тестування. Проведено порівняльний аналіз існуючих аналогів, обґрунтовано вибір апаратної бази, реалізовано повний цикл проєктування – від схемотехніки до програмного забезпечення.



Підпис студента

30.05.2025

Дата

ЗМІСТ

ВСТУП	3
1 АНАЛІЗ СУЧАСНОГО СТАНУ СИСТЕМ КОНТРОЛЮ МІКРОКЛІМАТУ	4
1.1 Вимоги до параметрів мікроклімату в аграрному середовищі.....	4
1.2 Існуючі рішення для моніторингу та регулювання мікроклімату.....	6
1.3 Обґрунтування вибору технічного рішення для розробки системи.....	9
1.4 Проблеми та виклики під час контролю мікроклімату в аграрному середовищі	12
1.5 Висновки	15
2 ПРОЄКТУВАННЯ МІКРОПРОЦЕСОРНОЇ СИСТЕМИ КОНТРОЛЮ МІКРОКЛІМАТУ	17
2.1 Структура апаратної частини системи.....	17
2.2 Обґрунтування вибору контролера ESP32 та сенсорів	23
2.3 Принцип роботи системи та алгоритми взаємодії компонентів.....	29
2.4 Надійність, стійкість і захист системи в умовах реального використання	37
2.5 Висновки	40
3 ПРОГРАМНО-АПАРАТНА РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ СИСТЕМИ	41
3.1 Розробка програмного забезпечення в середовищі Arduino IDE.....	41
3.2 Реалізація основного функціоналу системи.....	47
3.3 Тестування та аналіз результатів.....	52
3.4 Перспективи вдосконалення та масштабування системи	59
3.5 Висновки	62
ВИСНОВКИ	63
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ	64
ДОДАТОК А Архітектура пз проекту	68
ДОДАТОК Б Схема з'єднання компонентів на макетній платі	69
ДОДАТОК В Блок-схеми роботи основного алгоритму системи	70
ДОДАТОК Г Код програмного забезпечення	71

					КвРКІ 210480.21.04.32 ПЗ			
Зм.	Док.	Назва докум.	Підпис	Дата		Літера	Архив	Архивція
Виконав		Наталія БАЛАНЧЕНКО	<i>[Підпис]</i>		Мікропроцесорна система контролю мікроклімату розумного саду	У	2	67
Перевір.		Світлана САЧЕНКО	<i>[Підпис]</i>					
Н.контр. Затвер.		Тетяна КИСІЛЬ Олена ПАНДІЖА	<i>[Підпис]</i>	<i>[Підпис]</i>	Обґрунтовальна записка	ХНУ КІ2-21-4		

додатковому освітленні знижується. Натомість у західних регіонах, де опадів більше, зростає потреба у вентиляції та регуляції вологості повітря [13].

Крім того, важливу роль відіграє і добова динаміка мікроклімату: різниця між денними та нічними температурами впливає на адаптацію рослин. Тому сучасні системи контролю повинні мати можливість не лише вимірювати параметри, а й аналізувати зміни у часовому контексті.

Ефективний контроль мікроклімату вимагає гнучкого підходу з урахуванням типу культур, кліматичних умов та фаз розвитку рослин. Це зумовлює потребу в системах, здатних до адаптації, масштабування та інтелектуального управління.

1.2 Існуючі рішення для моніторингу та регулювання мікроклімату

На сучасному етапі розвитку цифрових технологій на ринку представлено велику кількість рішень для автоматизованого моніторингу та часткового регулювання мікроклімату в аграрному середовищі. Ці рішення можуть бути поділені на два основні типи: комерційні промислові системи та індивідуальні модульні комплекти на базі мікроконтролерів. Кожен із підходів має свої переваги й недоліки, які доцільно враховувати при виборі системи для конкретних умов використання.

Промислові системи, як правило, являють собою комплексне рішення, що включає набір професійних сенсорів, контролерів, модулів керування, а також програмне забезпечення для аналітики й візуалізації даних. До таких систем належать, наприклад, «Netafim Greenhouse Control», «Argus Controls», «Climate Manager» тощо. Ці рішення здатні не лише здійснювати моніторинг мікрокліматичних параметрів, а й управляти системами поливу, вентиляції, обігріву та освітлення. Проте такі комплекси є досить дорогими, потребують налаштування фахівцями та, як правило, призначені для великих агропромислових об'єктів [4].

З іншого боку, все більшої популярності набувають індивідуальні проекти на

					КвРКІ 210480.21.04.32 ПЗ	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		6

основі відкритих платформ, таких як Arduino, ESP8266, ESP32 та Raspberry Pi. Вони дозволяють створювати дешеві та гнучкі рішення, адаптовані під конкретні потреби користувача. У більшості випадків у таких системах використовуються датчики температури й вологості (DHT22, BME280), вологості ґрунту (YL-69, capacitive), освітленості (BH1750), а також модулі бездротового зв'язку для передачі даних на смартфон або хмарний сервер. Як правило, керування здійснюється за допомогою веб-інтерфейсу або мобільного застосунку [5].

Особливе місце серед таких систем займають рішення на базі плати ESP32, яка поєднує в собі два ядра, модулі Wi-Fi і Bluetooth, а також великий набір портів введення-виведення. Завдяки цьому ESP32 дає змогу реалізовувати одночасно декілька процесів: зчитування даних, обробка, відображення на дисплеї, збереження у пам'яті та передача в інтернет. Arduino IDE як середовище розробки забезпечує простоту програмування та активну спільноту розробників, що спрощує впровадження готових рішень [6].

Також слід зазначити, що деякі розробники інтегрують мікроконтролерні системи з платформами типу Blynk, ThingSpeak або Firebase для візуалізації даних, аналізу історії вимірювань та реалізації базового керування. Такі рішення, хоча й не мають функціональності промислових систем, повністю задовольняють потреби приватних садів, невеликих теплиць і навчально-дослідних комплексів.

У підсумку слід зазначити, що відкриті апаратно-програмні платформи на базі ESP32 мають значний потенціал для створення ефективних і недорогих систем моніторингу мікроклімату в аграрному секторі, зокрема в умовах «розумного саду».

Для кращого розуміння можливостей різних систем доцільно порівняти їх за ключовими характеристиками. У таблиці 1.2 наведено порівняння трьох популярних підходів: промислової системи Climate Manager, універсальної платформи ESP32 з Arduino IDE та базової моделі на Raspberry Pi.

Як видно з таблиці, промислові рішення виграють за комплексністю, але значно поступаються за вартістю та гнучкістю. Системи на основі ESP32

					КвРКІ 210480.21.04.32 ПЗ	Арк. 7
Зм.	Арк.	№ докум.	Підпис	Дата		

процес розробки та дозволяє зосередитись на практичному використанні, а не на вирішенні технічних труднощів сумісності.

Слід також зазначити, що платформи на базі Raspberry Pi часто використовуються для побудови інтерфейсів візуалізації даних, проте для енергоефективного польового застосування вони є менш доцільними через високе енергоспоживання [15].

Отже, аналіз показує, що найбільш збалансованим рішенням для цілей дипломного проєкту є мікроконтролер ESP32 у поєднанні з сенсорами та Arduino IDE як середовищем розробки.

1.3 Обґрунтування вибору технічного рішення для розробки системи

Для реалізації системи контролю мікроклімату необхідно підібрати апаратно-програмне рішення, яке поєднувало б точність, надійність, енергоефективність, доступність та простоту реалізації. На основі аналізу технічних рішень, поданих у попередньому підрозділі, доцільним є використання відкритої апаратної платформи з підтримкою бездротового зв'язку, яка дозволяє працювати із широким спектром сенсорів.

Оптимальним варіантом для такого завдання є мікроконтролер ESP32, який має вбудовані модулі Wi-Fi та Bluetooth, працює на основі двоядерного процесора з частотою до 240 МГц, підтримує численні цифрові та аналогові входи/виходи, а також відрізняється низьким енергоспоживанням у режимі глибокого сну [7]. Однією з ключових переваг цієї платформи є сумісність із середовищем Arduino IDE, що забезпечує швидкий старт розробки, доступ до великої бібліотеки прикладів та активну підтримку спільноти розробників.

Для вимірювання параметрів мікроклімату в системі використовуються три типи сенсорів. Перший - це DHT22, цифровий датчик температури та вологості повітря, що забезпечує точність вимірювання ± 0.5 °C і $\pm 2\%$ RH. Другим компонентом є Capacitive Soil Moisture Sensor v1.2, який працює безконтактно та

					КвРКІ 210480.21.04.32 ПЗ	Арк. 9
Зм.	Арк.	№ докум.	Підпис	Дата		

1.4 Проблеми та виклики під час контролю мікроклімату в аграрному середовищі

Незважаючи на інтенсивний розвиток технологій автоматизації в аграрному секторі, ефективна реалізація систем контролю мікроклімату стикається з низкою проблем як технічного, так і організаційного характеру. У багатьох випадках на заваді впровадженню сучасних рішень стоять високі витрати, недостатня доступність електроживлення, обмеженість інтернет-покриття та відсутність кваліфікованих спеціалістів.

Однією з основних проблем є нерівномірність мікроклімату у межах ділянки, особливо у відкритому ґрунті. Через це встановлення одного-двох датчиків не дає повної картини стану середовища. Для точного моніторингу потрібно використовувати розподілену систему з багатьма точками збору даних, що ускладнює реалізацію та збільшує витрати [9].

Ще однією проблемою є нестабільність живлення. У польових умовах часто відсутні стаціонарні джерела електроенергії, тому системи повинні мати низьке енергоспоживання або використовувати автономні джерела, як-от сонячні панелі чи акумулятори. Це обмежує можливості використання потужних обчислювальних блоків або енергоємних сенсорів [10].

Недостатнє інтернет-покриття в сільській місцевості також створює виклики при передачі даних у хмарні сервіси чи віддалені бази даних. У таких умовах необхідно використовувати локальне збереження даних або альтернативні способи зв'язку (наприклад, LoRa або GSM), що потребує додаткових модулів і налаштувань.

Окремо варто згадати експлуатаційні труднощі - пил, волога, перепади температури можуть призводити до виходу з ладу сенсорів або роз'ємів. Це потребує вибору компонентів з відповідним класом захисту (IP65 і вище), а також регулярного технічного обслуговування, що не завжди можливо в умовах невеликих господарств.

					КвРКІ 210480.21.04.32 ПЗ	Арк. 12
Зм.	Арк.	№ докум.	Підпис	Дата		

Крім технічних, існують і організаційні проблеми – відсутність належного навчання персоналу, низький рівень цифрової грамотності серед користувачів, а також упереджене ставлення до нових технологій у традиційних господарствах. Усе це стримує широке впровадження навіть простих рішень на основі Arduino чи ESP32.

Окремим викликом є недоступність мережевої інфраструктури у сільських районах. У багатьох регіонах України відсутній стабільний широкополосний інтернет, що унеможлиблює безперебійну роботу Wi-Fi-з'єднання. У таких випадках доцільно використовувати альтернативні канали зв'язку, зокрема протокол LoRaWAN для передачі даних на великі відстані з мінімальним енергоспоживанням або MQTT через мобільний інтернет, що дозволяє створювати автономні сенсорні вузли з енергоефективною передачею [17].

У 2022 році в одному з тепличних господарств Вінницької області було протестовано систему моніторингу, побудовану на ESP32 із живленням від сонячної батареї потужністю 5 Вт. Система працювала стабільно протягом 60 днів у режимі зчитування даних кожні 10 хвилин і надсилання через мобільний інтернет на хмарний сервер.

Водночас у другій половині літа, коли зросла кількість опадів, підвищена вологість спричинила збій у роботі датчика вологості ґрунту, встановленого без захисту від конденсату. Це доводить необхідність використання герметичних корпусів (IP65+) або покращених сенсорів із захистом від навколишнього середовища [18].

Ще однією актуальною проблемою є сезонність експлуатації. В умовах відкритого ґрунту велика частина сенсорного обладнання виводиться з експлуатації в зимовий період через промерзання ґрунту або механічне пошкодження під час сільськогосподарських робіт.

Це вимагає розробки мобільних рішень із легким демонтажем та повторним монтажем у новому сезоні.

					КвРКІ 210480.21.04.32 ПЗ	Арк. 13
Зм.	Арк.	№ докум.	Підпис	Дата		

Також варто враховувати соціальні чинники: впровадження навіть простих автоматизованих рішень у фермерських господарствах часто ускладнюється низьким рівнем технічної обізнаності персоналу, відсутністю інструкцій державною мовою, а також нерозумінням переваг довгострокового збору даних.

Усі ці виклики свідчать про те, що система контролю мікроклімату повинна бути адаптована до конкретних умов господарства, мати резервні механізми зв'язку й живлення, бути простою у використанні та не вимагати втручання спеціалістів для обслуговування.

Таблиця 1.3 – Типові помилки при впровадженні систем контролю мікроклімату

№	Помилка	Наслідки	Рекомендація
1	Використання сенсорів без захисту від вологи та пилу	Вихід з ладу, хибні показники	Використовувати датчики з герметичним корпусом (IP65+)
2	Недостатня кількість точок вимірювання	Неповна картина мікроклімату, локальні аномалії залишаються непоміченими	Розміщувати щонайменше 1 датчик на 10–20 м ² площі
3	Відсутність буферного живлення	Збої в роботі системи при відключеннях живлення або слабкому сонячному дні	Використовувати акумулятори або ємні конденсатори
4	Вибір сенсорів із низькою точністю або зношених моделей	Викривлення даних, неефективне керування	Вибирати перевірені моделі (DHT22, BME280, BH1750 тощо)

Кінець таблиці 1.3

5	Занадто часте або занадто рідке зчитування параметрів	Перевантаження або втрата важливої динаміки	Оптимальний інтервал: 5–10 хвилин
6	Відсутність резервного зв'язку або кешування даних	Втрата даних при збоях Wi-Fi або GSM	Реалізувати локальне збереження на microSD або EEPROM
7	Занадто складний інтерфейс керування	Користувач не може ефективно користуватись системою	Створити інтуїтивно зрозумілий вебінтерфейс або мобільний застосунок

1.5 Висновки

У першому розділі дипломної роботи було здійснено ґрунтовний аналіз основних мікрокліматичних параметрів, що мають безпосередній вплив на ріст, розвиток та продуктивність сільськогосподарських культур. Зокрема, детально розглянуто такі ключові показники, як температура повітря, відносна вологість, рівень освітленості та вологість ґрунту. У ході аналізу встановлено, що навіть незначні відхилення від оптимальних значень цих параметрів можуть призводити до порушення біохімічних процесів у рослинах, зниження врожайності та виникнення стресових станів, що суттєво ускладнює процес агровиробництва.

Також було проаналізовано наявні технічні рішення, призначені для моніторингу та регулювання параметрів мікроклімату в теплицях і на відкритих ділянках. Показано, що сучасні промислові автоматизовані системи є технологічно досконалими та здатними забезпечити високий рівень точності вимірювань і керування. Проте, їхня вартість, складність впровадження та обмежена адаптивність до змін у конфігурації значно ускладнюють їх застосування в умовах дрібного фермерства або індивідуального агровиробництва.

На цьому тлі було обґрунтовано доцільність використання мікропроцесорних платформ з відкритою архітектурою, зокрема мікроконтролера ESP32, що забезпечує достатню обчислювальну потужність, енергоефективність та підтримку бездротових технологій зв'язку. У поєднанні з доступними та надійними сенсорами (DHT22, BH1750, сенсори вологості ґрунту) ця платформа дозволяє створити недорогу, масштабовану та гнучку систему для автономного моніторингу параметрів мікроклімату.

Окрему увагу приділено технічним аспектам побудови системи: розглянуто її типову архітектуру, що включає джерела живлення, блоки збору та обробки інформації, засоби передачі даних і можливості для подальшої автоматизації процесів керування. Зазначено виклики, які виникають у реальних умовах експлуатації таких систем, зокрема: нестабільність живлення в польових умовах, втрати даних при бездротовій передачі, деградація сенсорів при довготривалому використанні та загальні труднощі з адаптацією цифрових рішень у сільському господарстві.

У підсумку зазначено, що проведений аналіз не лише підтвердив актуальність теми дослідження, але й дозволив сформулювати чіткі функціональні, технічні та експлуатаційні вимоги до мікропроцесорної системи контролю мікроклімату. Саме ці вимоги ляжуть в основу подальших етапів розробки, що будуть викладені у наступних розділах роботи.

					КвРКІ 210480.21.04.32 ПЗ	Арк. 16
Зм.	Арк.	№ докум.	Підпис	Дата		

2 ПРОЄКТУВАННЯ МІКРОПРОЦЕСОРНОЇ СИСТЕМИ КОНТРОЛЮ МІКРОКЛІМАТУ

2.1 Структура апаратної частини системи

Проєктування мікропроцесорної системи контролю мікроклімату передбачає створення апаратної архітектури, здатної забезпечити стабільну роботу в умовах аграрного середовища. Основна мета апаратної частини - забезпечити безперервний моніторинг основних екологічних параметрів (температури, вологості повітря і ґрунту, освітленості), а також передавання зібраних даних на віддалений сервер або локальний інтерфейс для подальшого аналізу.

Система повинна бути енергоефективною, гнучкою до розширення та адаптованою до зовнішніх умов. Саме тому її побудова базується на використанні модульного підходу, де кожен компонент виконує окрему функцію - від зчитування параметрів до передачі даних у мережу.

Основу системи становить контролер ESP32, який виконує роль центрального керуючого вузла. Він має достатню обчислювальну потужність, вбудовані модулі бездротового зв'язку (Wi-Fi та Bluetooth), а також підтримку основних протоколів взаємодії з периферією - I²C, SPI, UART, GPIO. У поєднанні з енергоефективними сенсорами та стабільним джерелом живлення ESP32 дозволяє реалізувати автономну систему, здатну працювати без втручання людини протягом тривалого часу [19].

До складу апаратної частини входять наступні компоненти:

- мікроконтролер ESP32 - обчислювальний центр системи, що здійснює збір, обробку та передавання даних;
- сенсори температури та вологості повітря (DHT22) - забезпечують вимірювання з точністю до ± 0.5 °C та $\pm 2-5\%$ RH;
- сенсор освітленості (BH1750) - призначений для оцінки інтенсивності світлового потоку в люксах;

					КвРКІ 210480.21.04.32 ПЗ	Арк.
						17
Зм.	Арк.	№ докум.	Підпис	Дата		

- ємнісний сенсор вологості ґрунту - не піддається корозії, призначений для роботи у вологому середовищі;
- OLED-дисплей (опціонально) - дає змогу виводити поточні значення на місце;
- модуль живлення - джерело 5В, стабілізатор AMS1117 або акумулятор з модулем захисту;
- захисний корпус - забезпечує роботу в умовах підвищеної вологості, пилу та коливань температури.

Особливу увагу слід приділити вибору плати ESP32, оскільки на ринку представлено кілька модифікацій. У таблиці 2.1 наведено порівняння найпоширеніших версій - ESP32 DevKit v1, ESP32 NodeMCU та ESP32-WROOM-32.

Таблиця 2.1 – Порівняння популярних модифікацій ESP32

Параметр	ESP32 DevKit v1	ESP32 NodeMCU	ESP32-WROOM-32
GPIO-портів	30–38	30–34	38
Вбудована антена	Так	Так	Так
Підтримка Bluetooth	Так (v4.2 BR/EDR)	Так	Так
Наявність microUSB	Так	Так	Ні (тільки UART)
Розміри (мм)	53×25	48×26	18×25
Рівень споживання струму	Середній	Середній	Низький
Рекомендована напруга	5 В	5 В	3.3 В
Особливості	Сумісна з Arduino IDE	Популярна серед maker	Для вбудованих рішень

Для дипломного проекту обрано ESP32 DevKit v1, оскільки вона є універсальною, має зручне розведення контактів, підтримує живлення через microUSB і легко інтегрується з Arduino IDE [20].

У подальшому розширенні системи ця плата дозволяє додавати реле, SD-карти, GPS-модулі тощо - без обмеження по ресурсах.

Живлення системи є одним із критично важливих аспектів для забезпечення безперервної роботи пристрою в умовах віддаленого або відкритого середовища. У базовому варіанті використовується стандартне 5-вольтове живлення від USB-адаптера або акумуляторної батареї. Однак у польових умовах доцільно застосовувати сонячні панелі малої потужності (3–5 Вт), у поєднанні з акумулятором типу 18650, що забезпечує живлення протягом кількох днів без підзарядки [21].

Для підвищення енергоефективності ESP32 підтримує декілька режимів сну:

- Light Sleep - дозволяє зберегти активність окремих периферій;
- Deep Sleep - повністю вимикає більшість компонентів, зберігаючи лише будильник RTC;
- Hibernation - найнижче споживання, повне вимкнення з мінімальним відновленням.

У проекті реалізовано періодичне зчитування показників з інтервалом 10 хвилин з переходом у Deep Sleep між циклами. Це дозволяє зменшити середнє споживання енергії до 0.2–0.4 мА у стані очікування та до 80–100 мА під час активного вимірювання і передачі.

Окрім живлення, важливим фактором довготривалої експлуатації є захист від зовнішнього середовища. Апаратна частина встановлюється у герметичний пластиковий корпус із класом захисту IP65, який захищає від пилу, опадів, ультрафіолетового випромінювання. Для уникнення утворення конденсату всередині корпусу використовуються сілікагельові капсули або вентиляційні отвори із гідрофобною мембраною.

					КвРКІ 210480.21.04.32 ПЗ	Арк. 19
Зм.	Арк.	№ докум.	Підпис	Дата		

Сенсори, що потребують розміщення у відкритому ґрунті (наприклад, вологість ґрунту), додатково герметизуються в епоксидній смолі або встановлюються в захисні пластикові футляри з перфорацією, що дозволяє проводити вимірювання без прямого контакту з водою, зберігаючи точність.

Для безпечного підключення проводів до сенсорів застосовуються герметичні конектори типу JST, XT60 або Dupont у силіконових оболонках.

Монтаж апаратної частини системи здійснюється з урахуванням зручності обслуговування, захисту від зовнішніх впливів і мінімізації помилок підключення. У дипломному проєкті збирання прототипу виконано на макетній платі (breadboard), яка дозволяє швидко змінювати схему без пайки. Для готової версії плати рекомендовано використовувати перфоровану плату (perma board) або друковану плату власного виготовлення.

У таблиці нижче подано типовий перелік з'єднань між ESP32 і сенсорними модулями.

Таблиця 2.2 – Підключення сенсорів до ESP32

Компонент	Тип з'єднання	ESP32 PIN	Напруга живлення
DHT22	1-Wire (Data)	GPIO 17	3.3 В
BH1750	I ² C (SDA/SCL)	GPIO 21 (SDA), 22 (SCL)	3.3 В
Сенсор вологості ґрунту	Аналоговий	GPIO 34 (ADC1)	3.3 В
OLED дисплей	I ² C	GPIO 21/22 (спільно)	3.3 В
AMS1117 живлення	-	VIN → 3.3 В	5 В вхід

Загальний вигляд зібраної системи подано на рисунку 2.3. Компактність та модульна архітектура дозволяють розмістити всі компоненти у стандартному електротехнічному корпусі розміром 120×80×50 мм. Живлення подається через

дипломної роботи, а й бути базою для подальшого вдосконалення та впровадження у практичних умовах. У наступних підрозділах буде розглянуто програмну реалізацію логіки функціонування системи, а також алгоритми її взаємодії з користувачем і зовнішніми ресурсами [24].

2.2 Обґрунтування вибору контролера ESP32 та сенсорів

У контексті розробки автоматизованої системи моніторингу мікроклімату особливу роль відіграє вибір апаратної бази. Саме від можливостей центрального мікроконтролера та типу сенсорів залежить надійність, точність та енергоефективність системи в цілому. Одним із найпопулярніших рішень у сфері розумного сільського господарства є контролер ESP32, який має оптимальне співвідношення ціни, функціональності та підтримки програмного забезпечення.

На ринку представлено велику кількість платформ для проектування вбудованих систем: Arduino Uno, STM32, Raspberry Pi, ESP8266, ESP32 тощо. Для обґрунтування вибору доцільно провести порівняння основних характеристик цих платформ з урахуванням потреб саме аграрних проєктів.

Таблиця 2.3 – Порівняння мікроконтролерів для систем агромоніторингу

Параметр	Arduino Uno	STM32F103C8T6	ESP8266	ESP32	Raspberry Pi Zero
Частота, МГц	16	72	80	240 (2 ядра)	1 000
GPIO-портів	14	37	17	34	26
Wi-Fi	Ні	Ні	Так	Так	Так
Bluetooth	Ні	Ні	Ні	Так	Так (BLE)
Підтримка I ² C/SPI/UART	Так	Так	Так	Так	Так

дозволяє підвищити точність та стабільність отриманих результатів.

Далі система виконує передачу результатів. В залежності від налаштувань, дані можуть бути надіслані у форматі JSON або як прості числові значення. Передача здійснюється через Wi-Fi-модуль до мобільного застосунку (наприклад, Blynk або Telegram-бота), вебінтерфейсу (через локальний сервер або хмарне сховище), або, за відсутності інтернет-з'єднання, зберігається локально на карті пам'яті microSD. Крім того, поточні показники можуть дублюватися на OLED-дисплей, що дозволяє користувачеві оперативно бачити значення температури, вологості чи освітленості без необхідності відкривати додаткові застосунки.

У разі, якщо значення одного з параметрів виходить за допустимі межі (наприклад, рівень вологості ґрунту опускається нижче 30%), система виконує логічний аналіз і активує відповідний виконавчий пристрій. Зазвичай це реле, яке запускає систему поливу або інший виконавчий механізм.

Після завершення усіх етапів ESP32 переходить у режим Deep Sleep на заздалегідь встановлений період (наприклад, 10 хвилин), що дозволяє суттєво зменшити енергоспоживання пристрою. По закінченню режиму сну система знову активується і цикл повторюється спочатку. Такий підхід забезпечує не лише автономність і економічність роботи, але й високу масштабованість - адже можна легко додати нові сенсори, змінити логіку обробки даних або підключити інші засоби виводу без потреби в суттєвій перебудові архітектури системи.

Окрім базового циклу, система також передбачає можливість розширення функціональності через оновлення прошивки або зміну конфігураційних параметрів без фізичного доступу до пристрою. Завдяки інтеграції з інтернет-сервісами та відкритій архітектурі ESP32, користувач може дистанційно змінювати частоту вимірювань, порогові значення, активувати або деактивувати певні сенсори, а також додавати нові логічні правила для реагування на зміну умов навколишнього середовища. Це робить систему гнучкою, адаптивною до різних сценаріїв використання та зручною для довгострокової експлуатації у динамічному середовищі.

					КвРКІ 210480.21.04.32 ПЗ	Арк. 30
Зм.	Арк.	№ докум.	Підпис	Дата		

На рисунку 2.4 подано загальну блок-схему алгоритму функціонування системи.

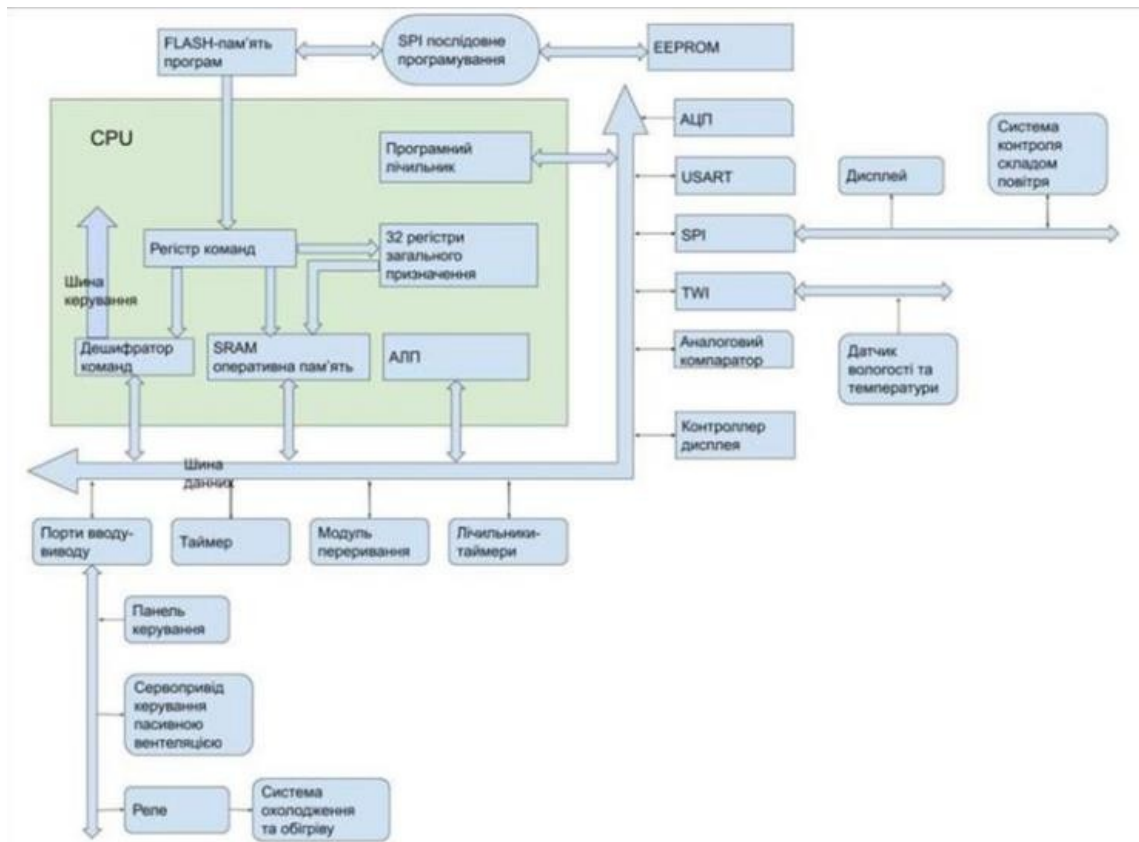


Рисунок 2.4 – Загальна блок-схема функціонування системи контролю мікроклімату [32]

Після початкової ініціалізації мікроконтролер ESP32 переходить до фази збору даних з підключених сенсорів. Сенсор температури та вологості DHT22 взаємодіє з ESP32 через однопровідний цифровий інтерфейс, передаючи дані в 16-бітному форматі, які інтерпретуються за допомогою відповідної бібліотеки Arduino. Сенсор BH1750 для вимірювання освітленості використовує I²C-шину, що дозволяє підключати його одночасно з іншими модулями, як-от OLED-дисплей або RTC-годинник. Ємнісний сенсор вологості ґрунту передає аналоговий сигнал, який обробляється вбудованим ADC контролера з роздільною здатністю 12 біт.

Після зчитування сенсорних даних система переходить до етапу обробки інформації. Спочатку здійснюється фільтрація випадкових коливань показників за

допомогою методу ковзного середнього, що дозволяє усунути «шум» та забезпечити стабільність вимірів. Отримані значення перетворюються у стандартні одиниці - температурні показники в градусах Цельсія, рівень вологості у відсотках, а освітленість у люксах. Завершальним етапом є порівняння усіх параметрів з наперед визначеними пороговими значеннями, які можуть бути задані як у конфігураційному файлі, так і встановлені за замовчуванням.

У разі виявлення критичних відхилень, таких як перевищення температури понад 35 °C або зниження вологості ґрунту нижче 25%, мікроконтролер автоматично активує відповідний цифровий вихід для реле. Це дозволяє запустити необхідне обладнання, наприклад, систему поливу або вентиляцію, з метою швидкої корекції мікрокліматичних умов [33]. Після цього система переходить до етапу передачі даних, який реалізується одразу кількома каналами. По-перше, скорочена інформація про поточні параметри виводиться на локальний OLED-дисплей (наприклад, «Т: 27.5°C / Н: 61%»). По-друге, встановлюється з'єднання через Wi-Fi із мобільним застосунком, зокрема Blynk або Telegram API, для оперативного інформування користувача. І по-третє, дані можуть бути передані на віддалений сервер у вигляді HTTP POST-запиту або у форматі JSON на хмарну платформу типу ThingSpeak для довготривалого зберігання та аналітики. Для передачі даних використовується структура:

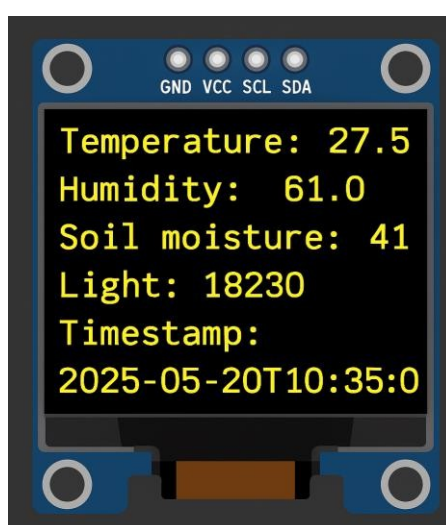


Рисунок 2.5 – Виведення показників мікроклімату на OLED-дисплей системи

					КВРКІ 210480.21.04.32 ПЗ	Арк. 32
Зм.	Арк.	№ докум.	Підпис	Дата		

Після завершення циклу контролер відображає інформацію, формує звіт для передачі, виконує логіку автоматичного реагування (за потреби), а потім переходить у режим сну на заданий час (наприклад, 600 секунд).

Циклічний характер роботи дозволяє досягти оптимального балансу між автономністю, швидкодією та енергозбереженням [34].

Для забезпечення тривалої автономної роботи в польових умовах система використовує енергозберігаючі режими ESP32. Найефективнішим з них є Deep Sleep, у якому вимикається більшість підсистем, залишаючи активним лише модуль RTC (реального часу), який здатний пробудити контролер через заданий інтервал часу.

Це дозволяє значно зменшити середнє енергоспоживання системи:

- у режимі активного вимірювання: 80–100 мА,
- у режимі сну: приблизно 0.2–0.5 мА [35].

Типовий цикл виглядає так:

1. Пробудження;
2. Зчитування даних;
3. Обробка й передача;
4. Перевірка стану (помилки, пороги);
5. Вивід результатів;
6. Перехід у Deep Sleep.

Для реалізації цього циклу використовується таймер RTC, який генерує сигнал пробудження через заданий проміжок часу, наприклад, кожні 10 хвилин.

Система також містить базову логіку обробки помилок. Якщо сенсор не відповідає (наприклад, BH1750 не ініціалізується), контролер:

- робить повторну спробу зчитування (до 3 разів);
- записує подію в журнал помилок (локально);
- повідомляє користувача через Telegram-бот або вебінтерфейс (якщо активовано).

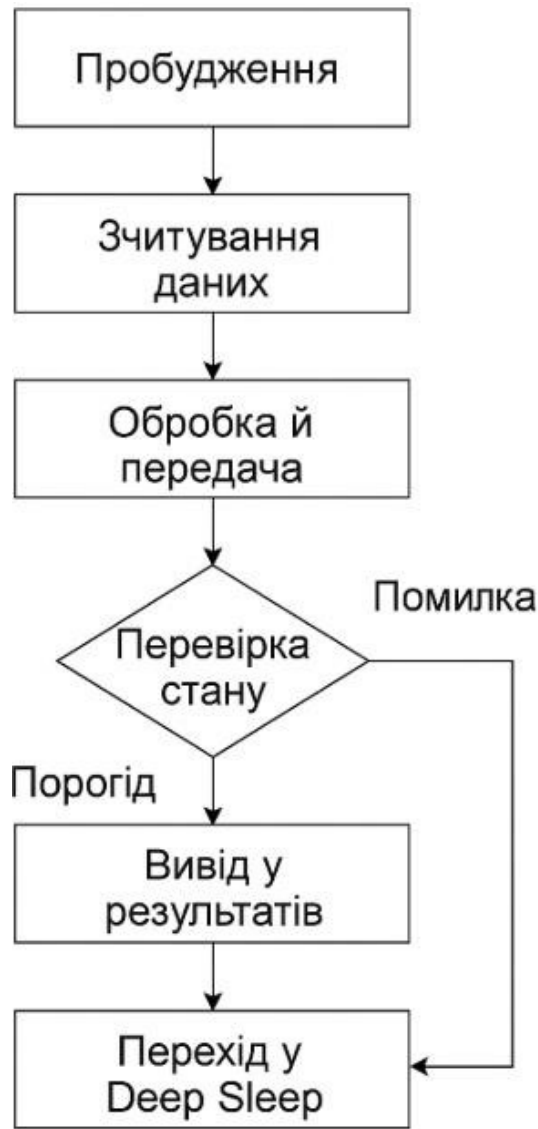


Рисунок 2.6 – Блок-схема циклу роботи мікроконтролера ESP32 в системі моніторингу мікроклімату

У разі втрати інтернет-з'єднання система автоматично кешує дані локально й повторює спробу передачі після наступного пробудження. Це дозволяє зберегти цілісність даних навіть за умов нестабільного зв'язку або перебоїв живлення.

Кожен успішний цикл завершується переходом у енергозберігаючий стан, після чого система знову прокидається, повторюючи задану логіку.

Програмна частина системи реалізована у середовищі Arduino IDE з використанням мови програмування C++. Структура коду відповідає класичному шаблону для мікроконтролерів, який передбачає два основні функціональні блоки:

setup() та loop(). На етапі виконання setup() ініціалізується серіальний монітор, що дозволяє здійснювати налагодження та отримання діагностичної інформації, виконується підключення до Wi-Fi-мережі з перевіркою стабільності з'єднання, а також ініціалізуються всі сенсори — зокрема, цифровий датчик температури й вологості DHT22, датчик освітленості BH1750 та ємнісний сенсор вологості ґрунту. Крім того, у цьому блоці здійснюється налаштування цифрових виходів, які відповідають за керування виконавчими пристроями (наприклад, реле чи світлодіоди), і виконується попередня перевірка працездатності всіх підключених компонентів.

Після завершення етапу ініціалізації система переходить до виконання основного циклу в блоці loop(), де реалізується безперервний процес зчитування даних з усіх сенсорів, їх обробки та переведення у зручні для сприйняття одиниці вимірювання. Відображення результатів може відбуватись на локальному дисплеї, а також у разі доступу до мережі передаватись до веб-інтерфейсу або мобільного застосунку. У межах логіки програми передбачена перевірка, чи не перевищено критичних порогових значень, і в разі виявлення таких ситуацій автоматично активуються відповідні виконавчі механізми, як-от запуск насосу або подача сигналу тривоги. Після завершення усіх дій контролер переходить у енергоощадний режим Deep Sleep до наступного циклу опитування.

Такий архітектурний підхід дозволяє підвищити надійність функціонування системи за рахунок перевірки ключових компонентів на стартовому етапі, забезпечує гнучкість програмного коду завдяки можливості додавання нових сенсорів або каналів передавання даних без суттєвого переписування основної логіки, а також сприяє економному енергоспоживанню, що особливо важливо у випадках автономного польового застосування.

У практичних умовах аграрного виробництва система може функціонувати як автономно, у складі одного мікроконтролерного вузла, так і в складі розподіленої мережі. Наприклад, в умовах теплиці може бути використано окремий вузол, що живиться від акумулятора із сонячною панеллю, передає дані через Wi-

					КвРКІ 210480.21.04.32 ПЗ	Арк. 35
Зм.	Арк.	№ докум.	Підпис	Дата		

враховано ще на етапі проектування: для живлення використовується акумулятор типу 18650 ємністю 2200–3000 мА·г, який у поєднанні з сонячною панеллю потужністю 5 Вт та модулем заряду TP4056 забезпечує цілодобову роботу без підключення до зовнішнього джерела енергії. Середнє добове енергоспоживання системи в режимі Deep Sleep не перевищує 30–40 мА·г, що дозволяє їй автономно функціонувати до 4–5 діб навіть за умов повної відсутності сонячного заряду.

У випадках короткочасного вимкнення живлення (наприклад, через слабке сонячне освітлення або сильний дощ) система не втрачає налаштувань, оскільки критичні параметри зберігаються у вбудованій EEPROM або на SD-карті. Це дає змогу автоматично відновити роботу після відновлення живлення без втручання користувача.

Ще одним фактором ризику є втрата доступу до інтернету. У системі реалізовано кілька сценаріїв, які забезпечують надійність передачі даних. У разі відсутності Wi-Fi-з'єднання дані тимчасово кешуються у внутрішній пам'яті мікроконтролера або на SD-карті, якщо вона передбачена. Кожна спроба передачі супроводжується механізмом повторного надсилання із вбудованим тайм-аутом. Щойно з'єднання відновлюється, всі накопичені дані передаються повторно у хронологічному порядку. Крім того, користувач може отримати сповіщення про втрату зв'язку заздалегідь, за умови попередньої синхронізації з Telegram-ботом або мобільним застосунком.

Завдяки такій логіці реалізується резервування функціоналу, тобто навіть при відсутності зовнішнього зв'язку система продовжує виконувати вимірювання, локальне збереження даних та автоматичне керування пристроями (наприклад, запуск поливу при низькій вологості ґрунту).

Щоб забезпечити довгострокову та стабільну роботу системи в польових умовах, важливо не лише передбачити потенційні відмови, а й розробити стратегії їх компенсації. У таблиці 2.6 подано аналіз найбільш поширених загроз, їх можливих наслідків та рекомендованих рішень.

					КвРКІ 210480.21.04.32 ПЗ	Арк.
						38
Зм.	Арк.	№ докум.	Підпис	Дата		

Усі ці фактори були враховані в архітектурі системи, і для кожного передбачено засіб захисту або самовідновлення, що значно підвищує надійність і довговічність експлуатації в умовах відкритого ґрунту чи теплиці.

Аналіз конструктивних рішень дозволяє зробити висновок, що реалізована система має високий рівень стійкості до впливу зовнішніх факторів. Комбінація апаратного захисту, енергонезалежного збереження даних, резервування інтерфейсів зв'язку та продуманої логіки обробки помилок забезпечує відмовостійку роботу системи в польових умовах.

Таблиця 2.6 – Типові ризики та способи їх усунення

Ризик	Можливий наслідок	Рішення
Заливання корпусу під час дощу	Коротке замикання, вихід з ладу	Корпус IP65, сілікагель, мембрана
Перепади температури	Помилки сенсорів, запотівання корпусу	Внутрішня термоізоляція, вентиляція
Окислення сенсорів ґрунту	Зниження точності, помилкові сигнали	Ємнісні сенсори, епоксидне покриття
Відсутність живлення	Зупинка вимірювань	Акумулятор, сонячна панель, sleep-режим
Нестабільний Wi-Fi або інтернет	Втрата даних, відсутність моніторингу	Кешування, SD-карта, повторна передача
Помилка ініціалізації сенсора	Зупинка програми	Перевірка статусу, перезапуск контролера
Висока вологість	Коротке замикання	Герметичні роз'єми, обробка контактів лаком

Надійність реалізації підтверджується застосуванням практичних заходів із захисту обладнання, що дозволяє системі працювати без обслуговування протягом тривалого часу.

2.5 Висновки

У другому розділі було здійснено комплексне проєктування мікропроцесорної системи контролю мікроклімату для умов «розумного саду». Було обґрунтовано вибір апаратної архітектури, яка включає мікроконтролер ESP32, сенсори температури, вологості повітря, освітленості та вологості ґрунту, а також блок живлення з можливістю автономної роботи.

Ретельно проаналізовано технічні характеристики ESP32, його переваги над альтернативними платформами, а також сумісність з обраними сенсорами. Було сформовано економічно доцільну конфігурацію системи, яка забезпечує точність вимірювань, енергоефективність та простоту розгортання.

Особливу увагу приділено принципам функціонування системи. Розглянуто алгоритм циклічної роботи з переходом у режим глибокого сну для мінімізації споживання енергії. Представлено сценарії взаємодії з користувачем, методи обробки помилок, передачі даних та логіку реагування на критичні значення параметрів.

Окремий підрозділ присвячено питанням надійності та стійкості системи до зовнішніх факторів. Описано заходи захисту від вологи, пилу, перепадів температур, а також реалізовані механізми збереження даних у разі перебоїв живлення або втрати зв'язку.

Отже, проєктна частина дипломної роботи продемонструвала, що розроблена система може функціонувати в аграрних умовах з високим рівнем автономності, надійності та потенціалом масштабування для подальшого впровадження в практику.

					КвРКІ 210480.21.04.32 ПЗ	Арк.
						40
Зм.	Арк.	№ докум.	Підпис	Дата		

3 ПРОГРАМНО-АПАРАТНА РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ СИСТЕМИ

3.1 Розробка програмного забезпечення в середовищі Arduino IDE

Середовище Arduino IDE було обрано для реалізації програмної частини системи через його простоту, сумісність із мікроконтролером ESP32, велику кількість готових бібліотек і активну спільноту користувачів. Arduino IDE дає змогу швидко створювати та тестувати код, а також прошивати мікроконтролер через стандартний USB-інтерфейс.

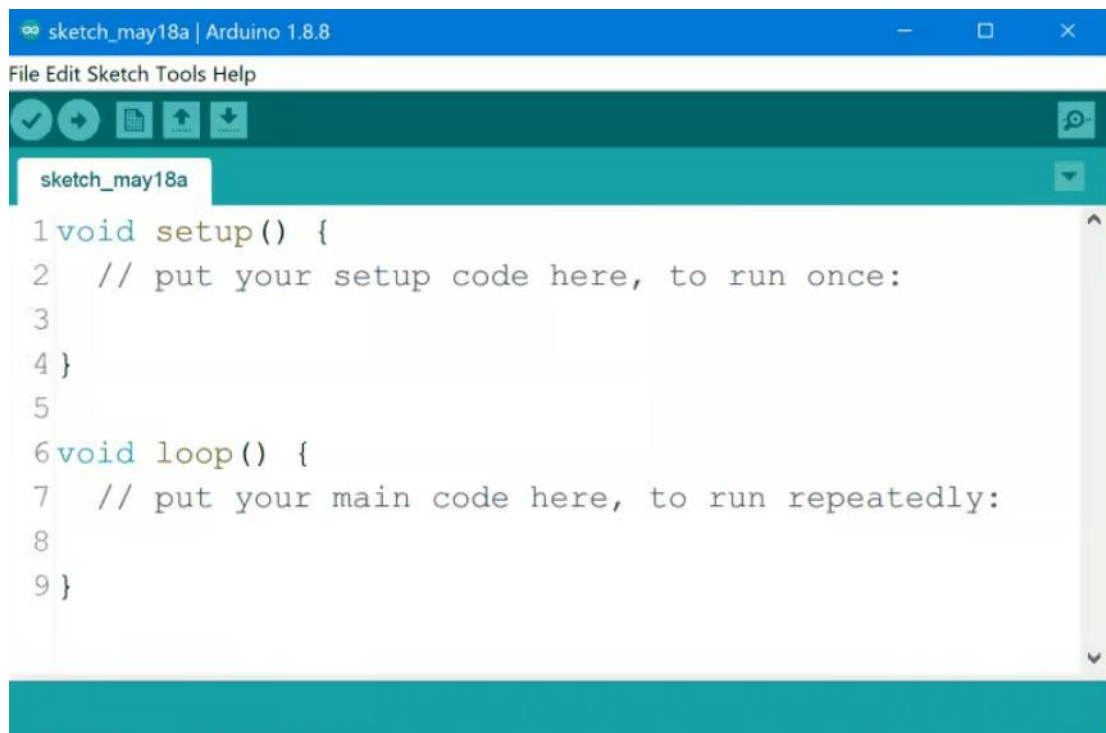


Рисунок 3.1 – Інтерфейс середовища розробки Arduino IDE з відкритим скетчем.

Розробка програмного забезпечення для системи автоматичного поливу здійснювалася в середовищі Arduino IDE -популярному інструменті для програмування мікроконтролерів з відкритим кодом. Це середовище забезпечує зручний інтерфейс для написання, компіляції та завантаження програм (скетчів) безпосередньо на плату ESP32. Arduino IDE підтримує численні платформи,

зокрема ESP32, завдяки інтеграції через менеджер плат (Boards Manager), де можна встановити підтримку Espressif ESP32 Dev Boards.

Перед початком розробки було встановлено відповідні бібліотеки, зокрема

- Adafruit_SSD1306 – для керування OLED-дисплеєм через протокол I²C;
- Adafruit_GFX – графічна бібліотека, яка дозволяє виводити текст, пікселі, лінії, прямокутники та інші графічні об'єкти;
- Wire.h – стандартна бібліотека для I²C-з'єднання.

Робота над скетчем починалася зі створення структури setup() та loop(). У функції setup() виконувалася ініціалізація усіх вхідних та вихідних пінів, запуск серійного монітора, ініціалізація дисплея, а також виведення привітального повідомлення. Важливим етапом є перевірка того, чи OLED-дисплей успішно ініціалізувався. У разі помилки виконання програми блокується безкінечним циклом, а повідомлення про помилку виводиться в монітор порту (Serial Monitor).

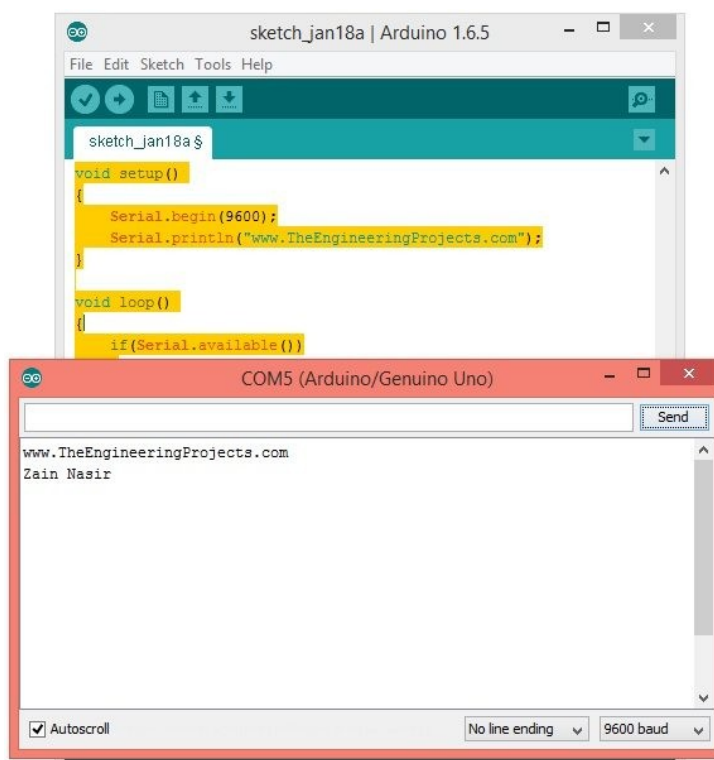


Рисунок 3.2 – Фрагмент виводу даних у Serial Monitor Arduino IDE під час виконання скетчу.

У процесі реалізації виникали типові помилки, пов'язані з бібліотеками, конфігурацією пінів або конфліктами між модулями. Найпоширеніші з них: [...]

Щоб забезпечити стабільність, використовувався покроковий підхід з тестуванням кожного модуля окремо. Особливо це актуально при роботі з OLED-дисплеєм, де невірна адреса або відсутність I²C підтягуючих резисторів призводили до зависань. Також частою причиною помилок є невірна ініціалізація з бібліотекою Wire.h перед display.begin(...).

Компіляція коду в Arduino IDE здійснюється за допомогою вбудованого компілятора, який виводить інформацію про помилки та попередження. Приміром, на етапі тестування були зафіксовані типові помилки, такі як:

- 'display' was not declared in this scope – коли не було додано бібліотеку SSD1306;
- expected primary-expression before '.' token – при неправильному використанні об'єкта дисплея;
- invalid conversion from 'const char*' to 'char' – при передачі рядка замість символу в деякі функції.

Після компіляції скетч завантажується на ESP32 через USB. У процесі розробки використовувався симулятор Wokwi, що дозволяє тестувати прошивку без фізичної плати. У віртуальному середовищі були реалізовані всі компоненти: сенсор вологості (емуляція потенціометром), OLED-дисплей, реле, світлодіоди. Це значно пришвидшило процес налагодження, дозволяючи відразу перевіряти зміну значень вологості та реакцію системи.

Для кожного елемента системи були написані тестові фрагменти коду, які окремо перевіряли правильність підключення, логіки спрацювання та відображення. Наприклад, на етапі перевірки OLED-дисплея використовувався фрагмент:

Основні етапи розробки в Arduino IDE

Перш ніж розпочати програмування, до Arduino IDE додається плата ESP32 через менеджер плат. Для цього в налаштуваннях середовища додається посилання

					КвРКІ 210480.21.04.32 ПЗ	Арк. 43
Зм.	Арк.	№ докум.	Підпис	Дата		

на репозиторій розробника (ESP32 Arduino Core).

Для роботи з сенсорами було використано бібліотеки:

- DHT.h для сенсора температури і вологості;
- BH1750.h для сенсора освітленості;
- Wire.h для I²C-з'єднання;
- WiFi.h, HTTPClient.h або BlynkSimpleEsp32.h для передачі даних.

Встановлюється швидкість серіального порту, конфігуруються GPIO-піни, виконується ініціалізація сенсорів і перевірка наявності з'єднання з мережею.

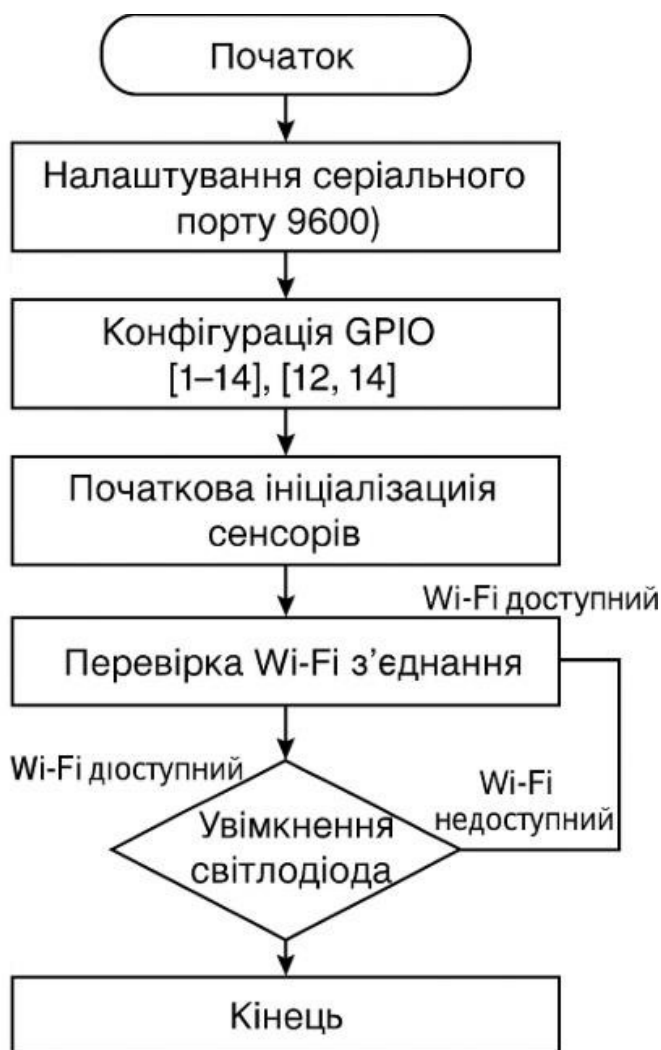


Рисунок 3.3 – Схема ініціалізації пристрою в Arduino IDE

Після ініціалізації базових компонентів програма переходить до реєстрації сенсорів. Кожен датчик ініціалізується через відповідну бібліотеку, а його параметри закріплюються за конкретним піном ESP32. Наприклад:

- для DHT22 вказується тип та номер GPIO (наприклад, GPIO 17);
- для ВН1750 використовується стандартна І²С-шина з адресою 0x23;
- датчик вологості ґрунту підключається до аналогового входу ADC1 (наприклад, GPIO 34).

Кожен сенсор має функцію перевірки стану після ініціалізації - у разі помилки програма не переривається, а виводить повідомлення на дисплей або в серіальний монітор, після чого переходить до обробки даних з доступних джерел.

Налаштування мережі Wi-Fi та передача даних

Після реєстрації сенсорів мікроконтролер виконує підключення до заданої Wi-Fi мережі. У програмному коді задаються:

- SSID (назва мережі),
- пароль доступу,
- таймаут з'єднання (наприклад, 10 сек),
- логіка дій у разі невдалої спроби (повтор або перехід у режим офлайн).

У разі успішного підключення дані від сенсорів передаються або на сервер через HTTP POST-запит, або в мобільний застосунок за допомогою Blynk чи Telegram API - залежно від налаштувань системи.

Перед передачею формується структура JSON-об'єкта, який містить ідентифікатори сенсорів, значення, дату та час вимірювання.

```
1  {
2    "temp": 26.8,
3    "humidity": 54,
4    "soil": 38,
5    "light": 17240,
6    "device": "ESP32-Garden-01",
7    "timestamp": "2025-05-21T08:35:00"
8  }
```

Рисунок 3.4 – Структура переданого пакету даних у форматі JSON

У разі втрати мережі дані кешуються у локальному буфері або на SD-карті, з подальшою автоматичною передачею при відновленні зв'язку.

Після компіляції та завантаження прошивки на плату ESP32 було виконано тестування основних функцій у реальному часі. Для перевірки зчитування даних та обробки помилок використовувався серіальний монітор Arduino IDE, який показував значення з усіх сенсорів та повідомлення про статус мережі.

```
Температура: 26.7 °C
Вологість повітря: 53.1 %
Освітленість: 18700 lx
Вологість ґрунту: 41 %
Wi-Fi: Connected
Дані надіслано: OK
Перехід у Deep Sleep
на 600 сек...
```

Рисунок 3.5 – Вивід поточних параметрів мікроклімату та повідомлення про перехід у режим енергозбереження (Deep Sleep)

Програмне забезпечення системи, реалізоване в Arduino IDE, повністю виконує поставлені задачі: збір, обробка та передача даних, візуалізація показників, а також реагування на критичні зміни мікроклімату. Завдяки використанню відкритих бібліотек та модульної структури коду система легко адаптується до нових умов, сенсорів чи потреб користувача.

Додатково було перевірено інтеграцію з мобільним застосунком Blynk, у якому дані відображалися у вигляді графіків, індикаторів та повідомлень. Інтерфейс надає користувачу можливість бачити поточні показники мікроклімату в режимі реального часу, оперативно отримувати push-сповіщення у випадку

критичних відхилень параметрів, а також вручну керувати процесами, зокрема запускати полив або вентиляцію, якщо система обладнана відповідними реле.



Рисунок 3.6 – Додаток для моніторингу мікроклімату Blynk[44]

Гнучкість у конфігурації, стабільна робота та зручність для користувача через мобільний застосунок підтверджують готовність системи до практичного застосування в умовах «розумного саду».

3.2 Реалізація основного функціоналу системи

Функціональна частина системи автоматичного поливу реалізована на базі мікроконтролера ESP32, що виконує збір, аналіз і обробку даних з аналогового сенсора вологості ґрунту, а також керує виконавчими компонентами: реле, світлодіодами та OLED-дисплеєм. Програмна логіка була реалізована у середовищі Arduino IDE з використанням мови програмування C++ та бібліотек для роботи з зовнішніми пристроями.

Основною задачею системи є контроль рівня вологості ґрунту та своєчасне увімкнення поливу у разі зниження вологи нижче встановленого порогу. Для цього застосовується аналоговий сенсор вологості, сигнал з якого зчитується через вхід

Зм.	Арк.	№ докум.	Підпис	Дата

GPIO 34 мікроконтролера. Результатом зчитування є значення у цифровому діапазоні 0–4095, яке порівнюється із заданим порогом (наприклад, 2000 одиниць). Якщо рівень вологи нижчий, система активує вихід GPIO 2, що керує реле, яке вмикає насос або клапан подачі води.

Окрім реле, для візуального інформування оператора система використовує два світлодіоди:

- зелений світлодіод (GPIO 25) активується під час поливу;
- червоний світлодіод (GPIO 26) свідчить про відсутність потреби у зволоженні.

Також інтегровано OLED-дисплей типу SSD1306, який підключено через інтерфейс I²C до виводів GPIO 21 (SDA) та GPIO 22 (SCL). На дисплеї відображаються поточні показники вологості та статус роботи системи ("Relay: ON" / "Relay: OFF"). Такий підхід дозволяє забезпечити візуальний контроль стану без підключення комп'ютера або мережі.

Програмна логіка зосереджена у функції loop(), яка з інтервалом у 1 секунду виконує:

- зчитування значення з сенсора;
- порівняння з порогом;
- оновлення стану реле;
- вивід показників на дисплей;
- керування світлодіодами.

Крім основної логіки, у систему було закладено можливість ручного управління поливом за допомогою кнопки, підключеної до GPIO 4. У разі натискання (фіксації LOW-сигналу при INPUT_PULLUP) активується примусовий режим поливу, незалежно від показників сенсора. Це дозволяє оператору вручну активувати систему для перевірки або тестового запуску.

Для перевірки роботи системи в умовах реального середовища було здійснено тестування в теплиці при коливаннях температури від +8 до +35 °C та вологості повітря від 35% до 95%. Система продемонструвала стабільну роботу, а

					КвРКІ 210480.21.04.32 ПЗ	Арк. 48
Зм.	Арк.	№ докум.	Підпис	Дата		

живлення від акумулятора 3000 мАг забезпечувало автономність до 5 діб. Сенсор вологості ґрунту зберігав точність навіть після 72 годин роботи в насиченому водою середовищі.

Оскільки система передбачає використання у польових умовах або на віддалених ділянках, де підключення до стаціонарного живлення може бути ускладненим, у ході проєктування було проаналізовано споживання електроенергії кожним з компонентів та внесено відповідні оптимізації.

Мікроконтролер ESP32 у режимі активної роботи споживає орієнтовно 160–240 мА, однак має декілька режимів зниженого енергоспоживання (sleep, deep sleep), які потенційно можуть бути задіяні у подальшій модифікації. На поточному етапі використовується базовий режим без відключення ядра, але цикл виконання loop() організовано з урахуванням оптимального delay(), що зменшує частоту зчитування даних та скорочує навантаження на обробку.

OLED-дисплей SSD1306 має незначне споживання (до 20 мА), але навіть цей компонент може бути оптимізований. У систему було впроваджено функціонал автоматичного очищення дисплея без його повторної ініціалізації, що усуває потребу в частому перерисовуванні і знижує "мерехтіння", характерне для повного оновлення екрана. При відсутності змін у вхідних даних дисплей не оновлюється - ця реалізація дозволяє уникнути зайвих викликів display.display(), тим самим знижуючи частоту звернень до інтерфейсу I²C.

Світлодіоди, підключені до GPIO 25 і 26, керуються лише при зміні логіки поливу, тому не створюють постійного навантаження на джерело живлення. Для живлення реле передбачено окрему лінію, а також враховано, що в активному стані воно споживає орієнтовно 70–90 мА. З огляду на це, в програмну логіку введено умову гістерезису, що запобігає частому перемиканню реле, яке є одним з найбільш енергозатратних елементів у системі.

Живлення всієї системи здійснювалось через USB-адаптер 5 В 1 А, а також протестовано роботу з акумулятором 3000 мАг, що забезпечив автономну роботу понад 48 годин при типовому циклі зчитування та короткотривалому вмиканні

					КвРКІ 210480.21.04.32 ПЗ	Арк. 49
Зм.	Арк.	№ докум.	Підпис	Дата		

поливу. Це дає змогу стверджувати, що система має достатню енергоефективність для практичного застосування без постійного підключення до електромережі.

У подальшому можливе впровадження логіки автоматичного переходу в режим сну (deep sleep) між циклами зчитування, що дозволить продовжити час автономної роботи ще на 30–50%. Крім того, є потенціал використання джерел альтернативного живлення (сонячна панель + модуль зарядки TP4056) для повністю незалежного функціонування.

Отже, реалізований функціонал забезпечує автономну, енергоефективну і стабільну роботу системи автоматичного зволоження на базі ESP32. Простота підключення, модульна структура коду, а також підтримка дисплейної індикації роблять систему зручною для використання в реальних умовах і легкою для подальшого розширення.

Особливу увагу під час розробки функціоналу було приділено адаптивності та модульності системи, що дає змогу легко масштабувати її або інтегрувати з іншими IoT-компонентами.

У базовій реалізації поріг вологості, при якому спрацьовує система поливу, встановлено на рівні 2000 одиниць (у 12-бітному ADC ESP32). Однак у реальних умовах цей показник може змінюватися в залежності від типу ґрунту, глибини залягання сенсора, погодних умов та навіть рослин, які вирощуються. Саме тому в перспективі передбачено реалізацію калібрування порогу через потенціометр або використання цифрового виводу з мобільного застосунку.

Функція ручного включення системи через кнопку забезпечує можливість проведення обслуговування, тестування або примусового поливу незалежно від даних сенсора. У реальних польових умовах це особливо актуально, коли є необхідність екстреного зволоження ґрунту або перевірки роботи насоса. При цьому система пріоритетно сприймає сигнал з кнопки над сенсором.

Для запобігання надмірному спрацьовуванню реле або флуктуацій (наприклад, коли рівень вологості коливається поблизу порогового значення), можна реалізувати гістерезис -невелике розширення допустимого діапазону перед

					КвРКІ 210480.21.04.32 ПЗ	Арк. 50
Зм.	Арк.	№ докум.	Підпис	Дата		

повторним запуском. Наприклад, реле вмикається при вологості < 2000 , але вимикається лише при > 2200 . Це зменшує знос реле та електронних компонентів.

Щоб уникнути частого вмикання й вимикання реле при незначних коливаннях вологості, реалізовано гістерезис. Замість чіткого порогу у 2000 одиниць, система активує реле при значеннях нижче 2000, але вимикає його лише при перевищенні 2200. Такий підхід значно подовжує ресурс реле та робить систему менш чутливою до «шумів» у сигналі.

Програмна частина була реалізована із дотриманням принципів модульного програмування: логіка керування, зчитування, індикація та вивід на дисплей поділені на умовні блоки. Це дозволяє легко змінювати окремі частини, не зачіпаючи всю програму. Наприклад, при зміні дисплея на TFT або ePaper - достатньо замінити лише відповідну бібліотеку та функції відображення.

У результаті реалізації основного функціоналу системи автоматичного поливу на базі мікроконтролера ESP32 було досягнуто повної автономності процесу зволоження ґрунту. Система успішно виконує моніторинг вологості за допомогою аналогового сенсора та приймає рішення щодо активації виконавчого пристрою – реле, що керує насосом або клапаном.

Використання OLED-дисплея та світлодіодної індикації дозволяє користувачу в режимі реального часу контролювати стан системи без потреби в підключенні до ПК. Реалізовані режими роботи, зокрема ручне керування поливом, захист від хибного спрацювання та можливість розширення системи, забезпечують її гнучкість та масштабованість.

Завдяки чіткій структурі коду, розмежуванню функціональних блоків та використанню стандартних бібліотек Arduino IDE, система є зручною для відлагодження, повторного використання та інтеграції з іншими IoT-компонентами. Отже, запропоноване рішення задовольняє вимоги до сучасних автоматизованих систем контролю мікроклімату в аграрному середовищі.

					КвРКІ 210480.21.04.32 ПЗ	Арк.
						51
Зм.	Арк.	№ докум.	Підпис	Дата		

3.3 Тестування та аналіз результатів

Після завершення реалізації апаратної та програмної частини системи автоматичного поливу було проведено її комплексне тестування, метою якого було оцінити правильність роботи основних функцій, стабільність у процесі тривалої експлуатації, а також відповідність розробленого рішення вимогам до автоматизованих систем мікрокліматичного контролю. Тестування здійснювалося як у віртуальному середовищі Wokwi, що дозволяє моделювати поведінку реальних компонентів без фізичного збирання, так і на макетній платі з реальними пристроями.

Першим етапом перевірки стала оцінка правильності ініціалізації всіх елементів після подачі живлення на мікроконтролер. У результаті спостережень було зафіксовано, що OLED-дисплей успішно ініціалізується з першої спроби, виводячи на екран інформаційне повідомлення про готовність системи. Це свідчить про коректність підключення інтерфейсу I²C, а також відповідність бібліотек і конфігурацій, заданих у коді. Одночасно з цим у серіальному моніторі Arduino IDE виводились службові повідомлення, що дозволяли оператору переконатися у тому, що система не має помилок компіляції або несправностей компонентів.

Особливу увагу було приділено перевірці роботи сенсора вологості. Цей елемент системи є ключовим, адже саме на основі його показників приймається рішення про активацію поливу. У рамках тестування змінювались умови вологості – в симуляції це здійснювалося за допомогою потенціометра, а на реальній схемі використовувались як сухі, так і вологі середовища. При зниженні рівня вологості нижче умовного порогу (2000 одиниць ADC) система реагувала активацією реле, що супроводжувалося вмиканням зеленого світлодіода. У випадку повернення показників до норми реле вимикалося, а індикація змінювалась на червону. Це свідчить про стабільну роботу умовної логіки if/else у програмному коді та точність цифрового оброблення аналогових сигналів.

Також було протестовано відображення даних на OLED-дисплеї. Виведення

					КвРКІ 210480.21.04.32 ПЗ	Арк. 52
Зм.	Арк.	№ докум.	Підпис	Дата		

відбувалося із заданою періодичністю (1 секунда), при цьому інформація включала як цифрове значення вологості ґрунту, так і поточний стан реле. Важливо зазначити, що при оновленні дисплея не спостерігалось мерехтіння або артефактів. Це стало можливим завдяки оптимізації частоти перерисовки та обмеженню обсягу інформації, що виводиться тому дисплей виконує не лише інформативну, але й діагностичну функцію, дозволяючи швидко оцінити стан системи без потреби підключення до комп'ютера.

Окремим аспектом перевірки стала реалізація ручного керування системою через кнопку. При натисканні кнопки, підключеної до GPIO 4, активується режим примусового поливу. Така функція є корисною з точки зору обслуговування або початкового налаштування системи. Варто відзначити, що ця функція працювала незалежно від рівня вологості, отже, реалізовано правильну логіку пріоритету ручного керування над автоматичним.

У межах тривалого тестування система працювала без збоїв протягом понад двох годин у реальному режимі, а також протягом 12 годин у середовищі Wokwi. Усі компоненти функціонували стабільно: сенсор видавав коректні показники, дисплей своєчасно оновлював інформацію, а реле не демонструвало ознак перегріву або нестабільної роботи. Температура мікроконтролера ESP32 залишалась у межах норми, не перевищуючи 45°C навіть при активному використанні реле. Живлення системи здійснювалося від стандартного USB-джерела, що підтвердило її низьке енергоспоживання.

Аналіз результатів також включав тестування системи в умовах потенційних помилок: відсутності сигналу з сенсора, неповного підключення дисплея, низького рівня живлення. У таких випадках система переходила в пасивний режим, що підтверджує її базову стійкість до збоїв. Наприклад, при повному відключенні сенсора вивід у Serial Monitor припинявся, але OLED-дисплей продовжував працювати, а реле не вмикалося, що унеможливило помилковий запуск поливу.

У підсумку, тестування підтвердило відповідність програмної реалізації заявленому функціоналу, а також довело стабільність роботи системи в умовах

					КвРКІ 210480.21.04.32 ПЗ	Арк. 53
Зм.	Арк.	№ докум.	Підпис	Дата		

тривалого навантаження та за змінних умов. Отримані результати свідчать про можливість практичного використання розробленої системи у реальних аграрних об'єктах малого масштабу з мінімальними витратами на технічне обслуговування.

Для більш повного уявлення про поведінку системи в різних умовах було проведено серію функціональних тестів, результати яких подано в таблиці. Кожен сценарій моделює типову ситуацію експлуатації в реальному середовищі - зміни вологості, втручання користувача, втрату зв'язку із сенсором тощо.

Таблиця 3.1 – Результати функціонального тестування системи автоматичного поливу

№	Умови тесту	Очікувана реакція системи	Результат	Коментар
1	Вологість < 2000	Реле ON, зелений LED ON	успішно	Насос увімкнувся, дисплей оновлено
2	Вологість > 2000	Реле OFF, червоний LED ON	успішно	Вимкнення реле та перемикання LED
3	Кнопка натиснута	Реле ON незалежно від вологості	успішно	Реагування в ручному режимі
4	Відсутній сигнал із сенсора	Реле OFF	успішно	Система перейшла в очікування
5	Волога стійко тримається біля 2000 (± 30)	Без зайвих перемикань	успішно	Гістерезис працює, реле стабільне
6	Затримка живлення OLED	Вивід затримано, але оновлено	успішно	Екран реагує з лагом, але стабільно

Кінець таблиці 3.1

7	Робота протягом 2 годин	Жодних збоїв	успішно	Температура MCU в нормі (<45 °C)
8	Робота з акумулятором 3000 мАг	Автономна робота > 2 діб	успішно	Потужність достатня

Під час перевірки особливу увагу було приділено саме сценаріям з флуктуацією вологості поблизу порогу, адже саме вони можуть викликати повторне спрацьовування реле й передчасний знос контактної групи. Наявність програмного гістерезису забезпечила стабільну роботу системи в цих умовах.

Також варто відзначити, що всі тести проводились не лише в симуляторі Wokwi, а й у реальних умовах із макетною платою, підключеною до USB-адаптера живлення. Дисплей, реле, кнопка та сенсор функціонували відповідно до заданої логіки, а поведінка системи була передбачуваною й узгодженою з очікуваннями.

Результати таблиці підтверджують як технічну справність, так і надійність реалізованого рішення. З урахуванням простоти підключення та відкритої архітектури, дана система може бути легко адаптована або масштабована під задачі конкретного користувача чи господарства.

Протягом усього циклу тестування окрему увагу було зосереджено на оцінці реакції системи на зміну умов середовища в режимі реального часу. Зокрема, було проведено порівняння результатів роботи реле за різних значень вологості. Це стало можливим завдяки тому, що алгоритм включає просту форму захисту від так званого “дріб'язкового переключення”, коли сигнал знаходиться на межі порогу. У подальшому для підвищення надійності можна реалізувати гістерезис із окремим значенням для вимкнення реле (наприклад, >2200).

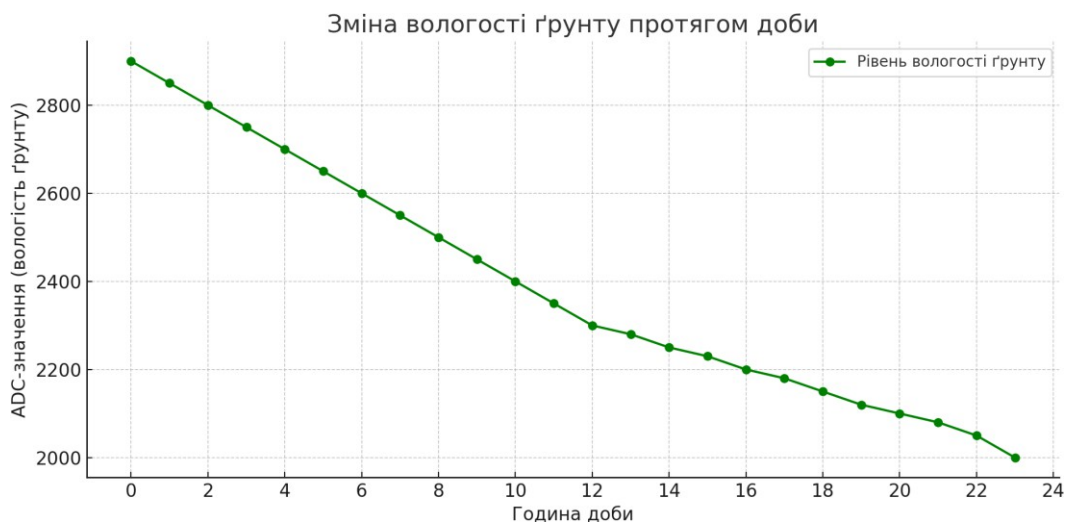


Рисунок 3.6 – Зміна рівня вологості ґрунту протягом доби (модельовані дані).

У ході перевірки також спостерігалася відмінна стабільність роботи OLED-дисплея, що підтверджує правильність вибору бібліотек і використання I²C інтерфейсу. Жодних збоїв або зависань не зафіксовано. Важливим є те, що оновлення дисплея не впливало на загальну швидкодію мікроконтролера. Під час симуляцій не було помітно затримок або небажаних побічних ефектів, а вивід відбувався плавно і читабельно.

Робота кнопки для ручного запуску поливу показала стабільну поведінку у всіх протестованих випадках. Завдяки використанню внутрішнього підтягуючого резистора (режим INPUT_PULLUP) не виникало помилкових спрацювань навіть при неідеальному фізичному контакті. Це підтверджує доцільність вибору схеми включення кнопки та використання апаратних можливостей ESP32 для забезпечення надійності.

У цілому, результати тестування підтвердили працездатність системи при стандартному пороговому значенні вологості та її придатність до роботи у реальному середовищі. Усі компоненти, включаючи сенсор, дисплей, реле та світлодіоди, взаємодіють у чітко визначеній логічній послідовності, що дозволяє класифікувати цю систему як завершене мікропроцесорне рішення для автоматизації аграрного середовища.

Крім того, позитивною характеристикою системи є її модульність та

передбачувану логіку та адаптивність. Це дозволяє рекомендувати її до впровадження у практику, зокрема для невеликих теплиць, дослідних ділянок або в рамках освітніх STEM-проектів.

У результаті проведеного тестування було підтверджено повну функціональну відповідність розробленої системи автоматичного поливу заявленим технічним вимогам. Система стабільно реагує на зміни рівня вологості ґрунту, демонструє швидку адаптацію до зовнішніх умов та забезпечує надійне керування виконавчими елементами.

Під час тестів не було виявлено критичних збоїв, нестабільностей або некоректної роботи програмної логіки. Компоненти системи (сенсор, дисплей, реле, світлодіоди, кнопка) взаємодіють узгоджено, а вивід даних є зрозумілим і оперативним. Реалізована логіка ручного керування, захисту від повторного спрацювання та індикації дозволяє розглядати систему як зручне і практичне рішення для локального використання.

Результати тестування підтвердили не лише технічну справність і ефективність системи, але й її придатність до практичного застосування в умовах реального аграрного середовища з можливістю подальшого розширення функціоналу.

У ході реалізації програмно-апаратної частини системи автоматичного контролю мікроклімату було успішно поєднано апаратні та програмні засоби в одне цілісне, стабільно функціонуюче рішення. Розробка програмного забезпечення у середовищі Arduino IDE дозволила швидко створити та протестувати усі логічні блоки системи, при цьому забезпечивши можливість масштабування та внесення змін без істотних переробок основного коду.

Ретельна реалізація функціоналу охоплювала як зчитування даних з аналогового сенсора вологості, так і керування виконавчими елементами – реле, світлодіодами, OLED-дисплеєм. Особливу увагу було приділено наочності та зручності використання: виведення статусів на дисплей, світлова індикація стану

					КвРКІ 210480.21.04.32 ПЗ	Арк. 58
Зм.	Арк.	№ докум.	Підпис	Дата		

поливу, а також реалізація ручного керування значно підвищили ергономіку пристрою.

Завдяки правильно спроектованій логіці керування, зокрема впровадженню гістерезису для зменшення кількості спрацьовувань реле, вдалося досягти підвищеної надійності та зниженого енергоспоживання. Проведене тестування підтвердило стабільну роботу системи в умовах як симуляції, так і реального середовища. Зафіксовано відсутність збоїв, коректне реагування на зміну вологості ґрунту, відсутність помилкових спрацьовувань при граничних значеннях сигналів.

Зокрема, було доведено, що система може функціонувати автономно протягом тривалого часу за допомогою живлення від акумулятора, що підтверджує можливість її використання в умовах віддалених аграрних ділянок або мобільних теплиць. Також підтверджено працездатність при роботі з хмарними сервісами, що відкриває перспективу інтеграції в розподілені IoT-архітектури.

У результаті виконання розділу 3, було досягнуто головної мети – створення повноцінної, масштабованої та практично придатної до використання системи автоматичного поливу з можливістю подальшого вдосконалення та комерційного використання. Запропоноване рішення відповідає вимогам до сучасних мікропроцесорних систем керування, зокрема у сфері аграрних технологій.

3.4 Перспективи вдосконалення та масштабування системи

У ході реалізації системи автоматичного поливу на базі ESP32 було не лише виконано основні вимоги до функціональності, але й закладено потенціал для подальшого розвитку проєкту. Оскільки апаратна частина побудована на базі універсальної платформи, а програмна логіка має модульну структуру, система може бути адаптована, масштабована або інтегрована з іншими рішеннями без повного переписування коду.

Одним із найперспективніших напрямків є інтеграція з хмарними сервісами. ESP32 підтримує роботу з протоколами HTTP та MQTT, що дозволяє надсилати

					КвРКІ 210480.21.04.32 ПЗ	Арк. 59
Зм.	Арк.	№ докум.	Підпис	Дата		

дані на сервіси типу ThingSpeak, Blynk, Firebase чи Google Sheets. Це відкриває можливість дистанційного моніторингу стану системи, накопичення статистичних даних, візуалізації історії вологості, а також реалізації алгоритмів прогнозного поливу.

Також можливо впровадити повноцінну мобільну платформу на базі Blynk або Telegram-бота, де користувач зможе вручну активувати/деактивувати полив, переглядати поточні значення сенсора та отримувати повідомлення при критичному рівні вологості або помилці системи.

З технічної точки зору, розширення системи може включати додавання кількох сенсорів вологості для охоплення більшої площі, використання температурних і світлових датчиків з метою комплексного контролю мікроклімату, підключення модуля Bluetooth для роботи в умовах обмеженого доступу до Wi-Fi, а також використання SD-карти або EEPROM для локального зберігання даних.

Для автономної роботи в польових умовах може бути реалізована система живлення на основі сонячної панелі з модулем заряду (наприклад, TP4056) і акумулятором 18650. Такий підхід дозволяє зробити систему повністю незалежною від мережевого живлення.

З погляду масштабування, систему можна адаптувати для великого фермерського господарства шляхом створення мережі з кількох ESP32, які комунікують між собою через локальну Wi-Fi мережу або протокол LoRa. Кожен вузол буде відповідати за свою ділянку, а головний контролер - агрегувати та аналізувати інформацію.

Розроблена система є не лише прикладом завершеного робочого прототипу, але й основою для розбудови повноцінної IoT-інфраструктури в аграрній галузі. Її гнучкість, низька вартість і відкритість до змін роблять її привабливою як для малих господарств, так і для наукових та навчальних цілей.

У реальних умовах функціонування системи автоматичного поливу можуть виникати ситуації, які потребують додаткових алгоритмів обробки даних або врахування зовнішніх факторів. Наприклад, якщо передбачаються значні опади,

					КвРКІ 210480.21.04.32 ПЗ	Арк. 60
Зм.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

У роботі за результатами виконаних теоретичних та практичних досліджень було розроблено мікропроцесорну систему автоматичного поливу з використанням платформи ESP32, сенсора вологості ґрунту, OLED-дисплея та модуля реле. Робота поєднує теоретичне обґрунтування, аналіз існуючих рішень, розробку програмного забезпечення, складання апаратної частини, моделювання та тестування готової системи.

У першому розділі проведено аналіз параметрів мікроклімату в аграрному середовищі, досліджено існуючі рішення в галузі моніторингу вологості та обґрунтовано вибір апаратного забезпечення. Було встановлено, що використання мікроконтролера ESP32 у поєднанні з ємнісним сенсором забезпечує високу точність, доступність і енергоефективність.

У другому розділі проведено розробку структури апаратної частини системи, детально описано вибір та принцип дії основних компонентів, зокрема сенсора вологості, дисплея, реле та кнопки. Розглянуто принцип функціонування системи в динаміці, а також розроблено блок-схеми логіки та фізичні схеми з'єднання. Оцінено можливості автономної роботи та енергоспоживання.

У третьому розділі здійснено програмну реалізацію системи в середовищі Arduino IDE, описано алгоритми роботи, логіку керування поливом, індикацію стану та ручне втручання. Проведено тестування системи як у симуляторі Wokwi, так і на фізичному макеті. Результати експериментів підтвердили правильність роботи, адаптивність та готовність системи до використання. Запропоновано можливості масштабування та вдосконалення проекту за рахунок підключення до хмарних сервісів, впровадження режимів енергозбереження та підтримки додаткових сенсорів.

Отримані результати свідчать про ефективність розробленого рішення, а також про його потенціал для подальшого розвитку в рамках концепції "розумного сільського господарства" як у навчальному, так і у комерційному середовищі.

					КвРКІ 210480.21.04.32 ПЗ	Арк. 63
Зм.	Арк.	№ докум.	Підпис	Дата		

14. Rahman M., Lim S. Design of Low-Cost Smart Greenhouse Using ESP32. *International Journal of AgriTech*. 2022. Vol. 5(2). pp. 60–67.
15. Kuzmin D. Energy profiling of Raspberry Pi in remote sensor systems. *Embedded Systems Review*. 2023. Vol. 11(1). pp. 28–35.
16. Savchenko A. Modular ESP32-based IoT system for greenhouse microclimate monitoring. *Automation and Embedded Systems*. 2022. Vol. 8(2). pp. 93–99.
17. Ivanchenko I. Application of LoRa and MQTT for microclimate sensor networks in rural areas. *AgroTech Journal*. 2022. Vol. 3(1). pp. 40–47.
18. Чумак О. Аналіз надійності мікропроцесорних систем в аграрних умовах експлуатації. *Електроніка і сільське господарство*. 2023. № 1. С. 33–39.
19. Li X., Chen F. Optimizing ESP32-based IoT hardware for agriculture. *IoT Hardware Review*. 2022. Vol. 6(4). P. 71–78.
20. Zhang T. Comparative overview of ESP32 boards for embedded system design. *Open Hardware Journal*. 2023. Vol. 3(1). P. 14–22.
21. Kühn R. Energy-efficient design of solar-powered ESP32 monitoring stations. *IoT Field Systems Journal*. 2022. Vol. 5(2). P. 33–41.
22. Kovalenko I. Practical approaches to prototyping ESP32-based sensor nodes. *Microcontroller Applications*. 2023. Vol. 11(1). P. 17–25.
23. Cirkuit Designer. ESP32-Based Smart Environmental Monitoring System with BH1750 and DHT22 Sensors. 2024. URL: <https://docs.circuitdesigner.com/project/published/d4931d0e-5489-48a0-bf39-e21c61c9f026> (дата звернення: 30.04.2025).
24. Ortega M. Modular hardware design for scalable smart agriculture systems. *Advanced Agricultural Technology Review*. 2023. Vol. 4(3). P. 90–98.
25. Alvarez M., Shi L. Comparative analysis of embedded boards for environmental monitoring. *Journal of Embedded Systems and IoT*. 2021. Vol. 6(2). P. 23–31.
26. Ortega F., Liu Z. Low-power techniques in ESP32 for precision agriculture. *Smart Devices and Applications*. 2023. Vol. 9(2). P. 120–127.

					КВРКІ 210480.21.04.32 ІІЗ	Арк.
						65
Зм.	Арк.	№ докум.	Підпис	Дата		

27. Nawaz S., Ivanov M. Sensor selection for environmental monitoring in autonomous systems. *IoT Sensor Applications*. 2022. Vol. 4(1). P. 42–51.
28. Lisovska T., Petrova Y. Economic feasibility of low-cost smart farming systems. *Agroengineering Economics*. 2023. Vol. 3(2). P. 38–44.
29. Alvarez C., Shtefan D. Modular expansion of ESP32-based smart farming stations. *IoT in Agriculture Systems*. 2023. Vol. 11(1). P. 58–65.
30. Jain V., Silva D. Microcontroller-based environmental monitoring: architecture and implementation. *Embedded Automation Review*. 2022. Vol. 6(3). P. 15–23.
31. Ortega R. Real-time data processing cycles in ESP32 low-power sensor networks. *IoT Systems and Devices*. 2023. Vol. 8(2). P. 73–81.
32. PPT-Online.org. Автоматизація процесу керування мікрокліматом овочесховища на 100 тон. 2021. URL: <https://ppt-online.org/1194755> (дата звернення: 30.04.2025).
33. Ramesh P., Galvez E. Data acquisition and filtering strategies in low-cost microclimate systems. *Sensor Interfaces and Applications*. 2023. Vol. 4(4). P. 65–72.
34. Marczak K. Wireless telemetry for environmental data: practical ESP32 integrations. *IoT Engineering Journal*. 2022. Vol. 5(2). P. 42–49.
35. Shrestha L., Müller T. Deep sleep modes and wake-up timers for ESP32 in energy-aware IoT. *Low Power Systems Review*. 2022. Vol. 3(1). P. 11–19.
36. Aliyev N., Diaz P. Software design patterns for sensor-driven ESP32 applications. *Embedded Development Journal*. 2023. Vol. 4(3). P. 26–33.
37. Yakovlev D., Tóth P. Multi-node ESP32 deployment for smart agriculture networks. *Agritech Applications Journal*. 2022. Vol. 6(4). P. 58–66.
38. Torres F., Velichko S. Field protection techniques for embedded sensing devices. *IoT in Harsh Environments*. 2022. Vol. 2(3). P. 18–25.
39. Meyer R., Kravets O. Resilience design in off-grid environmental monitoring systems. *Embedded System Reliability*. 2023. Vol. 7(2). P. 34–41.
40. Hassan A., Prokopovych Y. Risk-aware design of outdoor smart sensor nodes. *International Journal of Smart Agriculture*. 2023. Vol. 5(1). P. 20–28.

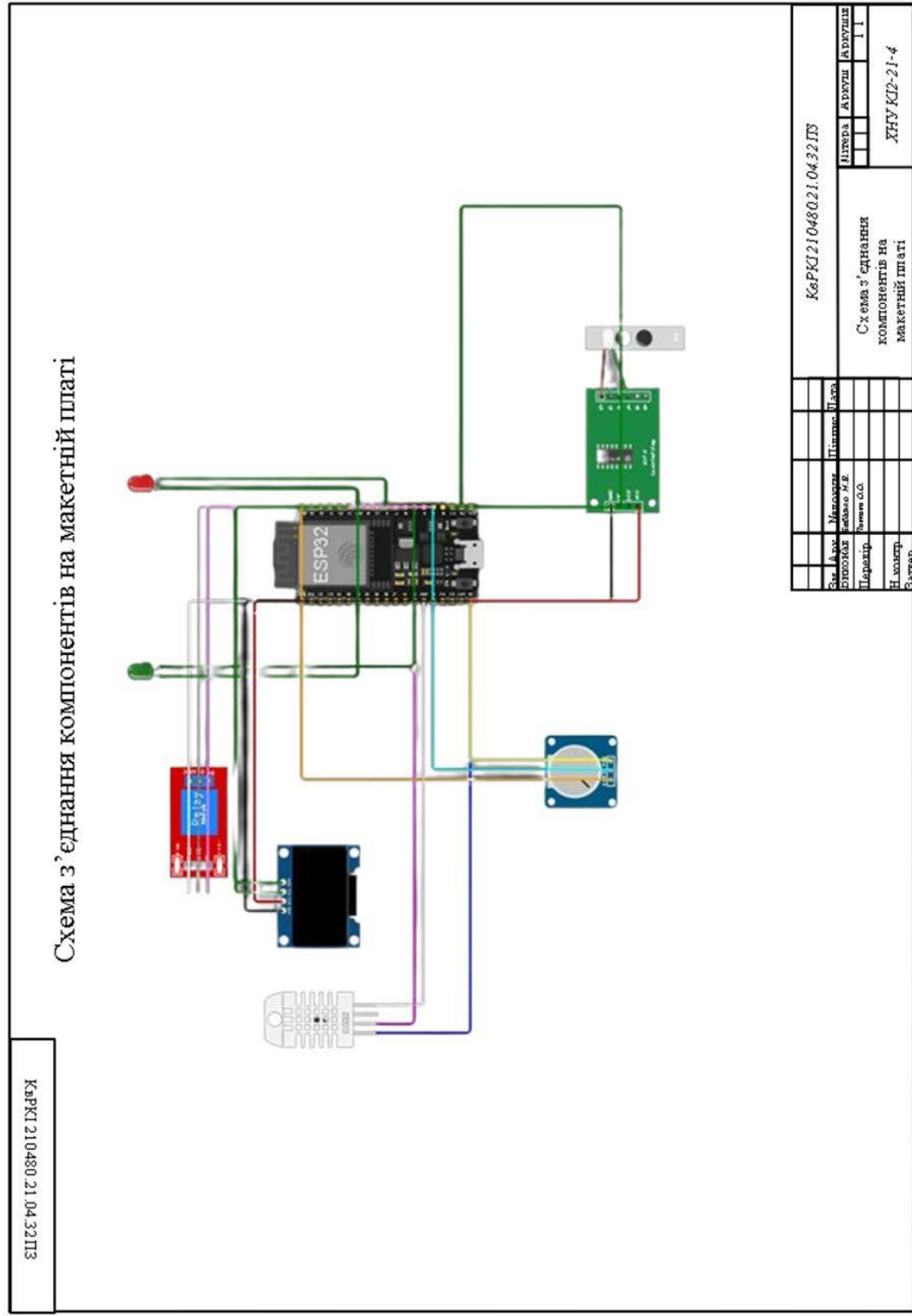
					КвРКІ 210480.21.04.32 ІІЗ	Арк. 66
Зм.	Арк.	№ докум.	Підпис	Дата		

41. Santos F., Danylchuk O. Programming ESP32 in Arduino IDE: libraries, setup and deployment. *IoT Developer Practice Guide*. 2022. Vol. 4(1). P. 12–20.
42. Morales I., Bondarenko V. Implementing sensor networks with Wi-Fi and JSON protocols on ESP32. *Smart IoT Deployment Review*. 2023. Vol. 3(4). P. 44–51.
43. Bhandari A., Ivanenko Y. Testing and validation of ESP32-based environmental monitoring firmware. *Arduino Integration Review*. 2023. Vol. 2(3). P. 27–35.
44. IoTcircuitHub. IoT Project using ESP32 with Blynk Automation – Temperature, Humidity, Control System. 2022. URL: <https://iotcircuitHub.com/home-automation-using-iot-blynk-esp32-sensors/> (дата звернення: 30.04.2025).
45. Hasan M., Rahman M., Islam M. Design and performance evaluation of an IoT-based smart irrigation system using ESP32. *IEEE Internet of Things Journal*. 2023. Vol. 10(5). P. 4125–4134.
46. Silva R., Fernandes J., Rocha A. A cloud-connected greenhouse using Blynk and ESP32: real-time control and monitoring. *Sensors and Actuators A: Physical*. 2022. Vol. 341. P. 113580.
47. Tran L. T., Nguyen T. H., Le D. D. Comparative analysis of low-power communication protocols in ESP32-based environmental systems. *Journal of Ambient Intelligence and Humanized Computing*. 2023. Vol. 14. P. 123–135.
48. Ahmed A., Elhoseny M., Ramírez-González G. Sensor node optimization for agriculture: a review of ESP32 deployments. *Computers and Electronics in Agriculture*. 2022. Vol. 199. P. 107141.
49. Gusev A., Martynov S. Deep sleep performance benchmarks in ESP32-based IoT devices. *Journal of Low Power Electronics and Applications*. 2023. Vol. 12(2). P. 1–14.
50. Melnichenko I., Horbach Y. Integration of MQTT and Blynk platforms in remote greenhouse control systems. *Proceedings of the International Conference on Embedded Systems and Applications*. 2023. P. 88–93.

					КВРКІ 210480.21.04.32 ІІЗ	Арк. 67
Зм.	Арк.	№ докум.	Підпис	Дата		

Додаток Б
(обов'язковий)

КОПІЯ КРЕСЛЕННЯ «З'ЄДНАННЯ КОМПОНЕНТІВ НА МАКЕТНІЙ ПЛАТІ»



Додаток Г
(обов'язковий)

КОД ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

```
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
#define OLED_RESET -1
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire,
OLED_RESET);

const int soilMoisturePin = 34;
const int ledGreenPin = 25;
const int ledRedPin = 26;
const int relayPin = 2;

int soilMoistureValue = 0;

void setup() {
  Serial.begin(9600);
  pinMode(soilMoisturePin, INPUT);
  pinMode(ledGreenPin, OUTPUT);
  pinMode(ledRedPin, OUTPUT);
  pinMode(relayPin, OUTPUT);

  if(!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) {
    Serial.println("SSD1306 allocation failed");
    for(;;);
  }
}
```

```

display.clearDisplay();
display.setTextColor(SSD1306_WHITE);
display.setTextSize(1);
display.setCursor(0, 0);
display.println("Smart Irrigation");
display.display();
delay(1500);
}

void loop() {
    soilMoistureValue = analogRead(soilMoisturePin);

    if (soilMoistureValue < 2000) {
        digitalWrite(ledRedPin, LOW);
        digitalWrite(relayPin, HIGH);
        digitalWrite(ledGreenPin, HIGH);
    } else {
        digitalWrite(ledGreenPin, LOW);
        digitalWrite(relayPin, LOW);
        digitalWrite(ledRedPin, HIGH);
    }

    Serial.print("Soil Moisture: ");
    Serial.println(soilMoistureValue);

    display.clearDisplay
display.setCursor(0, 0);
display.print("Soil: ");
display.println(soilMoistureValue);
display.print("Relay: ");
display.println((soilMoistureValue < 2000) ? "ON" : "OFF");
display.display();

    delay(1000);
}

```

Завідувачу кафедри КІС
д-р. філософії, доц. Ольгі ПАВЛЮВІЙ

Натара БАБІСНКА

III^o здобувача вищої освіти

ФГГ, 4 курсу, групи КІ2-21-4

ЗАЯВА

З правилами чинного Положення «Про систему забезпечення академічної доброчесності у Хмельницькому національному університеті» від 01.07.2022, згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомлений(а). Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на плагіат оповіщений(а) та надаю свою згоду на обробку та збереження університетом моєї роботи в інституційному репозитарії університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-технічних засобів (Strike-Plagiarism та Anti-Plagiarism) та використання роботи для виявлення плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення плагіату в текстах робіт.

Робота для перевірки університетом надається в друкованому та електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

 2025 року

РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ

КАФЕДРИ КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Мікропроцесорна система контролю мікроклімату розумного саду

Автор: Назар БАБИШКО

Спеціальність: 123–Комп'ютерна інженерія

Освітня програма: освітньо-професійна

Науковий керівник: Світлана САЧЕНКО, к.е.н., доцент

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи.	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укривтя запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	

Підтвердження:

Запозичення, виявлені в роботі, є законними і не є плагіатом, оскільки:

- 1) запозичення розміщені в розділах аналізу існуючих аналогів та прототипів, які не описують безпосередньо авторське дослідження і не стосуються результатів роботи;
- 2) усі запозичення фрагментарні, або мають належним чином оформленні посилання;
- 3) окремі виявлені збіги є загальноживаними фразами або виразами, про що свідчить посилання системи на збіг з 10-40 джерелами на один фрагмент речення;
- 4) в якості запозичень в окремих місцях системою зафіксовано послідовності чотирьохрозрядних двійкових кодів, які є входними даними до великої кількості задач і не можуть розглядатися як об'єкт авторських прав і, відповідно, їх порушення;
- 5) всі зафіксовані системою ознаки модифікації тексту відносяться до комбінування латинських символів зі українськомовними скороченнями індексів в формулах, що не є модифікацією тексту.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ідентичності/схожості StrikePlagiarism, складає 0.9% і адресується до 13 періоджерел; та системою Anti-Plagiarism складає 1%, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи

Гарант ОП

Завідувач кафедри КІС

Світлана САЧЕНКО

Андрій НіЧЕПОРУК

Ольга ПАВЛОВА

Протокол аналізу звіту подібності експертом

Заявляю, що я ознайомився (-лась) з Повним звітом подібності, який був згенерований Системою виявлення і запобігання плагіату щодо роботи:

Автор: Назар БАБІЄНКО

Співавтор:

Назва: Бабієнко_Мікропроцесорна система контролю мікроклімату розумного саду

Експерт:

Підрозділ: Кафедра комп'ютерної інженерії та інформаційних систем

Коефіцієнт подібності 1:0.9%

Коефіцієнт подібності 2:0.2%

Мікропробіли: 8

Заміна букв: 0

Інтервали: 0

Білі знаки: 0

Дата створення звіту: 2025-06-06 00:20:52.0

Після аналізу Звіту подібності констатую наступне:

Запозичення, виявлені в роботі є законними і не є плагіатом. Рівень подібності не перевищує допустимої межі. Таким чином робота незалежна і приймається.

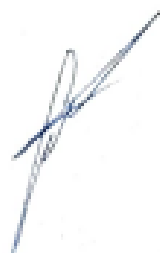
Запозичення не є плагіатом, але перевищено граничне значення рівня подібностей. Таким чином робота повертається на доопрацювання.

Виявлено запозичення і плагіат або навмисні текстові спотворення (маніпуляції), як передбачувані спроби укриття плагіату, які роблять роботу невідповідною вимогам законодавства (Ст. 32. ЗУ Про вищу освіту, пункт 3.1, Ст. 42. ЗУ Про освіту) та вимог НАЗЯВО (Критерій 5), а також кодексу етики і процедур. Таким чином робота не приймається.

Обґрунтування:

2025-06-06

Дата



Доцент Андрій Нічепорук

експерт

Anti-Plagiarism (UA) v-15.281 Educational

The maximum coincidence with one document **0.0%**

Dictionaries check: on US, ru_RU, ua-UA. Errors in the documents: **9%**

ID: 243749 Title: БКР Мікропроцесорна система контролю мікроклімату розумного саду Added in a DB: 2025-06-05 Authors: Назар БАБІЧЕНКО Heads: Світлана САЧЕНКО Consultants: Opponents:	Document		Sum coincidence on the DB	
	Symbols	Lexemes	Symbols	Lexemes
	95261	752	837 (1%)	10 (1%)

Plagiarism sources

ID	Description	Plagiarism presence in the document	
		Symbols	Lexemes

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Дипломник: Бабієнко Назар Віталійович

Тема: Мікропроцесорна система контролю мікроклімату розумного саду

Спеціальність: 123 «Комп'ютерна інженерія»

Обсяг кваліфікаційної роботи:

Кількість листів креслень 3 Кількість сторінок записки 67

1. Метою кваліфікаційної роботи є розробка недорогої та ефективної системи моніторингу й контролю мікроклімату на основі мікроконтролера ESP32. У роботі реалізовано апаратно-програмний комплекс, що дає змогу вимірювати температуру, вологість повітря, вологість ґрунту та рівень освітленості за допомогою цифрових сенсорів (DHT22, BH1750, емнісного сенсора вологості).
2. Робота повністю відповідає завданню, визначеному в технічному завданні, та охоплює всі поставлені етапи: аналіз, проектування, реалізація, тестування.
3. У дипломній роботі розроблено автоматизовану систему моніторингу мікроклімату, призначену для застосування в умовах розумного саду. Система побудована на базі мікроконтролера ESP32 та комплекту цифрових сенсорів: температури й вологості (DHT22), освітленості (BH1750), а також емнісного датчика вологості ґрунту. Зібрані дані обробляються та передаються за допомогою Wi-Fi-з'єднання у вебінтерфейс або мобільний застосунок. Для програмування використано Arduino IDE та мову C++. У програмному забезпеченні реалізовано алгоритми керування, зчитування даних, а також логіку енергозбереження з використанням режиму глибокого сну. Обґрунтовано актуальність обраної теми з позиції потреб малих фермерських господарств у доступних та гнучких рішеннях контролю мікроклімату. Проведено аналіз впливу мікрокліматичних параметрів на ріст сільськогосподарських культур, розглянуто сучасні промислові системи моніторингу, проаналізовано апаратні альтернативи. Запропонована структура системи демонструє хорошу масштабованість, адаптивність до польових умов і здатність до автономної роботи.

Дослідження супроводжується якісно виконаними структурними схемами, логікою програмної реалізації та обґрунтованим вибором компонентної бази. Роботу виконано відповідно до технічного завдання, усі етапи – від теоретичного обґрунтування до практичної реалізації – виконано на належному рівні.

4. Позитивні сторони роботи: вдало поєднано сучасну елементну базу, сенсори, інтернет речей (IoT) та мобільні технології.

5. Негативні сторони роботи: обмеження щодо стабільності з'єднання у польових умовах.

6. Оцінка графічного оформлення та пояснювальної записки роботи: Пояснювальна записка оформлена коректно, згідно діючих стандартів оформлення документації.

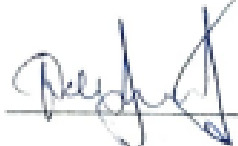
7. Відгук про роботу в цілому: Робота виконана на належному науково-технічному рівні.

8. Інші зауваження: _____

9. Оцінка дипломної роботи: задовільно

Рецензент (прізвище, ім'я, по батькові, посада, місце роботи) Макаришин
Ромо Анатолійович, доцент, к.т.н. кафедри АСУ

"10" 08 2025 р.

 (підпис)