

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра комп'ютерних систем управління

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

**«Комп'ютерно-інтегрований інтелектуальний інтерфейс для управління
мікрофермою»**

Виконав: студент 2 курсу, групи 2АКІТР-23м
спеціальності 174 – Автоматизація,
комп'ютерно-інтегровані технології, та робототех-
ніка

 Ігор БРЕДЕЛСВ

Ім'я ПРІЗВИЩЕ

Керівник: професор, завідувач кафедри КСУ

степінь, звання, посада

 В'ячеслав КОВТУН

Ім'я ПРІЗВИЩЕ

«4» грудня 2024 р.

Опонент: доцент каф. АІПТ

степінь, звання, посада

 Юрій ІВАНОВ

Ім'я ПРІЗВИЩЕ

«4» грудня 2024 р.

Допущено до захисту

Зав. кафедри КСУ

В'ячеслав КОВТУН

«4» грудня 2024 р.

Вінниця – 2024 рік

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра комп'ютерних систем управління
Рівень вищої освіти другий (магістерський)
Галузь знань – 17 – Електроніка, автоматизація та електронні комунікації
Спеціальність – 174 – Автоматизація, комп'ютерно-інтегровані технології та робототехніка
Освітньо-професійна програма – Інтелектуальні комп'ютерні системи

ЗАТВЕРДЖУЮ
Зав. кафедри КСУ

В'ячеслав КОВТУН

“14” жовтня 2024 року

ЗАВДАННЯ
НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ
студенту Бределєву Ігорю Володимировичу

(прізвище, ім'я, по батькові)

- Тема роботи. «Комп'ютерно-інтегрований інтелектуальний інтерфейс для управління мікрофермою»
керівник роботи Ковтун В'ячеслав Васильович
затверджені наказом ВНТУ від “17” вересня 2024 року №310
- Термін подання студентом роботи “1” грудня 2024 року
- Вихідні дані до роботи: об'єкт автоматизації – мікроферма; параметри мікроферми для контролю та регулювання – вологість, температура землі та повітря, освітленість; пристрій керування – мікроконтролерний модуль; датчики – вологості землі та повітря, температури повітря та землі, освітленості, перелік матеріалів для розробки:
 - Автоматизована мікроферма на основі системи Smart Wick з використанням Інтернету речей https://www.researchgate.net/publication/337019517_Automated_Smart_Wick_System-Based_Microfarm_Using_Internet_of_Things.
 - Система мікроферм як критичне втручання в рух альтернативної їжі: https://www.researchgate.net/publication/379724831_First_you_need_the_farmers_The_microfarm_system_as_a_critical_intervention_in_the_alternative_food_movement
 - Co-Design and Experimentation of a Prototype of Agroecological Micro-Farm Meeting the Objectives Set by Climate-Smart Agriculture: https://www.researchgate.net/publication/366971229_Co-Design_and_Experimentation_of_a_Prototype_of_Agroecological_Micro-Farm_Meeting_the_Objectives_Set_by
- Зміст текстової частини: Вступ; Аналіз об'єкта автоматизації і технічне обґрунтування доцільності розробки; Розробка схем автоматизованої системи управління; Розробка та тестування програмного забезпечення; Економічна частина.
- Перелік ілюстративного матеріалу (з точним зазначенням обов'язкових креслень)
 - UML-діаграма діяльності інтелектуальної системи контролю мікроклімату – 1 аркуш;
 - Структурна схема мікроферми – 1 аркуш;
 - Апаратна схема мікроферми – 1 аркуш;
 - Схема програми контролю мікрофермою – 1 аркуш;
 - Схема структурна ієрархічна АСУТП – 1 аркуш.

Розділ	Прізвище, ініціали та посада консул- ьтанта	Підпис, дата	
		завдання видав	виконання прийняв
4	Кавецький В.В – доцент кафедри еконо- міки підприємства і виробничого мене-		

КАЛЕНДАРНИЙ ПЛАН

[illegible]

(Ім'я ПРИЗВИЩЕ)

АНОТАЦІЯ

УДК 621.374.415

Бределєв І.В. Комп'ютерно-інтегрований інтелектуальний інтерфейс для управління мікро-фермою. Магістерська кваліфікаційна робота зі спеціальності 174 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка», освітня про-грама – Інтелектуальні комп'ютерні системи. Вінниця: ВНТУ, 2024. - 125 с.

На укр. мові. Бібліогр.: 29 назв; рис.: 36; табл. 18.

Метою виконання магістерської кваліфікаційної роботи є розробка автоматизованої системи контролю мікроклімату для мікроферм, яка забезпечує ефективне управління параметрами середовища.

У магістерській кваліфікаційній роботі розроблено автоматизовану систему контролю мікроклімату для мікроферм, що забезпечить оптимальні умови вирощування рослин через ефективне управління освітленням, вологістю ґрунту та іншими параметрами.

У роботі проаналізовано основні методи контролю вологості ґрунту та освітлення, порівняно їхні переваги та недоліки. Особливу увагу приділено розробці системи, що базується на мікроконтролері NodeMCU, для моніторингу параметрів мікроклімату й автоматичного керування насосами, освітленням та іншими виконавчими пристроями.

У конструктивній частині розроблено архітектуру системи, обрано необхідні датчики та обладнання, розроблено алгоритми автоматизації з використанням мови програмування Arduino IDE. Також реалізовано взаємодію компонентів системи через хмарні платформи для віддаленого моніторингу й управління.

Проект демонструє економічну доцільність автоматизації завдяки зниженню витрат на ресурси та підвищенню ефективності вирощування рослин.

Ключові слова: автоматизація, IoT, контроль мікроклімату, мікроферма, датчики вологості, температури та освітленості, NodeMCU, Arduino IDE.

ABSTRACT

Bredeliev I.V. Computer-Integrated Intelligent Interface for Micro-Farm Management. Master's Thesis in specialty 174 "Automation, Computer-Integrated Technologies, and Robotics," educational program – Intelligent Computer Systems. Vinnytsia: VNTU, 2024. - 125 p.

In Ukrainian. Bibliography: 29 sources; illustrations: 36; tables: 18.

The purpose of the master's qualification thesis is to develop an automated microclimate control system for microfarms, ensuring effective management of environmental parameters.

The study analyzes the main methods of controlling soil moisture and lighting, comparing their advantages and disadvantages. It considers the use of modern sensors for moisture, temperature, and light, as well as the potential integration of IoT for automating data collection and management. Particular attention is paid to designing a system based on the NodeMCU microcontroller for monitoring microclimate parameters and automatically controlling pumps, lighting, and other actuators.

The design section includes the development of the system architecture, selection of necessary sensors and equipment, and creation of automation algorithms using the Arduino IDE programming language. Interaction between system components is implemented via cloud platforms for remote monitoring and control.

The project demonstrates the economic feasibility of automation by reducing resource costs and increasing the efficiency of plant cultivation.

Keywords: automation, IoT, microclimate control, microfarm, moisture, temperature, and light sensors, NodeMCU, Arduino IDE.

ЗМІСТ

ВСТУП.....	4
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	6
1.1 Причини для розробки мікроферм	6
1.2 Інтернет речей (IoT)	7
1.3 Існуючі мікроферми	9
1.4 Цільове призначення мікроферм	14
1.5 Технології для контролю оптимальних умов у теплицях	16
1.5.1 Аналіз методів контролю температури	16
1.5.2 Аналіз методів контролю вологості ґрунту.....	20
1.5.3 Аналіз методів контролю освітленням	22
1.6 Системи контролю мікроклімату.....	25
1.7 Висновки до розділу.....	27
2 РОЗРОБКА СХЕМ АВТОМАТИЗОВАНОЇ СИСТЕМИ І ВИБІР АПАРАТНИХ ЗАСОБІВ ДЛЯ ЇЇ РЕАЛІЗАЦІЇ	28
2.1 Опис структури управління мікрофермою	28
2.1.1 Визначення вимог	30
2.2 Розробка апаратної схеми мікроферми	31
2.3 Розробка структурної схеми мікроферми	33
2.4 Вибір необхідного обладнання для автоматизації мікроферми	35
2.4.1 Вибір типу керуючого модуля.....	36
2.4.2 Вибір датчиків вологості і температури повітря	39
2.4.3 Вибір датчика температури землі.....	41
2.4.4 Вибір датчика вологості ґрунту	44
2.4.5 Вибір датчика освітлення.....	46
2.4.6 Вибір виконавчих механізмів	48
2.5 Вибір системи виведення інформації	51
2.5 Підсумки другого розділу	53
3 РОЗРОБКА АЛГОРИТМІВ І ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	55
3.1 Обґрунтування вибору програмної платформи.....	55
3.2 Розробка алгоритму роботи системи мікроферми	57
3.3 Розробка програмного забезпечення	58

3.4 Результати тестування системи контролю мікроферми.....	66
3.5 Висновки до розділу.....	72
4 ЕКОНОМІЧНА ЧАСТИНА.....	74
4.2 Розрахунок узагальненого коефіцієнта якості розробки.....	77
4.3 Розрахунок витрат на проведення науково-дослідної роботи.....	79
4.3.1 Витрати на оплату праці.....	79
4.3.2 Відрахування на соціальні заходи.....	82
4.3.3 Сировина та матеріали.....	82
4.3.4 Розрахунок витрат на комплектуючі	84
4.3.5 Спецустаткування для наукових (експериментальних) робіт	85
4.3.6 Програмне забезпечення для наукових (експериментальних) робіт	85
4.3.7 Амортизація обладнання, програмних засобів та приміщень	86
4.3.8 Паливо та енергія для науково-виробничих цілей	87
4.3.9 Службові відрядження.....	88
4.3.10 Витрати на роботи, які виконують сторонні підприємства, установи і організації	88
4.3.11 Інші витрати.....	88
4.3.12 Накладні (загальновиробничі) витрати	88
4.4 Розрахунок економічної ефективності науково-технічної розробки при її можливій комерціалізації потенційним інвестором	89
4.5 Висновки до розділу.....	93
ВИСНОВКИ.....	94
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	95
ДОДАТКИ.....	98
Додаток А (обов'язковий). Протокол перевірки навчальної (кваліфікаційної) роботи	99
Додаток Б (обов'язковий). Технічне завдання	100
Додаток В (вибірковий). Лістинг програми	105
Додаток Г (обов'язковий). Ілюстративна частина	114

ВСТУП

Актуальність теми. Зростання глобальної потреби у продовольстві та необхідність сталого використання ресурсів роблять питання автоматизації управління мікрокліматом в мікрофермах надзвичайно актуальним. Сучасні виклики, зумовлені змінами клімату, обмеженням доступу до ресурсів та зростанням населення, вимагають впровадження технологічних рішень, що забезпечують ефективне вирощування сільськогосподарської продукції з мінімальним впливом на довкілля. Використання автоматизованих систем управління мікрофермами дозволяє значно підвищити продуктивність навіть на невеликих площах.

Сьогодні такі системи можуть інтегрувати датчики вологості, температури та освітленості для моніторингу параметрів середовища, а також засоби автоматизації, які забезпечують контроль насосів, освітлення й інших пристроїв. Особливо важливим є застосування IoT-технологій, які надають можливість віддаленого управління та аналізу даних через хмарні платформи, що спрощує процеси вирощування навіть для початківців.

Автоматизація також сприяє зменшенню витрат води, добрив та енергії, роблячи вирощування екологічно чистим і стійким. Таким чином, розробка автоматизованої системи контролю мікроклімату є актуальним напрямком для підвищення ефективності сільськогосподарського виробництва.

Мета роботи полягає в розробці автоматизованої системи контролю мікроклімату для мікроферм, яка забезпечує ефективне управління параметрами середовища (освітлення, вологість ґрунту, температура) для створення оптимальних умов вирощування рослин.

Завдання роботи:

- Провести аналіз методів контролю вологості ґрунту, температури та освітлення, порівняти їхні переваги й недоліки.
- Розробити архітектуру автоматизованої системи мікроклімату, вибрати необхідне обладнання й датчики.

- Створити алгоритми та програмне забезпечення автоматизованого управління на базі NodeMCU з використанням Arduino IDE.
- Забезпечити інтеграцію системи з хмарними платформами для віддаленого моніторингу та управління.
- Оцінити економічну доцільність автоматизації.

Об'єктом дослідження є процеси автоматизованого управління мікрокліматом у мікрофермах.

Предметом дослідження є апаратно-програмний комплекс для автоматизації моніторингу та управління параметрами мікроклімату, включаючи освітлення, вологість ґрунту та температуру.

Іновація роботи полягає в запровадженні розподіленої системи датчиків та використання IoT для автоматизації збору даних і управління.

Практична цінність результатів роботи дозволяють створити доступну для впровадження автоматизовану систему вирощування рослин у мікрофермах, яка може використовуватися як у малих, так і у великих фермерських господарствах.

Особистий внесок полягає в розробці структури системи, вибір обладнання, створення алгоритмів автоматизації та програмного забезпечення для контролю параметрів мікроклімату.

Апробація наукової роботи: результати досліджень апробовано на LIII Всеукраїнській науково-технічній конференції факультету інтелектуальних інформаційних технологій та автоматизації (2024) [27].

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Причини для розробки мікроферм

Одним із основних принципів ідеї «міської стійкості» є місцеве виробництво їжі. Початок вирощування фруктів, овочів і трав у містах оптимізує попит і пропозицію, зменшить шкідливі викиди та транспортні витрати. Певною мірою це дає можливість підвищити продовольчу безпеку міст.

Проблема в тому, що міське землеробство не завжди є ефективним. Земля в межах міста коштує дорого, і навіть у занедбаних містах перспектива перетворення на сільськогосподарські угіддя може спричинити додаткові витрати, особливо якщо ці міста згодом будуть відновлені.

З точки зору міського сільського господарства, мікроферми здаються найбільш життєздатними. Тип «малого фермерського господарства, яке можна організувати на площі менше 2 га в міській або приміській зоні». На мікрофермах використовується невелике автоматизоване модульне обладнання для вирощування харчових продуктів, яке можна встановити практично в будь-якому будинку чи квартирі[4].

Однією з ключових частин роботи мікроферми є автоматизація. Система автоматично контролює процеси поливу, вентиляції та освітлення за допомогою датчиків, які контролюють умови навколишнього середовища, такі як вологість, температура та освітлення. Це дозволяє мінімізувати втрати і зробити процес ефективним і надійним.

Штучний інтелект також відіграє ключову роль в аналізі даних від датчиків і прогнозуванні оптимальних умов для вирощування сільськогосподарських культур. Це допомагає керувати ресурсами та налаштовувати систему для досягнення максимальних результатів із мінімальними втратами.

Інтелектуальний інтерфейс виступає як міст між людиною та системою управління, використовуючи алгоритми для оптимізації процесів. Наприклад, система може аналізувати дані з датчиків для прогнозування майбутніх умов та надавати рекомендації для покращення врожайності. Інтерфейси, що базуються на AI, можуть

автоматично адаптуватися до умов мікроферми та навіть приймати рішення без втручання людини, забезпечуючи автоматичне навчання.

Мікроферми, які є невеликими автономними сільськогосподарськими одиницями, використовують ці системи для ефективного вирощування культур у контрольованих умовах. Це можуть бути системи гідропоніки, де рослини ростуть без ґрунту, отримуючи необхідні поживні речовини через воду, або аеропоніки, де корені рослин отримують поживні речовини через туман. Важливим аспектом є вертикальне фермерство, яке дозволяє вирощувати рослини в стелажах для максимальної економії простору.

Інтеграція з технологіями IoT дозволяє підключати різноманітні датчики до системи, що надає можливість віддаленого контролю за допомогою смартфона або комп'ютера. Такі датчики в режимі реального часу збирають інформацію про стан рослин, ґрунту, води і повітря, забезпечуючи можливість миттєвого реагування на зміни[5].

1.2 Інтернет речей (IoT)

Інтернет речей (IoT) — це система взаємопов'язаних пристроїв, які мають у собі датчики, програмне забезпечення та інші технології, що дозволяють обмінюватися даними через інтернет або інші мережі. Завдяки IoT пристрої можуть автоматично взаємодіяти один з одним без участі людини, що створює нові можливості для автоматизації та оптимізації різних процесів. Це є особливо важливим у сільському господарстві та управлінні мікрофермами[6].

Щоб зрозуміти, що таке Інтернет речей і з чого він складається, корисно мати огляд основних компонентів, які відіграють ключову роль у його функціонуванні:

- Пристрої - Датчики, які збирають дані з навколишнього середовища: Температура, вологість, світло, рух і пристрої, які виконують дії на основі отриманих даних: Вмикають світло або регулюють температуру.
- Комунікації – мережеві протоколи та інші технології для обміну інформацією між основними компонентами та центральними системами.

- Обробка інформації – Хмарні платформи, які дозволяють зберігати та аналізувати дані в реальному часі, а також аналітичні інструменти для ідентифікації та прогнозування шаблонів.
- Інтерфейс – для користувачів IoT Інтернету речей він надає мобільні та веб-додатки, які надають доступ до даних, керування пристроями, налаштування параметрів та програмні інтерфейси для взаємодії між різними системами та програмами.
- Безпека – шифрування інформації для захисту від несанкціонованого доступу та механізми для обмеження неавторизованих користувачів.

Головний принцип — забезпечити комфортну взаємодію між пристроями, а також між людьми та пристроями(Рис. 1.1)[28].



Рисунок 1.1 — Принцип IoT

У спрощеному вигляді ця схема виглядає так: датчики та пристрої записують критичні параметри та надсилають їх далі по ланцюжку до центральної системи, де

аналітичні інструменти та програми формують дані, щоб надати розуміння поточних подій[7].

Переваги IoT:

1. Підвищена врожайність – безперервний моніторинг усіх показників надає фермерам точну інформацію про стан їхніх посівів і дозволяє вчасно вжити заходів для оптимального росту та розвитку їх культур.
2. Оптимізація використання ресурсів - Раціональний розподіл води, добрив та інших компонентів дозволяє уникнути надлишків і надмірних витрат, тим самим зменшуючи такі витрати, як гроші, час і зусилля.
3. Покращення якості продукції (створення оптимальних умов для росту рослин) дозволяє значно підвищити врожайність.

1.3 Існуючі мікроферми

Міні-теплиці відрізняються за розміром, конструкцією, матеріалами та призначенням. Найменші варіанти можуть використовуватися в домашніх умовах або на балконі. Переносні моделі зручні тим, що з настанням тепла їх легко можна перевезти на дачу. На ділянці такі теплиці можна пересувати, розміщуючи їх там, де це необхідно, наприклад, для захисту невеликих посадок від негоди або для акліматизації розсади, висадженої у відкритий ґрунт. Ці ферми здатні ефективно використовувати простір, воду, енергію та інші ресурси, і їх часто використовують у міських або домашніх умовах. Ось кілька прикладів існуючих видів мікроферм та їх особливостей:

1. Вертикальні ферми

Вертикальні ферми використовують багатоярусні стелажі для вирощування рослин, що дозволяє оптимізувати простір у міських умовах. Ці ферми зазвичай розташовані у приміщеннях, де можна точно контролювати освітлення, температуру, рівень вологості та поживні речовини. Використовуючи системи гідропоніки чи аеропоніки, вертикальні ферми не потребують великих площ ґрунту і здатні значно підвищити врожайність(рис. 1.2).



Рисунок 1.2 — AeroFarms

AeroFarms (США) — одна з найвідоміших вертикальних ферм, яка використовує аеропоніку для вирощування листових овочів без ґрунту. Їх технологія дозволяє заощадити до 95% води порівняно з традиційним сільським господарством, а врожайність збільшується у кілька разів через щільне розташування рослин у вертикальних стелажах[22].

2. Гідропонічні мікроферми

Гідропоніка — це метод вирощування рослин без ґрунту, де коріння рослин знаходяться у водному розчині, багатому на поживні речовини. Така система ідеально підходить для вирощування на невеликих просторах, наприклад, на балконах, дахах будівель або в спеціальних контейнерах. Гідропонічні мікроферми потребують менше води, ніж традиційні методи вирощування, і забезпечують високий контроль за умовами росту(рис. 1.3).



Рисунок 1.3 — Freight Farms

Freight Farms (США) пропонує повністю автоматизовані гідропонічні системи, розміщені в транспортних контейнерах. Це дозволяє вирощувати культури практично в будь-якому місці — на дахах будівель, у міських районах або навіть у пустелях[23].

3. Аеропонічні мікроферми

Аеропоніка є ще одним типом безґрунтового вирощування, де коріння рослин знаходяться у повітрі і зрошуються туманом, що містить поживні речовини. Цей метод дозволяє ще більше економити воду та збільшувати швидкість росту рослин за рахунок постійного доступу до кисню(рис. 1.4).



Рисунок 1.4 — Tower Garden

Tower Garden — це популярна аеропонічна мікроферма для домашнього використання. Вона має вертикальну структуру, що дозволяє вирощувати до 20 рослин на одному невеликому просторі. Завдяки своїй простоті та ефективності ця система здобула популярність серед садівників-аматорів та малих підприємств[24].

4. Домашні мікроферми

Це компактні системи, призначені для вирощування овочів та трав на невеликих ділянках у домашніх умовах — наприклад, на кухні або балконі. Такі ферми часто використовують прості гідропонічні або аеропонічні технології і можуть бути повністю автоматизованими для зручності користувачів(рис. 1.5).



Рисунок 1.5 — Click and Grow

Click and Grow — це розумний сад для вирощування трав і маленьких овочів у домашніх умовах. Ця система працює на основі гідропоніки та використовує спеціальні «смарт-грунти» з необхідними поживними речовинами. Користувачі просто додають воду, а система сама регулює рівень освітлення та живлення[25].

5. Аквапонічні ферми

Аквапоніка поєднує вирощування риб та рослин у симбіотичній системі. Риби виділяють відходи, які слугують добривом для рослин, а рослини, в свою чергу, фільтрують воду, забезпечуючи чисте середовище для риб. Це повністю екологічна система, яка не потребує синтетичних добрив і використовує мінімум води.



Рисунок 1.6 — Back to the Roots

Back to the Roots пропонує маленькі аквапонічні ферми для домашнього використання, де можна одночасно вирощувати зелень та рибу. Це дозволяє користувачам легко спостерігати за природним колообігом і отримувати корисну зелень і водночас доглядати за акваріумними рибками[26].

6. Контейнерні ферми

Контейнерні ферми — це мобільні та автономні ферми, які використовують транспортні контейнери для вирощування рослин у контрольованих умовах. Такі ферми зазвичай обладнані системами гідропоніки або аеропоніки та повністю автоматизовані. Контейнерні ферми можуть бути встановлені практично в будь-якому місці та забезпечувати свіжі продукти навіть у регіонах з важкими кліматичними умовами(рис. 1.7).



Рисунок 1.7 — Growtainer

Growtainer створює модульні системи, розміщені у контейнерах, які можна використовувати як для комерційних, так і для особистих потреб. Завдяки контролю за мікрокліматом вони дозволяють вирощувати культури незалежно від зовнішніх умов[27].

7. Розумні ферми зі штучним інтелектом

Такі мікроферми використовують системи автоматизації та штучного інтелекту для повного контролю за процесами вирощування. Вони можуть самостійно регулювати рівень світла, температури та вологості на основі зібраних даних. Інтелектуальні ферми забезпечують максимально ефективне використання ресурсів і можуть адаптуватися до змін зовнішніх умов(рис. 1.8).



Рисунок 1.8 — Square Roots

Square Roots — це мережа контейнерних ферм, що використовують технології штучного інтелекту для оптимізації вирощування рослин. Вони поєднують вертикальне фермерство з автоматизованими рішеннями, що дозволяє отримувати високі врожаї в невеликих міських просторах[28].

1.4 Цільове призначення мікроферм

Мікроферми мають велике значення у сучасному світі, адже вони дозволяють максимально ефективно використовувати обмежений простір для вирощування рослин, грибів або навіть для розведення риби в аквапонічних системах. Їхнє цільове призначення залежить від багатьох факторів: від особистих потреб до глобальних викликів у сфері сільського господарства, продовольчої безпеки та екології.

Для особистого використання мікроферми стають незамінним інструментом для вирощування свіжих продуктів прямо вдома. Сучасна тенденція до здорового способу життя спонукає людей віддавати перевагу органічним і свіжим продуктам. Мікроферма дозволяє забезпечити себе зеленню, овочами чи навіть ягодами без використання хімічних добрив чи пестицидів. До того ж це стає можливим незалежно від пори року чи кліматичних умов. Для мешканців міст, де бракує місця для традиційного садівництва, це просто знахідка, яка поєднує ефективність і зручність.

Окрім практичної користі, мікроферми виконують роль навчальних платформ. Вони можуть стати основою для вивчення агрономії, біології, хімії або навіть робототехніки. Наприклад, люди, які цікавляться автоматизацією або електронікою, можуть створювати системи керування мікрофермою, що регулюють полив, освітлення чи температуру. Це сприяє розвитку креативності й технічних навичок, перетворюючи ферму на експериментальний простір.

Для підприємців мікроферми мають значний потенціал як інструмент комерційної діяльності. У світі, де попит на екологічно чисті продукти зростає, вони можуть використовуватися для створення невеликих бізнесів. Наприклад, вирощування зелені для ресторанів або продаж мікрозелені, яка останнім часом стала дуже популярною через свої корисні властивості. Уявіть собі невеличкий бізнес, де свіжа продукція доставляється прямо до споживача, зберігаючи свою свіжість і якість завдяки локальному вирощуванню.

Ще одна важлива сфера застосування мікроферм — це вирішення глобальних проблем, зокрема нестачі продовольства або дефіциту ресурсів. У районах, де ґрунт малопродатний для сільського господарства, або в місцях з обмеженим доступом до води, мікроферми дозволяють вирощувати їжу з мінімальними витратами. Технології гідропоніки чи аквапоніки, які використовуються в мікрофермах, споживають значно менше води, ніж традиційне землеробство, а компактність дозволяє встановлювати такі ферми навіть у густонаселених міських районах чи вертикальних спорудах.

Мікроферми також мають значення для людей, які прагнуть зменшити свій екологічний слід. Вирощуючи продукти локально, можна значно скоротити викиди вуг-

лецю, пов'язані з транспортуванням продуктів. До того ж використання автоматизованих систем дозволяє уникати перевитрати ресурсів, таких як вода чи електроенергія.

1.5 Технології для контролю оптимальних умов у теплицях

Дотримання заданих кліматичних параметрів є важливою частиною успішного функціонування системи контролю мікроклімату. Підбір оптимальних і наближених до ідеальних умов вирощування культур також є важливою частиною цієї роботи, оскільки базується на мікрокліматі.

Основними завданнями контролю є:

- Контроль температури.
- Керування вологістю ґрунту.
- Керування системами освітлення.

Раніше автоматизація мікроферм була дорогим і часом нерентабельним процесом, але поки що вирішення цієї проблеми не дуже дороге, і виграш, швидше за все, принесе ще більше. Мікроконтролери - це програмовані мікросхеми, які можуть керувати різноманітними електронними пристроями. Мікроконтролер містить одне або кілька ядер процесора, пам'ять і програмовані периферійні пристрої вводу/виводу.

1.5.1 Аналіз методів контролю температури

Температура — це фізична величина, яка характеризує ступінь нагрятості тіла, середовища або системи. Вона визначає, наскільки швидко рухаються частинки речовини: чим інтенсивніший рух атомів і молекул, тим вища температура. Це поняття є центральним у термодинаміці, адже саме через температуру описують тепловий стан системи.

Фізично температура пов'язана з енергією. Чим вища температура, тим більше енергії має система. Абсолютний нуль — це найнижча теоретично можлива температура, за якої рух частинок практично припиняється. Його значення становить $-273,15^{\circ}\text{C}$ або 0 Кельвінів.

Для вимірювання температури використовують різні шкали. Найпоширенішою є шкала Цельсія, де 0 градусів відповідає точці замерзання води, а 100 — точці кипіння. В науці частіше використовують шкалу Кельвіна, яка починається з абсолютного нуля, що полегшує розрахунки.

У мікрофермі контроль температури є критично важливим для забезпечення оптимальних умов росту рослин чи утримання тварин. Для цього застосовуються різні методи, які можна розділити на такі категорії:

1. Пасивні методи.

- Пасивні методи контролю температури базуються на природних фізичних явищах і не потребують активного енергоспоживання. Одним із основних способів є теплоізоляція, яка допомагає зменшити втрати тепла взимку і перегрів улітку. Для цього використовують матеріали, що добре утримують тепло, такі як пінополістирол, мінеральна вата або полікарбонат. Особливо ефективними є прозорі ізоляційні матеріали для теплиць, які пропускають світло, але утримують тепло всередині.
- Теплова маса також відіграє важливу роль. Це матеріали, що накопичують тепло вдень і поступово віддають його вночі. Наприклад, ємності з водою чи каміння можуть стабілізувати температурні коливання, поглинаючи зайве тепло вдень і підтримуючи тепло вночі.
- Природна вентиляція дозволяє охолоджувати приміщення за рахунок потоків повітря. Відкриття верхніх і нижніх вентиляційних отворів сприяє створенню природної тяги, завдяки якій гаряче повітря виходить назовні, а холодне входить всередину. Це забезпечує циркуляцію повітря без використання вентиляторів.

- Ще одним важливим методом є регулювання потрапляння сонячного світла. Для цього використовують штори, жалюзі чи тіньові сітки, які знижують нагрівання приміщення у спекотні дні. Розташування будівель також має значення: правильна орієнтація може зменшити вплив надмірного сонячного світла або сильних вітрів.
- Ці методи є екологічними, енергоефективними та простими у застосуванні. Вони дозволяють підтримувати комфортні умови навіть у системах, де важлива економія енергії або її автономне використання.

2. Активні методи.

- Активні методи контролю температури передбачають використання пристроїв, які працюють на електроенергії для точного регулювання теплового режиму. Ці методи дозволяють оперативно підтримувати потрібну температуру незалежно від зовнішніх умов, що особливо важливо у середовищах із чутливими процесами, як-от вирощування рослин чи утримання тварин.
- Для обігріву часто використовують теплові пристрої, наприклад, конвектори, тепловентилятори чи інфрачервоні обігрівачі. Інфрачервоні обігрівачі є ефективними, оскільки нагрівають не повітря, а поверхні та об'єкти, що створює локальний обігрів у потрібних зонах. Також поширеними є теплові плівки чи мати, які забезпечують підігрів ґрунту для стимуляції росту рослин.
- Для охолодження використовують вентилятори або системи кондиціонування повітря. У приміщеннях, де важливе енергозбереження, можуть застосовувати системи випаровувального охолодження, які працюють за рахунок зрошення поверхонь водою. Це дозволяє знижувати температуру без значного використання енергії.
- Ключовим елементом активних методів є автоматизація. Завдяки датчикам, система може безперервно відслідковувати температуру та передавати ці дані на контролери, такі як NodeMCU або Arduino. Конт-

ролери аналізують ці дані та вмикають чи вимикають обігрівачі, вентилятори або інші пристрої через реле. Такий підхід мінімізує людський фактор і забезпечує точне дотримання заданих параметрів.

- Активні методи дозволяють створювати стабільний мікроклімат навіть у несприятливих умовах. Вони ідеально підходять для систем, де потрібно підтримувати високу точність температурного режиму, наприклад, у теплицях, фермах чи лабораторіях.

У нашому проєкті ми будемо використовувати активний метод контролю температури з підключенням датчиків DHT22 та DS18B20. Цей вибір базується на їхніх характеристиках і здатності забезпечувати точний моніторинг умов у мікрофермі, що критично важливо для автоматизації та оптимізації мікроклімату.

DHT22 є надійним датчиком для вимірювання температури та вологості повітря. Його перевага в тому, що він поєднує два показники, які часто взаємопов'язані у кліматичному контролі. Завдяки своїй точності ($\pm 0,5$ °C для температури та ± 2 –5% для вологості) він дозволяє створювати ефективну систему регулювання мікроклімату, наприклад, вмикати обігрівачі або системи зволоження при відхиленні від заданих параметрів.

DS18B20 ми обрали через його вузьку спеціалізацію на вимірюванні температури з високою точністю ($\pm 0,5$ °C) і стабільністю роботи навіть у складних умовах. Цей датчик ідеально підходить для моніторингу температури ґрунту чи інших точкових зон, оскільки може працювати в широкому діапазоні температур (-55 °C до $+125$ °C). Його зручність у підключенні до NodeMCU через протокол 1-Wire дозволяє легко інтегрувати кілька таких датчиків до системи.

Активний метод із використанням цих датчиків дозволяє автоматизувати управління температурою. Контролер NodeMCU отримуватиме дані від датчиків і за допомогою реле активуватиме обігрівачі чи охолоджувальні пристрої залежно від потреб. Такий підхід забезпечує стабільність, точність і оперативність, що важливо для підтримання оптимальних умов у мікрофермі.

1.5.2 Аналіз методів контролю вологості ґрунту

Контроль вологості ґрунту є важливим аспектом управління мікрокліматом у мікрофермах, що забезпечує оптимальні умови для росту рослин. Сучасні технології пропонують кілька підходів до вимірювання вологості, кожен із яких має свої особливості. Найпоширенішим методом є використання датчиків, які дозволяють отримувати точні дані в реальному часі. До них належать резистивні, ємнісні та цифрові моделі. Резистивні датчики працюють на основі зміни опору вологого ґрунту, але їхній термін служби обмежений через корозію. Ємнісні датчики є більш довговічними, оскільки вони не піддаються корозії, а цифрові датчики, окрім вимірювання вологості, можуть враховувати температурну компенсацію, що підвищує точність.

Електропровідні датчики вимірюють рівень вологості ґрунту, використовуючи зміну електричного опору між двома електродами, які вставляються в ґрунт. Коли ґрунт насичується водою, його електропровідність зростає, і опір зменшується, що дозволяє визначити рівень вологості. Ці датчики є простими в конструкції та дешевими, що робить їх доступними для широкого використання, особливо серед початківців. Однак вони мають значні обмеження. Основною проблемою є корозія електродів через постійний контакт із вологою, що знижує довговічність датчика. Крім того, їхня точність залежить від сольового складу ґрунту: якщо у ґрунті є багато солей, електропровідність змінюється незалежно від вологості, спотворюючи результати.

Ємнісні датчики вологості працюють за іншим принципом. Вони вимірюють зміну ємності між двома пластинами сенсора, яка залежить від вмісту води в ґрунті. Вода має високу діелектричну проникність, тому її присутність значно впливає на ємність. На відміну від електропровідних датчиків, ємнісні не мають безпосереднього контакту з ґрунтом, що виключає проблему корозії та забезпечує більшу довговічність. Вони також менш залежні від сольового складу, тому їх показники є більш стабільними та точними. Проте ємнісні сенсори коштують дорожче, що може бути обмеженням для невеликих або аматорських проєктів.

Термоелектричні датчики вологості ґрунту використовують залежність температурних властивостей вологого ґрунту від його вологості. Коли ґрунт насичений водою, його теплоємність і теплопровідність змінюються, і ці параметри фіксуються сенсором. Цей метод менш популярний, оскільки він дуже чутливий до зовнішніх факторів, таких як температура навколишнього середовища або зміни кліматичних умов. Термоелектричні сенсори мають певну специфічну нішу, але для широкого використання, особливо в автоматизованих системах, вони поступаються іншим методам.

Порівнюючи ці три типи датчиків, можна сказати, що електропровідні є найдоступнішими, але водночас найбільш вразливими до зовнішніх впливів і мають обмежену довговічність. Ємнісні датчики пропонують значно вищу точність і надійність, але їхня вартість робить їх менш привабливими для проєктів із обмеженим бюджетом. Термоелектричні сенсори, хоча й базуються на цікавому принципі, потребують ретельного врахування зовнішніх умов, що обмежує їх використання у стандартних системах. З огляду на ці аспекти, для довготривалого та надійного контролю в автоматизованих системах найкращим вибором будуть ємнісні сенсори, тоді як електропровідні залишаються хорошим варіантом для простих і недорогих рішень.

Альтернативним підходом є візуальна оцінка вологості, яка базується на зовнішньому вигляді ґрунту, однак цей метод є суб'єктивним і не підходить для автоматизації. Ще один метод – ваговий аналіз, при якому ґрунт зважується до і після висушування. Він забезпечує високу точність, але не підходить для постійного моніторингу.

Інтеграція датчиків у системи Інтернету речей (IoT) відкриває нові можливості для автоматизації. Дані про вологість можна передавати на хмарні сервери, де вони обробляються, дозволяючи дистанційно керувати поливом і зберігати історію показників. Такий підхід значно спрощує управління мікрокліматом і підвищує ефективність вирощування рослин. У результаті найкращим рішенням для мікроферм є використання сучасних ємнісних або цифрових датчиків у поєднанні з IoT-технологіями. Це дозволяє автоматизувати процеси, знизити витрати та забезпечити стабільний розвиток рослин.

Використання IoT-систем (Інтернету речей) для моніторингу та управління вологістю ґрунту є сучасним і ефективним підходом у сільському господарстві, зокрема в мікрофермах. Цей підхід базується на підключенні датчиків і виконавчих пристроїв до мережі Інтернет для збору, передачі та аналізу даних у режимі реального часу.

Система складається з кількох основних компонентів. Першим елементом є датчики, які вимірюють рівень вологості ґрунту. Дані з цих пристроїв передаються на мікроконтролери, які обробляють отриману інформацію. Завдяки підключенню до Wi-Fi мікроконтролери надсилають ці дані на сервери або хмарні платформи. На цьому етапі інформація може зберігатися, аналізуватися та візуалізуватися для користувача через спеціальні додатки чи веб-інтерфейси.

IoT-системи дозволяють не лише спостерігати за станом ґрунту, але й керувати процесами поливу. Наприклад, якщо рівень вологості падає нижче встановленого порогу, система автоматично активує насос чи клапани, забезпечуючи зрошення. Це дає змогу уникати як пересихання, так і переволоження ґрунту, оптимізуючи витрати води.

Додатковою перевагою IoT є можливість аналізу даних. Це дозволяє прогнозувати споживання води, оцінювати ефективність системи поливу та виявляти проблеми в роботі обладнання. За допомогою мобільних додатків чи комп'ютерних платформ користувач може отримувати повідомлення про зміну умов у реальному часі, незалежно від свого місця перебування.

Хоча впровадження IoT-систем потребує початкових інвестицій у обладнання та налаштування, переваги, такі як автоматизація, економія ресурсів і підвищення ефективності, компенсують ці витрати. У результаті IoT стає важливим інструментом для управління мікрокліматом у сучасних мікрофермах.

1.5.3 Аналіз методів контролю освітленням

Контроль освітлення є ключовим аспектом у підтримці оптимального мікроклімату для рослин, оскільки світло безпосередньо впливає на процес фотосинтезу, ріст

і розвиток культур. Основні методи керування освітленням включають ручне керування, використання таймерів, сенсорів освітленості та регуляторів інтенсивності, кожен із яких має свої особливості, переваги та недоліки.

Ручне керування освітленням є найпростішим підходом, коли людина самостійно вмикає та вимикає лампи в залежності від потреб рослин. Такий метод може бути ефективним для невеликих проєктів або домашніх мікроферм, де контроль здійснюється за фіксованим графіком. Проте цей спосіб залежить від людського фактора і не здатен враховувати природні зміни освітлення, що може призводити до недостатнього або надмірного освітлення.

Більш прогресивним є використання таймерів, які дозволяють автоматизувати процес увімкнення та вимкнення ламп. Таймери забезпечують стабільний фотоперіод, необхідний для багатьох культур, і позбавляють від необхідності постійного втручання. Водночас цей метод є статичним: таймери не враховують зміни природного світла протягом дня, через що можливе перевитрачання електроенергії або недостатнє освітлення у похмурі дні.

Найбільш сучасним і точним методом є використання сенсорів освітленості, які вимірюють рівень світла в реальному часі. Сенсори освітленості – це пристрої, що вимірюють рівень світла в оточенні, зазвичай у люксах, і є важливими компонентами автоматизованих систем керування освітленням. Вони застосовуються в сільському господарстві, розумних будинках, промисловості та багатьох інших сферах, де потрібно контролювати освітлення залежно від зовнішніх умов.

Принцип роботи сенсорів освітленості базується на здатності певних матеріалів реагувати на кількість світла, яке на них потрапляє. Більшість сучасних сенсорів використовують фотодіоди або фототранзистори, що генерують електричний сигнал, пропорційний інтенсивності світла. Деякі сенсори мають спеціальні фільтри для точного вимірювання спектра видимого світла, подібного до сприйняття людського ока.

Одним із популярних прикладів сенсорів є модуль BH1750, який широко використовується в проєктах на базі Arduino або NodeMCU. Цей сенсор має високу точність, оскільки він вимірює освітленість у люксах без необхідності складного калібрування. BH1750 може працювати в різних діапазонах чутливості, що робить його

універсальним для застосування в приміщеннях із слабким освітленням або на відкритому просторі. Сенсор підключається через протокол I²C, що забезпечує просте підключення та передачу даних.

Інший приклад – модуль GY-302, який також базується на BH1750. Цей сенсор популярний у проєктах DIY завдяки своїй компактності, низькому енергоспоживанню та доступній ціні. Як і BH1750, він може вимірювати широкий діапазон освітленості – від кількох люксів до десятків тисяч, що дозволяє його використовувати для точного моніторингу світлових умов.

Головними перевагами сенсорів освітленості є:

- Точність і стабільність: Вони забезпечують надійне вимірювання рівня освітлення, навіть у складних умовах.
- Енергоефективність: Сучасні сенсори споживають дуже мало енергії, що дозволяє використовувати їх у системах із живленням від батарей.
- Простота інтеграції: Завдяки стандартним протоколам, таким як I²C, ці сенсори легко інтегруються в різноманітні системи.

Проте сенсори освітленості мають і певні обмеження. Наприклад, їх чутливість може змінюватися під впливом зовнішніх факторів, таких як пил або вологість. Також деякі моделі менш точні у вимірюванні спектральних характеристик світла, що може бути важливим у сільськогосподарських системах, де враховується не лише інтенсивність, а й спектральний склад світла.

Регуляція інтенсивності світла за допомогою димерів є ще одним сучасним рішенням. Димери дозволяють адаптувати освітлення під конкретні потреби, наприклад, збільшуючи його інтенсивність у період активного росту або зменшуючи для економії енергії. У поєднанні з сенсорами освітленості та мікроконтролерами такі системи створюють найбільш адаптивне середовище, яке відповідає потребам рослин і мінімізує енергетичні витрати.

Інтеграція систем освітлення в IoT дозволяє ще більше автоматизувати процеси. За допомогою хмарних платформ можна дистанційно керувати освітленням, аналізувати історичні дані про інтенсивність світла та адаптувати налаштування системи відповідно до потреб конкретних культур. Крім того, можна використовувати комбінації

датчиків освітленості та таймерів, щоб створити більш комплексну систему, яка враховує як природне світло, так і встановлені графіки.

1.6 Системи контролю мікроклімату

Автоматизовані системи контролю мікроклімату є потужними інструментами для забезпечення оптимальних умов вирощування рослин у теплицях та мікрофермах. Вони дозволяють автоматизувати моніторинг та регулювання параметрів середовища, таких як температура, вологість, освітлення, що сприяє підвищенню ефективності та продуктивності сільського господарства.

Ключовими елементами цих систем є сенсори та датчики, які відповідають за вимірювання різних параметрів мікроклімату. Температурні датчики контролюють рівень тепла, гігromетри вимірюють вологість повітря, а датчики освітленості слідкують за інтенсивністю світла[11].

Контролери отримують дані з сенсорів і регулюють параметри мікроклімату відповідно до заданих значень. Вони можуть бути програмованими або базуватися на простих алгоритмах управління, з можливістю відправлення повідомлень про несправності або аномалії(рис. 1.11).

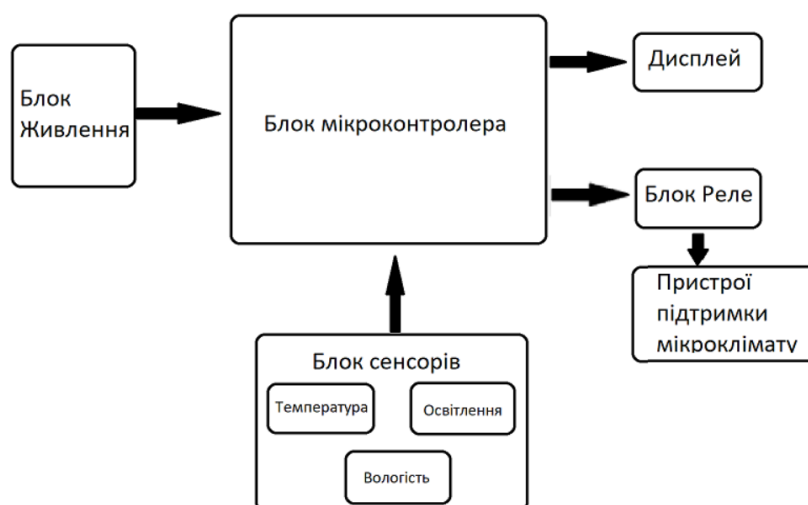


Рисунок 1.11 — Структурна схема контролю мікрокліматом

На поданій схемі представлено структурну організацію системи контролю мікроклімату. Розглянемо кожен із її компонентів і їхні функції:

1. Блок живлення забезпечує енергопостачання всієї системи. Він відповідає за стабільну подачу електроенергії на блок мікроконтролера, сенсори та реле. Надійне джерело живлення є критично важливим для безперебійної роботи системи.
2. Блок мікроконтролера є центральним елементом системи. Його функція — приймати дані від сенсорів, обробляти їх і приймати рішення щодо керування пристроями підтримки мікроклімату. Це "мозок" системи, який виконує алгоритми автоматизації.
3. Блок сенсорів включає датчики, що збирають інформацію про стан навколишнього середовища. У цій схемі передбачені три основні типи сенсорів:
 - Температури, що контролюють рівень тепла в середовищі.
 - Вологості, які оцінюють рівень води у ґрунті або повітрі.
 - Освітленості, що фіксують рівень світла для підтримання оптимальних умов.
4. Дисплей використовується для візуалізації даних. Він може показувати точні показники з сенсорів, стан системи або інші важливі параметри. Це зручний спосіб відображення інформації для користувача.
5. Блок реле виконує функцію перемикача. На основі команд від мікроконтролера реле вмикає або вимикає підключені пристрої підтримки мікроклімату. Це можуть бути лампи освітлення, системи поливу, вентиляції тощо.
6. Пристрої підтримки мікроклімату є виконавчими механізмами, які безпосередньо змінюють умови навколишнього середовища. Наприклад, система поливу зволожує ґрунт, лампи забезпечують додаткове освітлення, а вентилятори знижують температуру або підвищують рівень циркуляції повітря.
7. Програмне забезпечення для моніторингу та управління системою забезпечує інтерфейс для налаштування, перегляду даних у реальному часі та аналізу історичних даних.

Автоматизовані системи контролю мають багато переваг, таких як точність регулювання параметрів, економія ресурсів, зменшення трудозатрат та можливість моніторингу в реальному часі. Вони також забезпечують стабільність умов, що знижує ризик стресу для рослин і підвищує їхню стійкість до хвороб і шкідників.

Однак ці системи мають і деякі недоліки, зокрема високу вартість впровадження, складність налаштування та обслуговування, а також залежність від технологій, що може призвести до негативних наслідків у разі збою. Незважаючи на ці виклики, автоматизовані системи контролю мікроклімату значно підвищують ефективність управління вирощуванням рослин і стають цінними інструментами для сучасного сільського господарства.

1.7 Висновки до розділу

Обговорено значення комп'ютерно-інтегрованих технологій для управління мікрофермами, з акцентом на використання технологій IoT для підвищення ефективності процесів. Визначено переваги IoT, включаючи простоту використання та швидкість спрацювання.

Розглянуто існуючі мікроферми, зокрема різні типи міні-теплиць, які можуть бути адаптовані до конкретних потреб користувачів. Проаналізовано автоматизовані системи контролю, які забезпечують точність у регулюванні мікроклімату, економію ресурсів та зменшення трудозатрат.

Також розглянули автоматизовані системи контролю мікроклімату які є важливими інструментами для оптимізації умов росту рослин у теплицях і мікрофермах. Вони забезпечують автоматичний моніторинг і контроль таких параметрів, як температура, вологість і освітлення, значно підвищуючи ефективність і продуктивність мікроферм.

2 РОЗРОБКА СХЕМ АВТОМАТИЗОВАНОЇ СИСТЕМИ І ВИБІР АПАРАТНИХ ЗАСОБІВ ДЛЯ ЇЇ РЕАЛІЗАЦІЇ

2.1 Опис структури управління мікрофермою

Основні етапи розробки включають:

1. Визначення вимог: На початковому етапі важливо чітко визначити вимоги до системи, включаючи параметри мікроклімату, які необхідно контролювати, а також функції моніторингу та управління. Це допоможе сформулювати основні цілі та функціональність програмного забезпечення.
2. Проектування архітектури: Архітектура програмного забезпечення повинна включати всі основні компоненти системи. Основні елементи включають:
 - Сенсори: Визначити, які сенсори будуть використовуватися (наприклад, для вимірювання температури, вологості).
 - Контролер NodeMCU: Налаштування мікроконтролера для обробки даних від сенсорів та управління елементами.
 - Програмне забезпечення для моніторингу: Розробити інтерфейс для відображення даних, налаштування параметрів і отримання інформації.
3. Розробка програмного забезпечення: На цьому етапі здійснюється написання коду для кожного з компонентів системи. Це включає:
 - Програмування NodeMCU: Використання Arduino IDE для написання коду, який обробляє дані від сенсорів, здійснює управління та виконує комунікацію з сервером або мобільним додатком.
 - Створення веб-інтерфейсу: Розробка веб-додатку або мобільного додатку для моніторингу та управління системою.

4. Тестування та налагодження: Після завершення розробки важливо провести тестування всіх компонентів системи. Це включає перевірку коректності роботи сенсорів, управління акторами, а також взаємодії між NodeMCU та програмним забезпеченням. Виправлення виявлених помилок та оптимізація коду є важливими етапами для забезпечення стабільності системи.
5. Впровадження та моніторинг: Після успішного тестування система може бути впроваджена на мікрофермі або в теплиці. Важливо забезпечити постійний моніторинг роботи системи, щоб вчасно виявляти і вирішувати можливі проблеми.

Структура програмного забезпечення для системи контролю мікроклімату складається з кількох ключових елементів, кожен з яких виконує специфічну роль у забезпеченні ефективного управління умовами вирощування рослин. Огляд елементів системи контролю мікрокліматом включає:

Сенсорна мережа: Цей компонент складається з різних датчиків, які вимірюють фізичні параметри, такі як температура, вологість, освітленість та інші. Сенсори можуть бути розміщені в різних частинах мікроферми, забезпечуючи всебічний моніторинг умов.

Контролер: Це пристрій, який обробляє дані, отримані від сенсорів. Контролери можуть бути як простими мікроконтролерами, так і потужнішими промисловими комп'ютерами, здатними виконувати складні алгоритми обробки даних та управління.

Програмне забезпечення для моніторингу та управління: Це програмний інтерфейс, через який користувач може отримувати дані від сенсорів, налаштовувати параметри контролерів та здійснювати моніторинг в реальному часі. Це може бути веб-додаток, мобільний додаток або десктопна програма, що забезпечує зручний доступ до інформації та управлінських функцій.

Система комунікації: Це компонент, який забезпечує зв'язок між сенсорами, контролерами, акторськими елементами та програмним забезпеченням. Він може включати бездротові технології або дротові рішення, що забезпечують надійну передачу даних.

База даних: Для зберігання історичних даних про параметри мікроклімату, а також результатів роботи системи, потрібна база даних. Вона дозволяє вести моніторинг змін з часом, що корисно для аналізу ефективності управління та вдосконалення алгоритмів[12].

2.1.1 Визначення вимог

Метою цього проекту є створення комп'ютерно-інтегрованого інтелектуального інтерфейсу для керування мікрофермою, який може відстежувати та контролювати такі параметри, як температура, вологість та рівень освітлення.

Для впровадження системи необхідно забезпечити можливість вимірювання температури за допомогою відповідного датчика з виведенням даних у реальному часі через інтерфейс. Вологість також необхідно вимірювати за допомогою датчика, а дані відображати в інтерфейсі. Що стосується освітлення, система повинна мати можливість використовувати датчики для вимірювання інтенсивності світла. Інтерфейс користувача має бути зручним та забезпечувати можливість віддаленого доступу та керування через комп'ютер або мобільний пристрій.

При написанні програмного забезпечення алгоритм описує логічну послідовність операцій. Кожен алгоритм — це список визначених дій або інструкцій для вирішення певної проблеми. Алгоритм починається з опису інструкцій обчислювального процесу, і ці інструкції відбуваються у визначеному наборі станів, які також мають кінцевий стан. Переходи з одного стану в інший не обов'язково є детермінованими. Деякі алгоритми не включають випадкові елементи. Одним із способів візуалізації алгоритмів роботи системи, на додаток до добре відомої блок-схеми, є UML-діаграма діяльності інтелектуальної системи контролю мікроклімату (рис. 2.1).

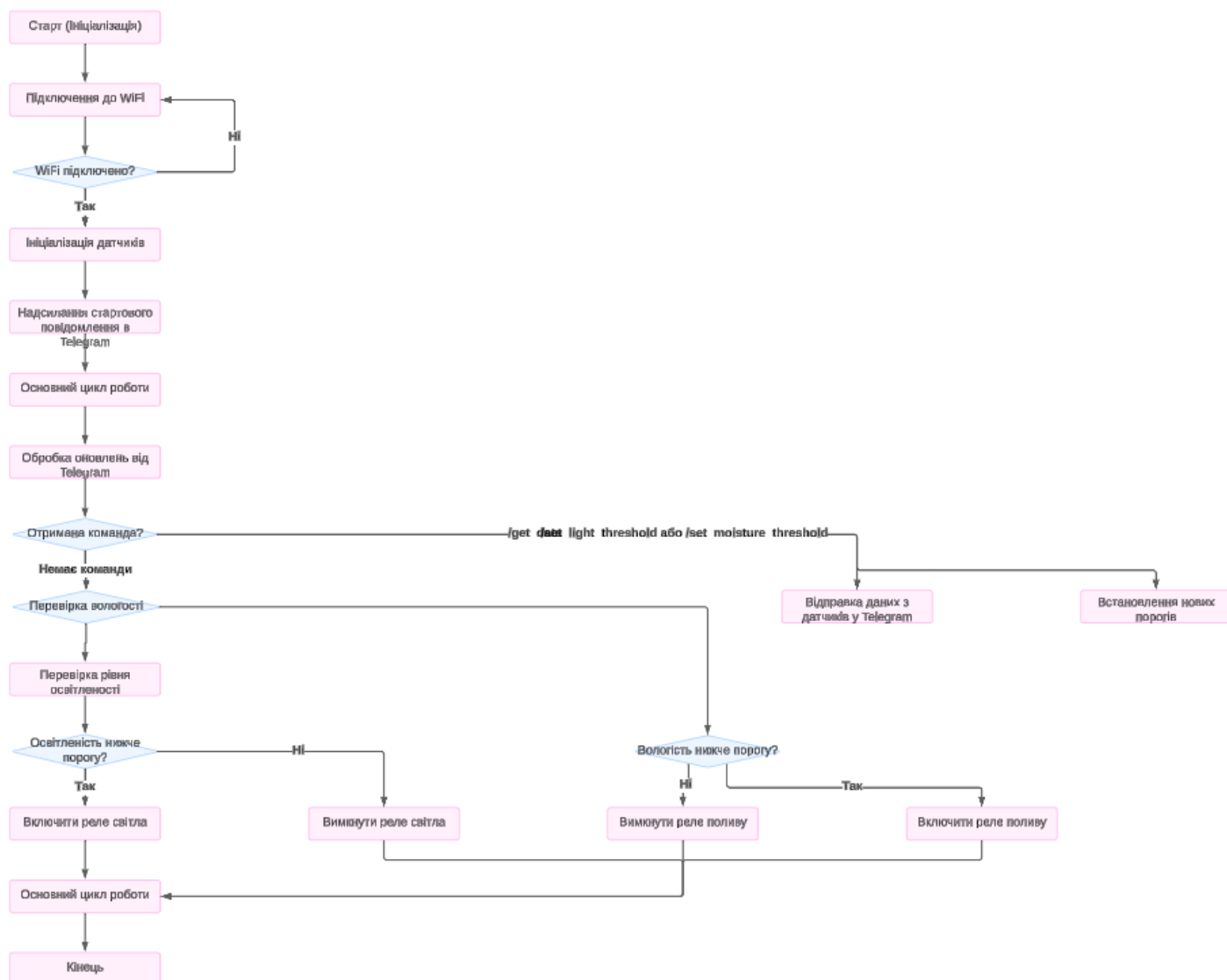


Рисунок 2.1 — UML-діаграма діяльності інтелектуальної системи контролю мікроклімату

2.2 Розробка апаратної схеми мікроферми

Останніми роками набуває популярності впровадження систем автоматизації в господарствах для покращення процесів вирощування та догляду за рослинами. Принципи «Розумного будинку», що полегшують керування життєдіяльністю приміщень для комфортного проживання, тепер широко застосовуються і в сільському господарстві, зокрема для обладнання мікроферм. Такі автоматизовані системи керування допомагають створити оптимальні умови для росту рослин, знижуючи трудовитрати та підвищуючи продуктивність.

Зокрема, автоматизація процесів у мікрофермах дозволяє власникам приватних господарств економити ресурси та стабільно отримувати якісний врожай. З огляду на це, розробка структурної схеми «розумної» мікроферми стає актуальним завданням для забезпечення комплексного підходу до управління господарством у загальній системі розумного житла. Спочатку розглянемо базові аспекти мікроферми з автоматизацією основних технологічних процесів для досягнення максимальної ефективності (Рис. 2.2).

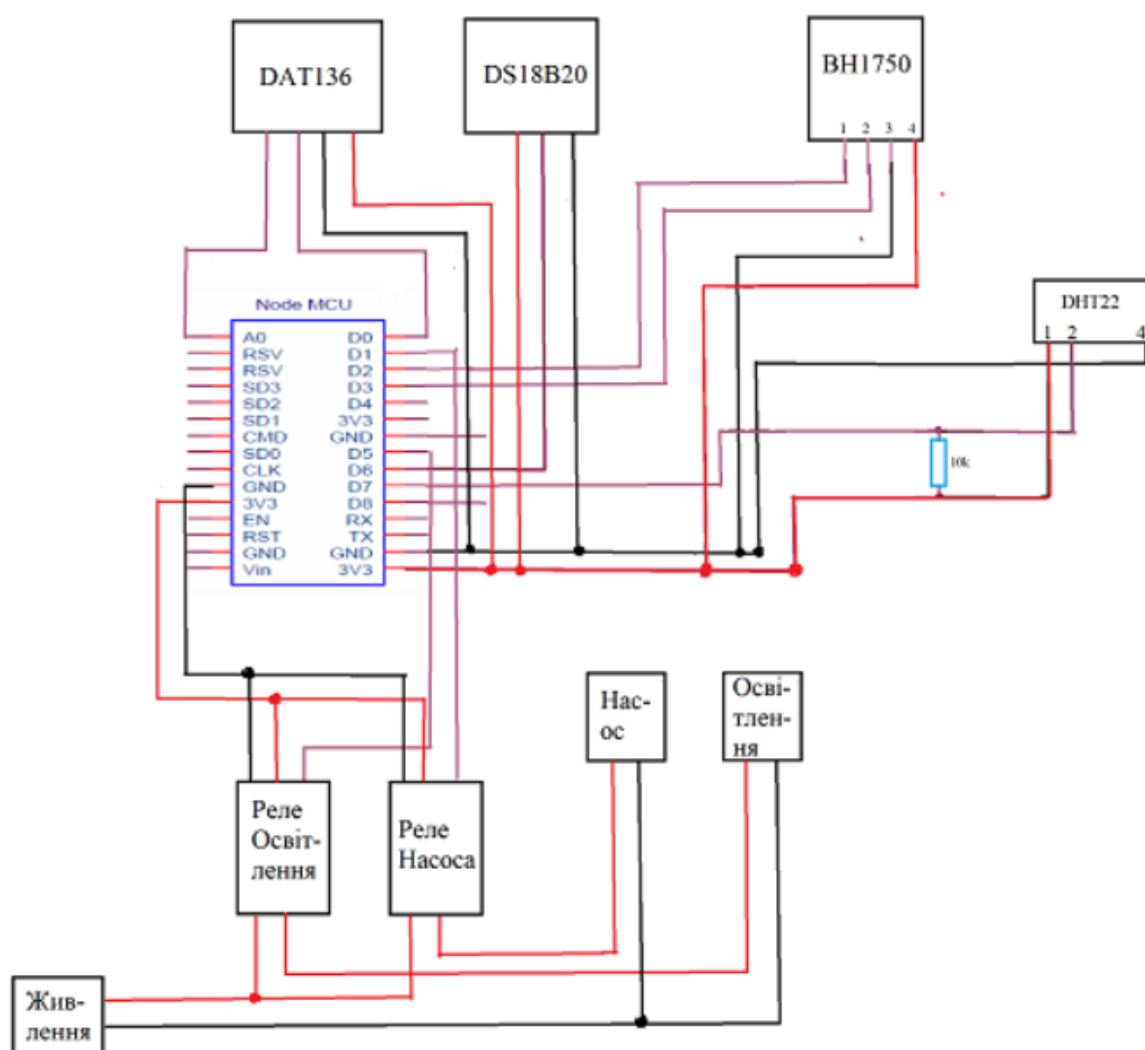


Рисунок 2.2 — Апаратна схема мікроферми

ESP8266 NodeMCU діє як центральний контролер, керуючи всіма підключеними компонентами, обробляючи дані з датчиків і керуючи реле. Датчики включають DHT22, який підключений до форми D7 і зчитує навітряну температуру і вологість, і

DS18B20, який підключений до D6 через OneWire і вимірює температуру ґрунту. Датчик освітленості BH1750 підключений до контактів D2 (SCL) і D3 (SDA) через інтерфейс I2C. Він зчитує інтенсивність світла, підключаючись до виводів D2 (SCL) і D3 (SDA) через інтерфейс I2C. Датчик вологості ґрунту вимірює рівень вологості через аналоговий вихід, підключений до A0, і логічний сигнал, що вказує на сухість або вологість ґрунту через цифровий вихід, підключений до D0.

Реле керування освітленням підключено до виводу D5 і вмикається або вимикається залежно від рівня освітленості (нижче або вище 3 люкс); реле поливу, підключене до D1, активується, коли рівень вологості ґрунту низький; NodeMCU підключається до мережі Wi-Fi для передачі даних до Telegram-бота. Він передає дані та надає актуальну інформацію про кліматичні параметри мікроферми у відповідь на команди користувача.

2.3 Розробка структурної схеми мікроферми

Система представляє собою електронний пристрій, призначений для автоматизації процесів поливу та освітлення рослин. Основним елементом є мікроконтролер NodeMCU, який виконує роль "мозку" системи. До нього підключено низку датчиків, що збирають інформацію про стан навколишнього середовища та рослини. Ці дані аналізуються мікроконтролером, і на їх основі приймаються рішення про включення або вимкнення виконавчих механізмів (насосів, ламп).

Датчики, такі як датчики вологості ґрунту, температури повітря і ґрунту, а також датчик освітленості, постійно збирають інформацію про стан навколишнього середовища. Отримані дані передаються на вхід мікроконтролера NodeMCU. Мікроконтролер, за допомогою вбудованого програмного забезпечення, аналізує отримані дані і порівнює їх із заданими значеннями. Наприклад, якщо вологість ґрунту опустилася нижче встановленого порогу, мікроконтролер віддає команду на включення насоса для поливу. Аналогічно, при недостатньому рівні освітленості включається підсвічування. Для виконання команд мікроконтролера використовуються реле.

Розглянемо 2 варіанта структурної схеми, з додатковим живленням, та без(Рис. 2.3.;2.4).

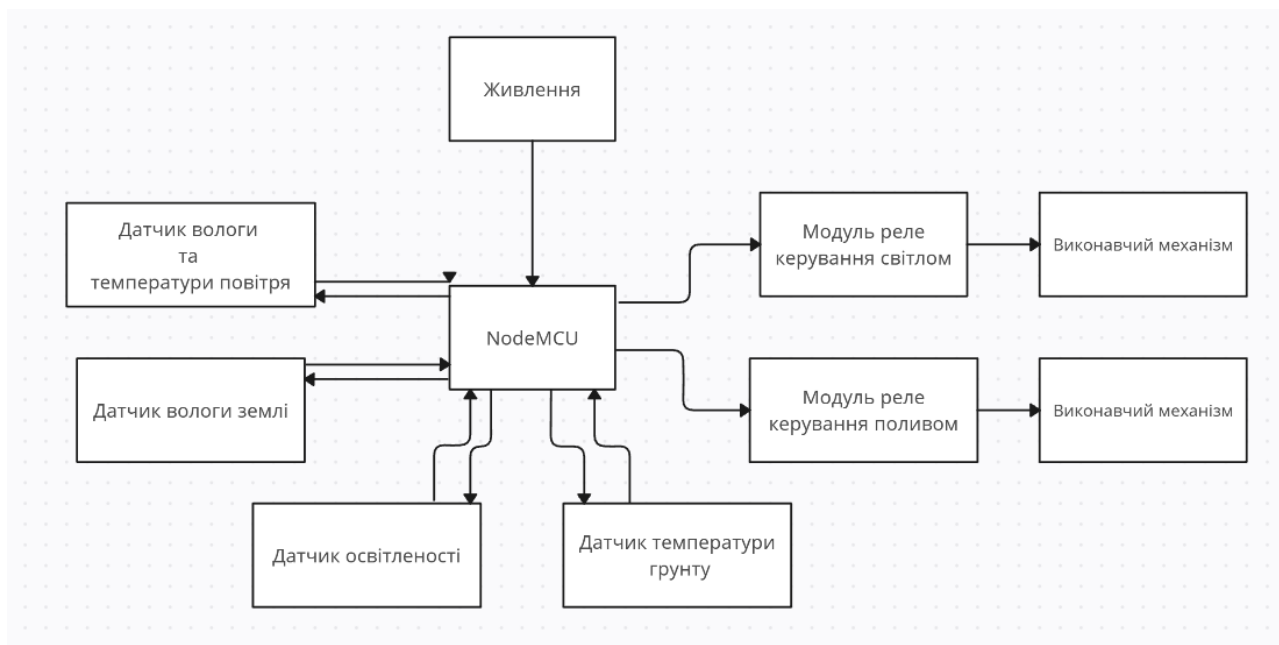


Рисунок 2.3 — Структурна схема мікроферми

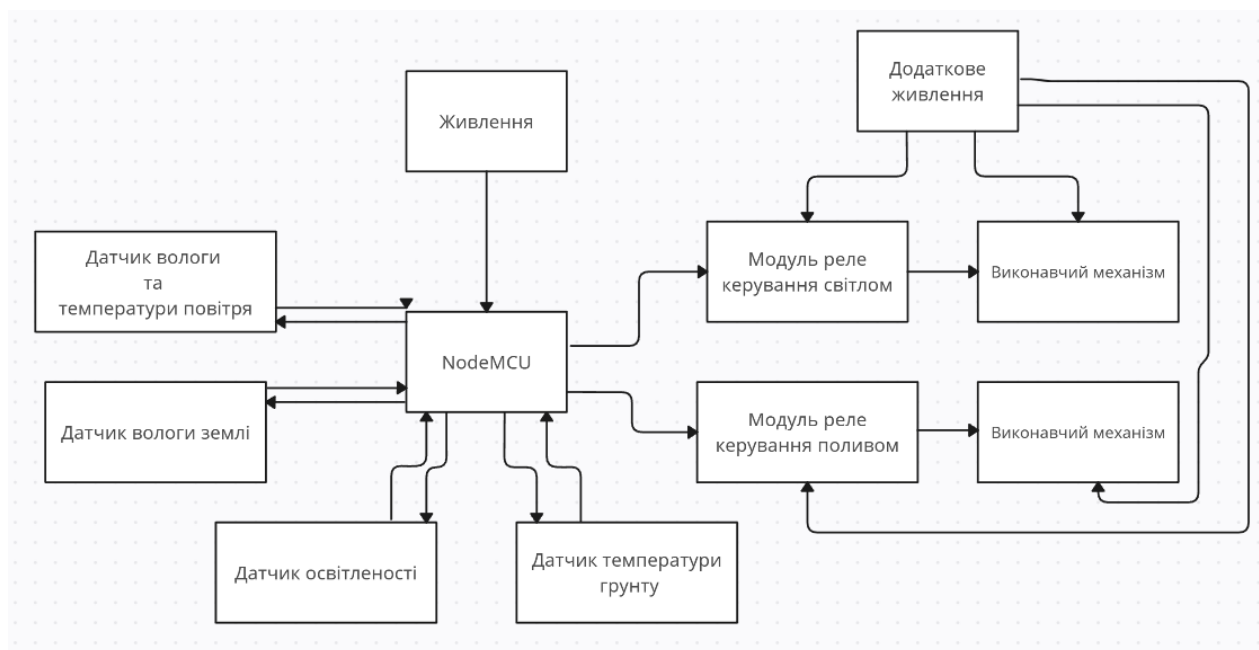


Рисунок 2.4 — Структурна схема мікроферми з додатковим живленням

Додаткове живлення для системи не тільки забезпечує стабільність, але й підвищує її надійність, особливо коли мова йде про автоматизовані рішення, які працюють без нагляду. Замість того, щоб покладатися лише на живлення мікроконтролера, ми маємо окреме джерело для енергоємних компонентів, що суттєво знижує ризик перегріву, який може виникати при надмірному споживанні енергії.

Додаткове джерело живлення дозволяє використовувати компоненти з різними вимогами до напруги. Наприклад, деякі датчики можуть потребувати 3,3 В, тоді як виконавчі пристрої (реле, двигуни) краще працюють на 5 В або більше. Використовуючи додаткове живлення, ми можемо підібрати кожному елементу найбільш підходящу напругу, не знижуючи ефективність і безпеку інших компонентів.

Окрім цього, додаткове живлення знижує вплив електромагнітних завад, оскільки навантаження розподіляється між джерелами, що зменшує можливість перешкод в передачі сигналів. Це особливо важливо для чутливих датчиків, які потребують стабільних умов для точної роботи.

Додаткове джерело живлення також підвищує гнучкість у розширенні системи, дозволяючи легко підключати нові компоненти або навіть об'єднувати різні системи без ризику перевантаження. Це корисно, якщо планується подальше розширення функціоналу.

2.4 Вибір необхідного обладнання для автоматизації мікроферми

Для досягнення усіх поставлених цілей, в роботі будуть використовуватися:

1. Модуль керування.
2. Сенсори температури землі та повітря, вологості землі та повітря, освітлення.
3. Виконавчі механізми такі як реле, до яких підключено лампочку та насос.
4. Джерело живлення

2.4.1 Вибір типу керуючого модуля

Arduino — це серія платформ, створених для простого й ефективного прототипування. Вона базується на мікроконтролерах Atmel або Microchip і програмується через зручне середовище[22].

Середовище Arduino IDE. Arduino підходить як для початківців, так і для досвідчених розробників завдяки простоті використання, великій кількості готових прикладів, бібліотек і широкій спільноті. Платформи, такі як Arduino Uno, Mega або Nano, дозволяють створювати різноманітні системи автоматизації, додаючи функціонал через спеціальні модулі — шилди, які забезпечують підтримку Wi-Fi, Bluetooth, дисплеїв, моторів тощо. Однак Arduino має обмежену обчислювальну потужність і пам'ять, що ускладнює його використання у складних системах чи проектах, які потребують обробки великих обсягів даних або високої швидкості(рис 2.5).

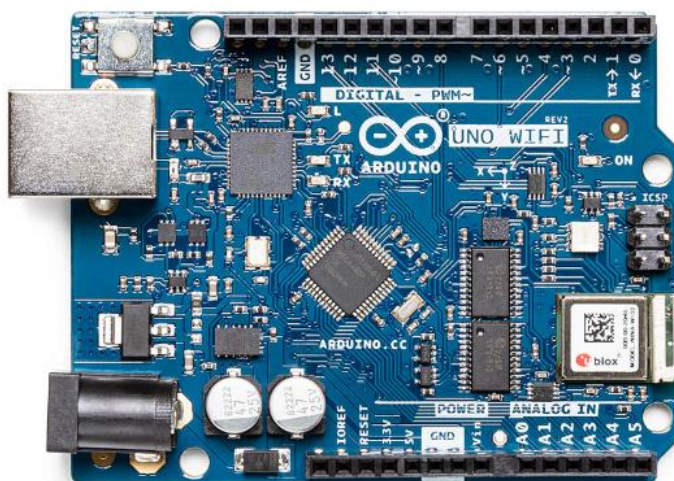


Рисунок 2.5 — Arduino UNO

NodeMCU, на відміну від Arduino, є більш сучасним і функціональним рішенням. Ця платформа базується на потужному мікроконтролері ESP8266 або ESP32, що забезпечує підтримку Wi-Fi, а також інколи Bluetooth, "з коробки". NodeMCU програмується як у середовищі Arduino IDE, так і за допомогою інших мов, наприклад Lua чи MicroPython, що робить його надзвичайно гнучким. Завдяки потужнішому процесору, більшій пам'яті й інтеграції бездротових технологій, NodeMCU ідеально підходить для проектів IoT. З ним можна не лише керувати пристроями, але й створювати

цілі системи автоматизації з віддаленим доступом та управлінням через інтернет. У порівнянні з Arduino, NodeMCU дозволяє більш ефективно працювати зі складними даними та реальним часом, але потребує трохи більше знань для початкового налаштування(рис 2.6).



Рисунок 2.6 — NodeMCU

Проект Arduino створив апаратне забезпечення з відкритим вихідним кодом і SDK програмного забезпечення для їхнього універсального контролера IoT. Подібно до NodeMCU, апаратне забезпечення Arduino являє собою плату мікроконтролера з USB-роз'ємом, світлодіодними індикаторами та стандартними контактами даних. Він також визначає стандартні інтер-фейси для взаємодії з датчиками чи іншими платами. Але на відміну від NodeMCU, плата Arduino може мати різні типи чіпів ЦП (зазвичай чіп ARM або Intel x86) з чіпами пам'яті та різними середовищами програмування. Також існує еталонний дизайн Arduino для чіпа ESP8266. Однак гнучкість Arduino також означає значні відмінності між різними постачальниками. Обидві платформи часто використовуються разом: Arduino для локального управління і NodeMCU як шлюз для зв'язку з мережею, створюючи таким чином потужну систему з можливістю обміну даними між компонентами. Наприклад, більшість плат Arduino не мають можливостей WiFi, а деякі навіть мають послідовний порт даних замість порту USB[9]. Основні технічні характеристики NodeMCU наведені в таблиці 2.1.

Таблиця 2.1 – Технічні характеристики модуля NodeMCU ESP8266

Процесор	ESP8266, 32-бітний RISC процесор з тактовою частотою до 80 МГц (підтримка до 160 МГц).
Пам'ять	64 КБ пам'яті для інструкцій. 96 КБ оперативної пам'яті (SRAM) для зберігання даних. 4 МБ флеш-пам'яті.
Wi-Fi	Вбудований Wi-Fi модуль IEEE 802.11 b/g/n з підтримкою TCP/IP.
Живлення	Напруга живлення: 5 В (вхід через micro-USB) або 3,3 В через VIN пін. Робоча напруга мікроконтролера: 3,3 В.
Цифрові порти (GPIO)	11 цифрових входів/виходів (GPIO). Піни можуть працювати в режимах вхід, вихід, ШІМ, I2C, SPI та ін.
Аналоговий вхід	Один аналоговий вхід (A0) з діапазоном напруги 0–3,3 В.
Інтерфейси	I2C (Serial Clock Line (SCL) і Serial Data Line (SDA)). SPI (інтерфейс для зв'язку з датчиками). UART (підтримує серійний інтерфейс для зв'язку та програмування).
Живлення	Споживання електроенергії в режимі сну — менше 1 мА, у робочому режимі — близько 70 мА.
Розміри	50 мм x 25 мм

З наведеної вище інформації, для поточного проекту краще застосувати плату NodeMCU ESP8266.

2.4.2 Вибір датчиків вологості і температури повітря

Вибір датчиків вологості і температури повітря є важливим етапом у створенні системи моніторингу мікроклімату. Від точності й стабільності їхньої роботи залежить ефективність автоматизації керування умовами вирощування.

Для нашого проекту основними критеріями вибору є точність вимірювання, стабільність роботи, сумісність із NodeMCU, доступність на ринку, а також ціна.

Одним із найпоширеніших варіантів є DHT22 . Цей датчик відзначається високою точністю: $\pm 0,5^{\circ}\text{C}$ для температури та $\pm 2\%$ RH для вологості. DHT22 має цифровий вихід, що дозволяє легко інтегрувати його з NodeMCU, використовуючи бібліотеки Arduino IDE. Ще однією перевагою є простота налаштування та широке розповсюдження, що спрощує пошук технічної інформації чи прикладів коду. Однак DHT22 має дещо повільний час відгуку (близько 2 секунд), що може бути обмеженням у системах з високою динамікою змін мікроклімату(рис. 2.7).[23]

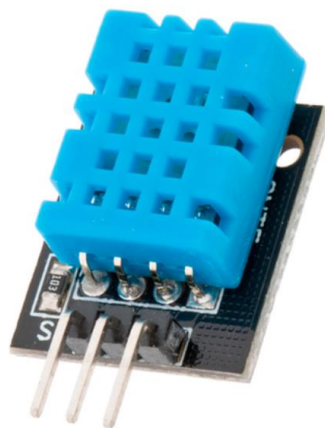


Рисунок 2.7 — Датчик DHT22

Як альтернативу, можна розглянути BME280. Цей датчик має значно ширший функціонал: окрім вимірювання температури (точність $\pm 1^{\circ}\text{C}$) і вологості (точність $\pm 3\%$ RH), він також фіксує атмосферний тиск. Це робить його більш універсальним і придатним для складніших завдань. Він підтримує як цифрові інтерфейси I²C і SPI,

так і більшу швидкість відгуку, що дозволяє збирати дані в реальному часі. Однак BME280 зазвичай дорожчий і може вимагати трохи більше знань для інтеграції.

Для систем, де потрібен лише базовий рівень точності, наприклад, у тестових проектах, можна використовувати DHT11. Цей датчик дешевший, але менш точний ($\pm 2^{\circ}\text{C}$ для температури і $\pm 5\%$ RH для вологості) та має вужчий діапазон вимірювань. Він також має подібний до DHT22 цифровий інтерфейс.

Ключовим фактором є робочий діапазон умов, у яких працюватиме система. Для більшості мікроферм діапазон $0\text{--}50^{\circ}\text{C}$ і $20\text{--}90\%$ RH, що покривають DHT22 та BME280, характеристики DHT22 наведені в таблиці. 2.2.

Таблиця 2.2 – Технічні характеристики сенсора температури DHT22

Тип датчика	Цифровий датчик для вимірювання температури та вологості.
Діапазон вимірювання	Температура: від -40°C до $+80^{\circ}\text{C}$. Вологість: від 0% до 100% RH (відносна вологість).
Точність вимірювань	Температура: $\pm 0.5^{\circ}\text{C}$. Вологість: $\pm 2\text{--}5\%$ RH.
Роздільна здатність	Температура: 0.1°C . Вологість: 0.1% RH.
Час реакції	До 2 секунд для стабільних показників.
Живлення	Напруга живлення: $3.3\text{--}5.5\text{ В}$. Споживаний струм у вимірювальному режимі: близько 1.5 мА .
Інтерфейс	Один цифровий пін для передачі даних. Вихідний сигнал: серійний цифровий.
Частота вимірювань	Приблизно раз на 2 секунди (максимальна частота запиту – 0.5 Гц).
Розміри	$15\text{ мм} \times 25\text{ мм} \times 7\text{ мм}$

DHT22 Цифровий датчик температури та вологості. Температура вимірюється за допомогою термістора з негативним температурним коефіцієнтом (NTC). Відносна вологість вимірюється ємнісним датчиком. Датчик відкалібрований виробником, що забезпечує високу стабільність і лінійність показників температури і вологості.

DHT22 складається з ємнісного датчика вологості і термістора для вимірювання температури. Ємнісний сенсор визначає рівень вологості, базуючись на зміні ємності, викликаній зміною вологості навколо датчика. Термістор вимірює температуру шляхом зміни свого опору залежно від температури навколишнього середовища. DHT22 використовує однодротовий інтерфейс для передачі даних. Це означає, що для обміну даними між мікроконтролером і датчиком використовується один провід (DATA), що робить підключення простішим. Наші потреби датчик повністю задовільняє.

2.4.3 Вибір датчика температури землі

Вибір датчика для вимірювання температури ґрунту є ключовим етапом у розробці системи моніторингу мікроклімату, оскільки температура ґрунту значно впливає на ріст і розвиток рослин. Для нашого проекту основними критеріями є точність, надійність, простота інтеграції з NodeMCU та стійкість до впливу вологи й агресивного середовища.

Ще одним потенційним варіантом є аналогові датчики, наприклад, NTC-термістори. Вони дешевші, але вимагають використання додаткових схем (наприклад, резистивного подільника напруги) і АЦП (аналогово-цифрового перетворення) для зчитування даних, що може бути менш зручним, особливо при роботі з NodeMCU, де доступний лише один аналоговий вхід.

Для проектів, де потрібна висока точність і стабільність, можна розглянути спеціалізовані сенсори для агрономічних застосувань, такі як серії SoilTemp. Вони забезпечують високий рівень точності й витримують екстремальні умови, але значно дорожчі й складніші в інтеграції.

Датчик температури DS18B20 – це цифровий сенсор, призначений для точного вимірювання температури в діапазоні від -55°C до $+125^{\circ}\text{C}$, розміщений у водонепроникному корпусі з рівнем захисту IP67. Завдяки цьому рівню захисту він здатний витримувати пил та вологу, що дозволяє використовувати його в різних умовах: як у середовищах із високою вологістю, так і в запилених місцях. Така конструкція датчика робить його незамінним для проєктів, де потрібно точне вимірювання температури в несприятливих умовах, наприклад, у теплицях, акваріумах або на виробничих об'єктах.[24]

DS18B20 використовує цифровий інтерфейс 1-Wire, що значно спрощує його інтеграцію з мікроконтролерами, такими як Arduino або NodeMCU. Цей інтерфейс дозволяє підключати кілька датчиків до одного пін-контакту мікроконтролера, що забезпечує зручність при створенні багатоточкових систем моніторингу температури. Завдяки міцній конструкції та простоті використання, датчик DS18B20 є одним з найпопулярніших для систем контролю клімату та автоматизації(рис. 2.8).



Рисунок 2.8 — Датчик DS18B20

Серцевиною датчика є сенсорний елемент DS18B20, який забезпечує високу точність вимірювань з похибкою до $\pm 0.5^{\circ}\text{C}$ у діапазоні від -10°C до $+85^{\circ}\text{C}$. Оскільки

датчик має захисну оболонку з ПВХ, рекомендується обмежити верхню межу вимірювання до 100°C, щоб уникнути пошкодження корпусу. Конструкція датчика дозволяє його занурювати в рідину для вимірювання температури, що робить його зручним для моніторингу рідин у різних промислових і побутових умовах. Основні технічні характеристики DS18B20 наведені в таблиці 2.3.

Таблиця 2.3 – Технічні характеристики сенсора температури DS18B20

Тип датчика	Цифровий датчик температури з інтерфейсом 1-Wire.
Діапазон вимірювання температури	Від -55°C до +125°C.
Точність вимірювань	Від -10°C до +85°C: $\pm 0.5^{\circ}\text{C}$.
Роздільна здатність	Програмно регульована: 9, 10, 11 або 12 біт. За умовчанням — 12 біт (0.0625°C).
Живлення	Напруга живлення: 3.0–5.5 В.
Інтерфейс	Однопровідний (1-Wire) протокол.
Час вимірювання	12-бітна роздільна здатність — до 750 мс.
Робочі умови	Робоча вологість: до 100% RH
Розміри	Герметизована трубка для занурення
Інші можливості	Унікальний 64-бітний серійний код дозволяє ідентифікувати кожен датчик у мережі

Для нашого проекту найкращим вибором є датчик DS18B20. Він забезпечує високу точність вимірювання, просте підключення за протоколом 1-Wire, водонепроникність для роботи у вологому середовищі та сумісність із NodeMCU. Ці характеристики роблять його оптимальним для моніторингу температури ґрунту.

2.4.4 Вибір датчика вологості ґрунту

Датчик вологості ґрунту - це невеликий пристрій, який дозволяє точно контролювати рівень вологості в ґрунті, особливо корисний у застосуваннях для автоматизації поливу рослин, як в умовах теплиць, так і вдома. Пристрій складається з двох основних компонентів: сенсорної пластини, яка занурюється в ґрунт для вимірювання вологості, і електронного модуля, що обробляє дані. Сенсор працює за принципом провідності – якщо ґрунт вологий, то провідність між його контактами зростає, а при висиханні – знижується. Електронний модуль, побудований на основі мікросхеми LM393, зчитує цей сигнал і переводить його в цифровий або аналоговий вихід, залежно від рівня вологості(рис. 2.9).

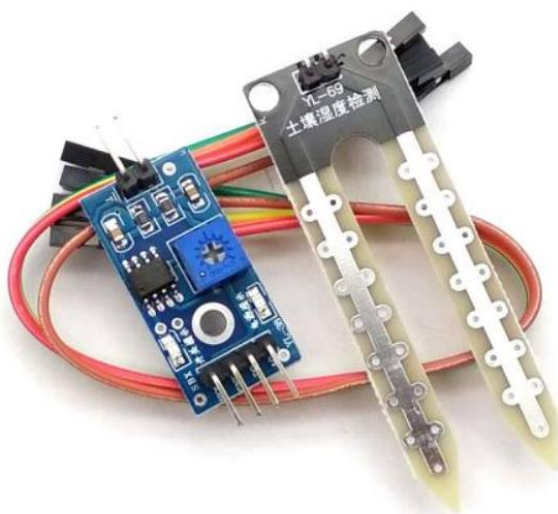


Рисунок 2.9 — Датчик вологості ґрунту

На платі є синій світлодіод, який вказує на наявність живлення, та червоний, який загоряється під час передачі даних до контролера. Основні технічні характеристики датчика наведені в таблиці 2.4.

Таблиця 2.4 – Технічні характеристики сенсора вологості

Тип модуля	Аналоговий і цифровий датчик для вимірювання вологості ґрунту з підсилювачем і контролем порогу спрацьовування.
Діапазон вимірювання	Від сухого до максимально вологого стану ґрунту
Принцип роботи	Вимірює електропровідність ґрунту. При високій вологості опір зменшується, при низькій — збільшується.
Інтерфейси та вихідні сигнали	Аналоговий вихід (A0): Напруга змінюється в залежності від вологості (від 0 до V_{cc}). Цифровий вихід (D0): Видає високий або низький сигнал, залежно від вологості. Поріг спрацьовування налаштовується потенціометром.
Живлення	Напруга живлення: 3.3–5 В. Споживання енергії: низьке, але для запобігання корозії зондів рекомендується вимірювати періодично, а не постійно.
Розміри	Датчик YL-69: 60 мм. Модуль YL-38: 30 мм x 15 мм.

Особливістю цього датчика є можливість регулювання чутливості через потенціометр, розташований на модулі. Після підключення до живлення та мікроконтролера користувач може за допомогою змінного резистора налаштувати точку, при якій датчик "спрацьовує" – тобто визначити поріг вологості, нижче якого буде вмикатися сигнал про сухість ґрунту. Цей сигнал легко обробляється через Arduino або інший

контролер, а цифровий вихід дозволяє використовувати датчик навіть без необхідності в складних аналогових обчисленнях, спрощуючи його інтеграцію в систему.

Живлення для датчика може надходити від Arduino, іншого мікроконтролера, або зовнішнього джерела з напругою 3.3–5 В. Плата обладнана 4-контактним штирьовим інтерфейсом для підключення до контролера, що включає контакти: VCC для живлення, GND для заземлення, D0 для цифрового виходу, та A0 для аналогового. Крім того, сенсорна пластина підключається до модуля за допомогою окремого 2-контактного роз'єму. Така конструкція дозволяє датчику легко інтегруватися в автоматизовані системи для регулювання поливу, а також забезпечує стабільну роботу в системах мікроклімату.

2.4.5 Вибір датчика освітлення

Для вибору датчика освітлення в нашому проекті слід врахувати потребу в точному вимірюванні рівня освітленості в люксах, сумісність із мікроконтролером NodeMCU та простоту інтеграції. Найбільш поширеними та ефективними варіантами є BH1750 та GY-302.

Цифровий датчик світла BH1750, який вимірює інтенсивність світла (освітленість) навколишнього середовища та передає ці дані у вигляді цифрового сигналу. Він використовується в багатьох проектах, пов'язаних з автоматизацією, таких як регулювання рівня освітлення в кімнатах і системах розумного будинку, а також моніторинг навколишнього середовища (рис. 2.10).

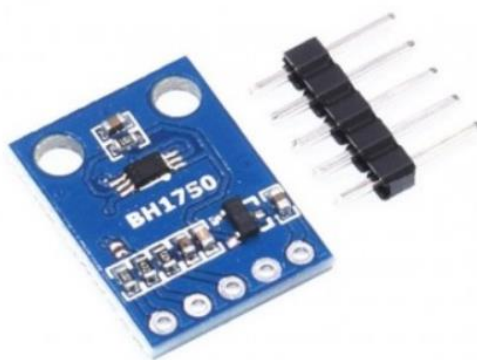


Рисунок 2.10 — Датчик BH1750

GY-302, який також базується на модулі BH1750, має ті ж переваги, але часто поставляється в зручнішому форм-факторі з додатковими підключеннями для роботи з NodeMCU. Він добре підходить для автоматизації освітлення, дозволяючи отримувати точні дані в реальному часі.

BH1750 працює на основі датчика світла, який вимірює інтенсивність видимого світла. Датчик перетворює світло в електричний сигнал і використовує внутрішній аналого-цифровий перетворювач для передачі інформації в цифровому вигляді. Завдяки цьому процесу вимірювання є дуже точними та стабільними[10]. Основні технічні характеристики датчика наведені в таблиці 2.5.

Таблиця 2.5 – Технічні характеристики датчика освітленості

Тип датчика	Цифровий датчик освітленості для вимірювання рівня освітленості в люксах
Діапазон вимірювання	Від 0 до 65535 люкс
Точність	Висока точність з лінійним вимірюванням освітленості.
Інтерфейс	Інтерфейс I2C. Адреса I2C за замовчуванням — 0x23 або 0x5C.
Живлення	Напруга живлення: 3–5 В. Споживання енергії: дуже низьке.
Час вимірювання	120 мс в режимі високої точності
Розміри	18 мм x 11 мм

Для нашого проєкту найкращим вибором є датчик BH1750. Він забезпечує високу точність вимірювання освітленості в діапазоні від 1 до 65535 люксів, легко інтегрується з NodeMCU через I2C-протокол і займає мінімум пінів. Завдяки надійності та простоті використання BH1750 ідеально підходить для автоматизації контролю освітлення, що є ключовим аспектом нашої системи.

2.4.6 Вибір виконавчих механізмів

Одноканальний релейний модуль на 5В для Arduino – це пристрій, який дозволяє управляти включенням і вимиканням потужних електричних приладів за допомогою мікроконтролера, наприклад, Arduino. Модуль працює з живленням 5 В і використовує одне реле для комутації високовольтного кола, з яким підключений, що дозволяє контролювати різні пристрої, такі як лампи, насоси або вентилятори, у проєктах автоматизації(рис. 2.11).



Рисунок 2.11 — Модуль реле

Кожен модуль оснащений оптоізоляцією, яка захищає мікроконтролер від високовольтних стрибків у керованому ланцюзі, забезпечуючи безпечну роботу. Інтерфейс підключення простий і зручний – вхідний сигнал управляє реле, яке, в свою чергу, замикає або розмикає контакти підключеного навантаження, вмикаючи або вимикаючи його. Модуль також має індикаторний світлодіод, який сигналізує про стан реле, що допомагає швидко оцінити його активність. Основні технічні характеристики модуля наведені в таблиці 2.6.

Таблиця 2.6 – Технічні характеристики модуля

Тип модуля	Одноканальний релейний модуль для комутації високовольтних та високострумівих навантажень.
Напруга живлення	5 В
Реле	Тип реле: SRD-05VDC-SL-C Напруга спрацювання реле: 5 В. Номинальний струм котушки: 70 мА.
Керуючий сигнал	Логічний рівень: 0 або 1 Напруга керування: 3.3–5 В
Контакти реле	COM NO (Normally Open) NC (Normally Closed)
Індикація	Світлодіод для індикації спрацювання реле
Розміри	50 мм x 25 мм x 18 мм

Вибір насоса для системи автоматизації мікроферми є важливим аспектом, оскільки від цього залежить ефективність і надійність подачі води для поливу. Основними критеріями вибору є тип насоса, його продуктивність, тиск, енергоефективність, простота інтеграції з системою та наявність засобів контролю.

Для автоматизованих систем, які базуються на мікроконтролерах, таких як NodeMCU, найчастіше обирають компактні насоси з постійним струмом. Це пояснюється їхньою невеликою потужністю, низьким енергоспоживанням і легкістю управління. Одним із найпоширеніших варіантів є погрузний насос, який підходить для роботи з невеликими об'ємами води та забезпечує надійну подачу рідини.

Цей компактний погрузний насос призначений для перекачування рідини і чудово підходить для домашньої автоматизації, таких як полив рослин, системи акваріумів і подібних проектів. При швидкості перекачування 80-120 літрів на годину він

забезпечує рівномірну подачу рідини на висоту від 40 до 110 см, що робить його корисним для невеликих систем, таких як ваша мікроферма(рис. 2.12).



Рисунок 2.12 — Погружний насос

Насос працює від низької напруги в діапазоні 2.5-6 В з максимальною потужністю до 1.5 Вт, що дозволяє йому ефективно працювати у автономних системах або від стандартного джерела живлення для електронних проєктів. Його компактний розмір і міцний пластиковий, герметизований корпус надають насосу стійкість до впливу рідини, забезпечуючи довговічність в умовах підвищеної вологості.

Для зручності підключення насос має вихідний діаметр 7.5 мм зовні та 4.5 мм усередині. Він легко інтегрується в мікроферму, де зможе автоматично підтримувати рівень вологості, забезпечуючи точний контроль за подачею води для рослин. Основні технічні характеристики насоса наведені в таблиці 2.7.

Таблиця 2.7 – Технічні характеристики насоса

Матеріал корпусу	пластик, загерметизований
Потужність	1.5 Вт
Швидкість перекачування рідини	80-120 л/год
Максимальна висота підйому рідини	40-110 см
Робоча напруга	2.5-6 В
Розмір	4.5 x 2.4 x 3.3 см

Вибір ламп для освітлення в системі мікроферми має вирішальне значення, оскільки від цього залежить ефективність фотосинтезу, а отже, і ріст рослин. Серед доступних варіантів найбільш поширеними є світлодіодні лампи (LED), люмінесцентні лампи, натрієві лампи високого тиску (HPS) та металогалогенні лампи (MH).

Світлодіоди є найсучаснішим рішенням завдяки низькому енергоспоживанню, довговічності та можливості налаштування спектра світла. Вони дозволяють забезпечити необхідний баланс червоного і синього спектрів, які є найбільш важливими для рослин. Люмінесцентні лампи — це доступний за ціною варіант, проте вони мають меншу ефективність та обмежений термін служби. Натрієві лампи забезпечують інтенсивний світловий потік, але вони споживають більше енергії та мають переважно жовтий спектр, який не завжди оптимальний для рослин. Металогалогенні лампи добре підходять для росту, проте їхній високий рівень тепловиділення може бути проблемою у компактних системах.

Для нашого проєкту найбільш підходящим варіантом є використання LED-ламп, зокрема спеціалізованих фітоламп. Вони забезпечують високу енергоефективність, оптимальний спектр світла для рослин і тривалий термін служби. Крім того, LED-лампи дозволяють регулювати інтенсивність освітлення та легко інтегруються в автоматизовані системи управління.

2.5 Вибір системи виведення інформації

NodeMCU пропонує різні способи виведення інформації, і вибір залежить від потреб проєкту. Найпростішим варіантом є використання серійної консолі через USB-з'єднання. Це зручно для тестування та налагодження, адже дані можна переглядати безпосередньо в Arduino IDE або інших програмах. Проте цей метод обмежений через необхідність постійного підключення до комп'ютера.

Переваги:

- Простота налаштування.
- Зручність для тестування та відлагодження.

Недоліки:

- Потребує постійного підключення до комп'ютера.
- Не підходить для дистанційного моніторингу.

Ще одним популярним підходом є налаштування вбудованого веб-сервера. У цьому випадку NodeMCU працює як точка доступу або підключається до локальної мережі, дозволяючи переглядати дані через браузер за допомогою веб-інтерфейсу. Це рішення ідеально для локального моніторингу без залежності від зовнішніх сервісів, але воно менш зручне для віддаленого доступу.

Переваги:

- Відображення даних у зручному веб-інтерфейсі.
- Доступ з будь-якого пристрою в межах локальної мережі.

Недоліки:

- Обмежена функціональність без додаткових налаштувань (наприклад, хмарного доступу).

Для складніших задач можна використовувати протоколи IoT, як-от MQTT або платформи на зразок ThingSpeak чи Blynk. Такі сервіси дозволяють передавати дані в інтернет для їх графічного відображення, зберігання історії чи отримання доступу до інформації з будь-якої точки світу. Це особливо актуально для віддаленого керування, але вимагає інтернет-з'єднання та може мати певні обмеження у безкоштовних версіях.

Переваги:

- Зручність для складних проектів.
- Віддалений доступ до даних.
- Підтримка зберігання історії даних.

Недоліки:

- Необхідність інтернет-підключення.
- Деякі сервіси платні або мають обмеження у безкоштовній версії.

Telegram-боти є одним із найзручніших методів виведення інформації. Завдяки Telegram Bot API можна легко надсилати дані безпосередньо в чат і отримувати ко-

манди від користувача. Це дозволяє не лише моніторити параметри, як-от температуру чи вологість, а й керувати пристроєм через знайомий інтерфейс Telegram, який доступний на будь-якому пристрої.

Переваги:

- Простий інтерфейс: Telegram доступний майже на всіх пристроях.
- Миттєві повідомлення: Дані відправляються прямо в чат.
- Двосторонній зв'язок: Легко реалізувати керування мікрофермою через команди в боті.
- Безкоштовно: Telegram API не має обмежень для більшості проєктів.

Недоліки:

- Потребує інтернет-з'єднання.

Нарешті, для автономних рішень можна підключити OLED-дисплей до NodeMCU. Це забезпечує компактний спосіб виведення даних без залежності від мережі чи інтернету, хоча й менш зручний для об'ємної інформації або віддаленого доступу.

Переваги:

- Ніяких залежностей від мережі чи інтернету.
- Компактний і автономний спосіб виведення інформації.

Серед усіх варіантів Telegram-бот виділяється своєю простотою використання, доступністю для більшості користувачів і можливістю організувати двосторонній зв'язок із системою.

2.5 Підсумки другого розділу

Для створення автоматизованої системи контролю мікроклімату для мікроферми обрали оптимальні компоненти, які забезпечать ефективність, надійність і простоту впровадження. Основою системи є NodeMCU, яке забезпечує зв'язок з датчиками та виконавчими модулями, дозволяючи створити гнучке та масштабоване рішення.

Для моніторингу вологості та температури повітря ми обрали датчик DHT22, який забезпечує високу точність і стабільність роботи. Для вимірювання температури ґрунту ідеально підходить DS18B20 завдяки його точності, простоті підключення та можливості використання у складних умовах. Для моніторингу рівня освітленості обрано BH1750, який має високу точність і простоту інтеграції з мікроконтролером.

Для забезпечення поливу використовується компактний занурювальний насос, який відповідає вимогам невеликих фермерських систем, а управління освітленням здійснюється за допомогою LED-фітоламп, які забезпечують оптимальний спектр світла для рослин та енергоефективність.

Таким чином, наша система є сучасним, економічно ефективним рішенням для автоматизації мікроферм. Вона дозволяє точно контролювати параметри середовища, забезпечувати необхідні умови для рослин і мінімізувати споживання ресурсів, що робить її зручною для використання як у домашніх умовах, так і в малих фермерських господарствах.

3 РОЗРОБКА АЛГОРИТМІВ І ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

3.1 Обґрунтування вибору програмної платформи

Для роботи у якості контролера було обрано модуль керування NodeMCU, а саме ESP8266, це популярний контролер який використовується у різних галузях, тому для його програмування можна використовувати кілька мов програмування та програмних середовищ.

Розберемо самі популярні середовища та їхні плюси:

1. Lua - прошивка для NodeMCU надає легкий інтерпретатор, що дозволяє писати та виконувати код мовою Lua прямо на платі ESP8266. Завдяки своїй скриптовій природі, Lua не потребує компіляції, що спрощує процес розробки та дозволяє швидко тестувати функціональність. Вона оптимізована для роботи на пристроях з обмеженою пам'яттю, тому підходить для невеликих проєктів, де важлива економія ресурсів.

NodeMCU Lua включає модулі для роботи з основними апаратними інтерфейсами, такими як GPIO, UART, I2C, PWM і ADC, що робить її зручною для підключення датчиків і керування різноманітними компонентами. Вбудована підтримка Wi-Fi дозволяє налаштувати з'єднання з мережею для обміну даними через HTTP, MQTT та інші протоколи, що є корисним для IoT-проєктів.

2. PlatformIO — це сучасне середовище розробки, яке інтегрується з популярними редакторами, як-от Visual Studio Code та Atom, і підтримує безліч платформ, зокрема ESP8266 (NodeMCU). Воно пропонує розширений функціонал для створення, налагодження та управління проєктами, що робить його популярним серед розробників мікроконтролерів.

PlatformIO дозволяє створювати проєкти з гнучкими налаштуваннями, де можна використовувати кілька бібліотек одночасно. Бібліотеки тут легко встановлюються через вбудований менеджер, а PlatformIO автоматично підбирає сумісні версії. Це забезпечує стабільність проєктів і спрощує роботу з оновленнями.

Особливістю PlatformIO є підтримка автоматизованих інструментів налагодження та вбудованого моніторингу послідовного порту, що дає змогу детально аналізувати помилки і відслідковувати роботу мікроконтролера під час виконання програми. PlatformIO також має можливість збирати код для кількох платформ одночасно, що корисно при розробці крос-платформних проєктів.

3. Найпопулярнішим середовищем для програмування NodeMCU є Arduino IDE. Воно завоювало популярність завдяки простоті, зручному інтерфейсу, великій кількості бібліотек і широкій підтримці спільноти. Arduino IDE дозволяє легко працювати з кодом, зокрема з Wi-Fi і різними модулями, що робить його ідеальним вибором для початківців і для швидкого створення IoT-проєктів.

Однією з ключових переваг Arduino IDE є велика бібліотека готових модулів, що дозволяє легко підключати різні датчики, реле, модулі зв'язку та інші компоненти. Бібліотеки тут додаються в кілька кліків, після чого їх можна використовувати в коді без додаткової конфігурації. IDE також має вбудовані приклади коду, що дозволяє швидко протестувати функції компонентів і розібратися, як вони працюють.

Arduino IDE підтримує простий і легкий синтаксис C/C++. Його код зазвичай складається з функцій `setup()` і `loop()`, що відповідають за ініціалізацію та основний цикл роботи програми. Це робить його інтуїтивно зрозумілим і дозволяє швидко адаптуватися до написання коду.

Arduino IDE надає вбудовану компіляцію та прошивку, що дозволяє завантажувати код на мікроконтролер всього кількома кліками. Інтерфейс має простий монітор послідовного порту для перегляду даних, які надсилає плата, що корисно для налагодження проєктів. Також має велику спільноту користувачів і документацію, де легко знайти приклади, відповіді на запитання та інструкції.

Тому оберемо Arduino, адже з ним легко почати навіть новачкам, і він не потребує складних налаштувань. Крім того, Arduino працює на всіх основних платформах, таких як Windows, macOS та Linux, а також підтримує багато популярних плат, як-от ESP8266. Завдяки відкритій архітектурі апаратного забезпечення є багато сумісних плат і модулів на різний бюджет, що дає гнучкість у виборі компонентів та розширенні можливостей проєкту.

3.2 Розробка алгоритму роботи системи мікроферми

Однією з головних особливостей скетчу Arduino є безперервне виконання циклів. Коли мікроконтролер вмикається, він ініціалізує глобальні змінні та бібліотеки, що використовуються головною функцією. Далі виводи налаштовуються на читання і запис даних, що стосуються підключених пристроїв. Ініціалізується та скидається енергонезалежна та енергонезалежна пам'ять.

Функція `setup` викликається лише один раз після запуску системи. Вона відповідає за налаштування датчиків (температури повітря, вологості, освітленості, вологості ґрунту) і реле, а також за підключення до Wi-Fi для отримання команд з Telegram. Після цього система готова до роботи.

Основна функція `loop` безперервно виконує кілька підпрограм. Першою з них є обробка команд з Telegram, яка перевіряє наявність нових повідомлень. Якщо отримано команду `/get_data`, система збирає дані з усіх сенсорів і надсилає їх у Telegram.

Другий викликаний блок називається `checkMoisture`, що відповідає за зчитування вологості ґрунту. Якщо показники вказують на сухий ґрунт, активується реле поливу, яке запускає механізм поливу. У разі вологого ґрунту реле вимикається.

Наступний блок — це `checkLight`, який вимірює рівень освітленості. Якщо освітленість нижча за встановлене значення, активується реле освітлення, що вмикає світло. І навпаки, якщо освітленість достатня, реле вимикається.

Останній етап включає затримку між перевірками, що дозволяє уникнути надмірних запитів до датчиків і реле. Опис програми представлений на рисунку 3.1.

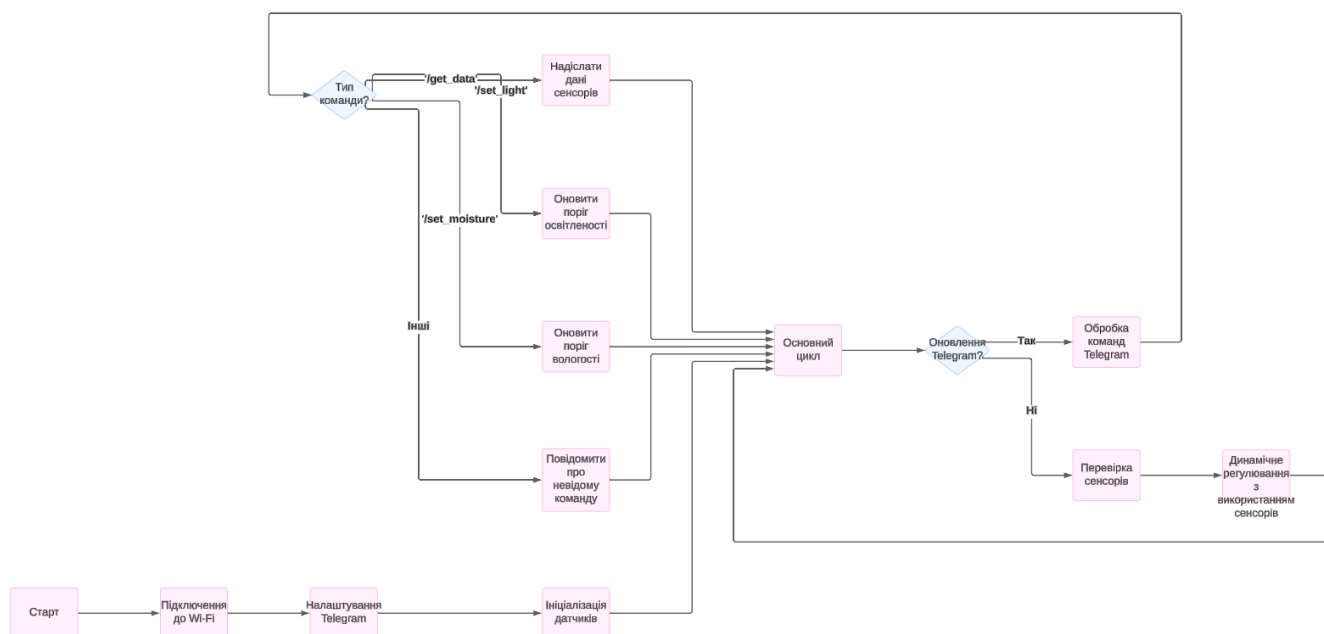


Рисунок 3.1 – Схема програми

3.3 Розробка програмного забезпечення

Програмне забезпечення автоматизованої системи контролю вологості і освітлення реалізоване у вигляді одного програмного модуля, що реалізує зазначений алгоритм управління. Код написано мовою C++ для платформи NodeMCU.

Написання коду починаємо з додавання необхідних бібліотек:

```
#include <ESP8266WiFi.h>
#include <WiFiClientSecure.h>
#include <UniversalTelegramBot.h>
#include <OneWire.h>
#include <DallasTemperature.h>
#include <Wire.h>
#include <BH1750.h>
#include <DHT.h>
```

ESP8266WiFi.h: для роботи з Wi-Fi на платі NodeMCU.

WiFiClientSecure.h: для безпечних з'єднань (HTTPS) з Telegram API.

UniversalTelegramBot.h: для інтеграції з Telegram.

OneWire.h та DallasTemperature.h: для роботи з цифровими термометрами DS18B20.

Wire.h: для роботи з I2C-шиною (використовується для датчика BH1750).

BH1750.h: для роботи з датчиком освітленості.

DHT.h: для роботи з датчиком температури і вологості DHT22.

Далі переходимо до налаштування параметрів Wi-Fi. Тут ми задаємо параметри для підключення до Wi-Fi, включаючи назву мережі (SSID) та пароль.

```
const char* ssid = "....";
```

```
const char* password = "....";
```

Ми визначаємо токен бота Telegram та Chat ID для надсилання повідомлень.

```
#define TELEGRAM_TOKEN "...."
```

```
#define CHAT_ID "...."
```

Визначаємо піни для реле освітлення та поливу

```
const int lightRelayPin = D5;
```

```
const int moistureRelayPin = D1;
```

Проводимо налаштування датчиків. DS18B20, підключеного до піну D6. Датчик вологості до D0.

```
const int moistureSensorPin = D0;
```

```
#define ONE_WIRE_BUS D6
```

```
OneWire oneWire(ONE_WIRE_BUS);
```

```
DallasTemperature sensors(&oneWire);
```

Ініціалізуємо датчик DHT22(виміряє вологість та температуру повітря).

```
#define DHTPIN D7
```

```
#define DHTTYPE DHT22
```

```
DHT dht(DHTPIN, DHTTYPE);
```

Ми визначаємо піни для аналогового та цифрового виходу датчика вологості.

```
#define SOIL_MOISTURE_ANALOG_PIN A0
```

```
#define SOIL_MOISTURE_DIGITAL_PIN D0
```

Тут зберігаємо ID останнього оновлення з Telegram, затримку між запитами та час останньої обробки бота.

```

long lastUpdateId = 0;
unsigned long botRequestDelay = 1000;
unsigned long lastTimeBotRan;

```

Оголошуємо функції для надсилання повідомлень, налаштування датчиків, обробки оновлень з Telegram та отримання даних з сенсорів.

```

void sendMessageToTelegram(String message);
void setupSensors();
void handleTelegramUpdates();
String getDataFromSensors();
float getSoilMoisture();
String getSoilMoistureStatus();

```

Функція `handleTelegramUpdates` відповідає за обробку оновлень повідомлень від Telegram-бота. Вона перевіряє нові команди, отримані від користувача, і в залежності від отриманих команд виконує відповідні дії.

Функція починається з виклику методу `bot.getUpdates(lastUpdateId + 1)`, який отримує нові оновлення (повідомлення) після останнього обробленого повідомлення, визначеного змінною `lastUpdateId`. Якщо є нові оновлення (повідомлення), вони обробляються в циклі `for`.

```

void handleTelegramUpdates() {
    int updates = bot.getUpdates(lastUpdateId + 1);

    if (updates > 0) {
        for (int i = 0; i < updates; i++) {
            String command = bot.messages[i].text;
            lastUpdateId = bot.messages[i].update_id;

            Serial.println("Received command: " + command);
            if (command == "/get_data") {
                String message = getDataFromSensors();
                sendMessageToTelegram(message);
            }
        }
    }
}

```



```

    } else if (command.startsWith("/set_light_threshold ")) {
        String thresholdStr = command.substring(20);
        lightThreshold = thresholdStr.toFloat();
        sendMessageToTelegram("Новий поріг освітленості: " + String(lightThreshold) + " люкс");
    } else if (command.startsWith("/set_moisture_threshold ")) {
        String thresholdStr = command.substring(23);
        moistureThreshold = thresholdStr.toFloat();
        sendMessageToTelegram("Новий поріг вологості ґрунту: " + String(moistureThreshold) + " V");
    }
}
} else {
    Serial.println("No new updates");
}
}

```

Для кожного нового повідомлення функція витягує текст команди за допомогою `bot.messages[i].text` та оновлює змінну `lastUpdateId` на поточний ID повідомлення, щоб уникнути повторної обробки одного й того ж повідомлення.

Далі виконується перевірка, яка команда була надіслана користувачем:

- Якщо команда `/get_data`, система викликає функцію `getDataFromSensors()` для отримання даних з датчиків і відправляє їх користувачу через `sendMessageToTelegram()`.
- Якщо команда починається з `/set_light_threshold`, зчитується значення порогу освітленості, яке відправляється користувачем, і змінна `lightThreshold` оновлюється на нове значення. Потім бот повідомляє користувача про зміну порогу освітленості.
- Якщо команда починається з `/set_moisture_threshold`, зчитується значення порогу вологості, яке відправляється користувачем, і змінна

moistureThreshold оновлюється на нове значення. Після цього користувач отримує повідомлення про зміну порогу вологості.

Якщо нових оновлень немає, в серійному моніторі виводиться повідомлення "No new updates". Це дозволяє перевірити, чи є нові команди для обробки.

Функція забезпечує інтерактивну взаємодію з користувачем через Telegram, дозволяючи налаштовувати пороги для освітленості та вологості ґрунту, а також отримувати актуальні дані з датчиків.

Надсилаємо привітання при запуску.

sendMessageToTelegram("Вітаємо, мікроферму було запущено, для отримання інформації з датчиків скористайтесь командою /get_data");

Ключовою особливістю програмного коду для NodeMCU є постійне виконання функції loop, що забезпечує неперервну роботу програми. Функція loop() викликає різні підпрограми для перевірки вологості, освітлення та обробки повідомлень з Telegram.

```
void loop() {
    checkTelegram();
    checkMoisture();
    checkLight();
}
```

На початку коду відбувається налаштування пінів для зчитування даних з датчиків та виводу сигналів на реле. Це критично важливий етап, оскільки він визначає параметри взаємодії з зовнішніми пристроями, такими як датчики вологості, освітленості та реле, що керують насосом або освітленням.

```
void setup() {
    Serial.begin(9600);
    pinMode(moistureSensorPin, INPUT);
    pinMode(lightSensorPin, INPUT);
    pinMode(relayPin, OUTPUT);
}
```

Функція `checkMoisture()` відповідає за зчитування значення з датчика вологості і керування реле для поливу. Якщо ґрунт сухий, реле активується, і відбувається подача води

```
void checkMoisture() {
    int moistureValue = analogRead(moistureSensorPin);
    if (moistureValue < moistureThreshold) {
        digitalWrite(relayPin, HIGH);
    } else {
        digitalWrite(relayPin, LOW);
    }
}
```

Аналогічно, функція `checkLight()` зчитує рівень освітленості та контролює реле освітлення. Якщо рівень освітленості нижчий за заданий поріг, реле вмикається для забезпечення належного освітлення.

```
void checkLight() {
    int lightValue = analogRead(lightSensorPin);
    if (lightValue < lightThreshold) {
        digitalWrite(relayPin, HIGH);
    } else {
        digitalWrite(relayPin, LOW);
    }
}
```

Функція `checkTelegram()` виконує перевірку нових повідомлень у Telegram. Якщо отримано команду, відповідні функції для управління вологою чи освітленням виконуються.

```
void checkTelegram() {
}
```

При отриманні команди відбувається зчитування даних, та відправка їх в Telegram.

```
String message = "Температура повітря: " + String(airTemperature) + " °C\n" +
```

```

        "Вологість: " + String(humidity) + " %\n" +
        "Температура землі: " + String(soilTemperature) + " °C\n" +
        "Освітленість: " + String(light) + " люкс\n" +
        "Вологість ґрунту (аналог): " + String(soilMoisture) + " V\n" +
        "Стан ґрунту (цифровий): " + soilMoistureStatus;

    return message;
}

```

Функція `checkMoisture` відповідає за контроль вологості ґрунту і управління реле поливу. Спочатку вона отримує значення вологості ґрунту через функцію `getSoilMoisture()`. Це значення є аналоговим і вимірюється через датчик вологості ґрунту, підключений до піну `NodeMCU`.

Далі функція виводить отримане значення вологості у серійний монітор для відслідковування поточної ситуації.

```

void checkMoisture() {
    float soilMoisture = getSoilMoisture();
    Serial.print("Аналогове значення вологості: ");
    Serial.println(soilMoisture);
    if (soilMoisture < moistureThreshold) {
        digitalWrite(moistureRelayPin, HIGH);
        Serial.println("Реле поливу увімкнено (вологість нижче порогу)");
    } else {
        digitalWrite(moistureRelayPin, LOW);
        Serial.println("Реле поливу вимкнено (вологість вище порогу)");
    }
}

```

Якщо значення вологості ґрунту менше ніж встановлений поріг, заданий змінною `moistureThreshold`, система вважає, що ґрунт недостатньо вологий і потрібно активувати полив. Для цього на пін `moistureRelayPin` подається високий рівень (`HIGH`), що вмикає реле і запускає насос або інший механізм поливу.

Якщо ж значення вологості перевищує поріг, це означає, що ґрунт має достатній рівень вологи, і реле поливу вимикається шляхом подачі низького рівня (LOW) на пін `moistureRelayPin`. Це припиняє подачу води.

Крім того, у серійному моніторі виводиться інформація про стан реле (увімкнено чи вимкнено) і поточне значення вологості ґрунту. Це дозволяє оператору або системі стежити за станом поливу в реальному часі.

Далі функція `checkLight()` відповідає за перевірку рівня освітленості в приміщенні за допомогою датчика BH1750, який підключений до NodeMCU. Спочатку функція зчитує значення освітленості, використовуючи метод `readLightLevel()` з об'єкта `lightMeter`, який відповідає за роботу з датчиком. Потім це значення порівнюється з заданим порогом освітленості

```
void checkLight() {
  float light = lightMeter.readLightLevel();
  Serial.print("Освітленість: ");
  Serial.println(light);
  if (light < lightThreshold) {
    digitalWrite(lightRelayPin, LOW);
    Serial.println("Реле освітлення увімкнено (освітленість нижче порогу)");
  } else {
    digitalWrite(lightRelayPin, HIGH);
    Serial.println("Реле освітлення вимкнено (освітленість вище порогу)");
  }
}
```

Якщо отримане значення освітленості менше заданого порогу (тобто освітленість занадто низька), вмикається реле освітлення. Це досягається подачею низького рівня на пін `lightRelayPin`, що активує реле і, вмикає лампу чи інше освітлювальне обладнання.

Якщо ж рівень освітленості вищий за поріг, реле вимикається шляхом подачі високого рівня на той самий пін `lightRelayPin`. Це призводить до того, що лампа або інше освітлення вимикається.

Усі дії фіксуються у серійному моніторі, що дозволяє відслідковувати, що відбувається в системі, виводячи значення освітленості та стан реле (увімкнено чи вимкнено). Така перевірка відбувається регулярно у циклі роботи програми, що дозволяє автоматично регулювати освітлення відповідно до зміни умов навколишнього середовища.

3.4 Результати тестування системи контролю мікроферми

Після завершення розробки та налаштування системи контролю на базі NodeMCU, було проведено всебічне тестування її функціональності. Тестування охоплювало всі аспекти системи, включаючи підключення до Wi-Fi, інтеграцію з Telegram, управління реле, а також роботу датчиків та управління освітленням та поливом..

Система успішно підключилася до мережі Wi-Fi без жодних збоїв. Ми перевірили стабільність з'єднання, і виявили, що сигнал залишається стабільним навіть на відстані.

Telegram-бот був налаштований успішно, і ми отримали підтвердження відправлення повідомлень. Система може отримувати команди з чату, наприклад, `/get_data`, що дозволяє користувачеві отримувати актуальну інформацію про стан мікроферми, команда для керування світлом `/set_light_threshold` та для керування вологою - `/set_moisture_threshold` .(Рис. 3.2).

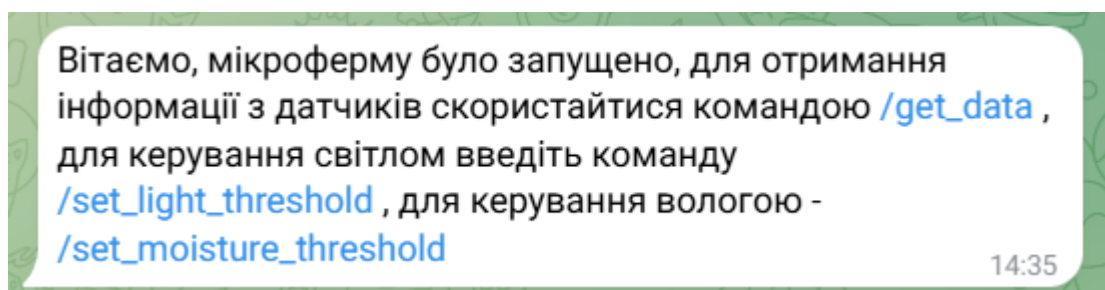


Рисунок 3.2 – Успішне отримання повідомлення про запуск

Тим часом змін на фермі не відбувається, так як при включенні реле вимкнуті, та показники не виставлені(Рис. 3.3).



Рисунок 3.3 – Ввімкнення мікроферми

Для початку переглянемо показники з датчиків, для цього скористуюся командою `/get_data`, контроллер збирає всі показники та надсилає данні майже миттєво(Рис. 3.4).

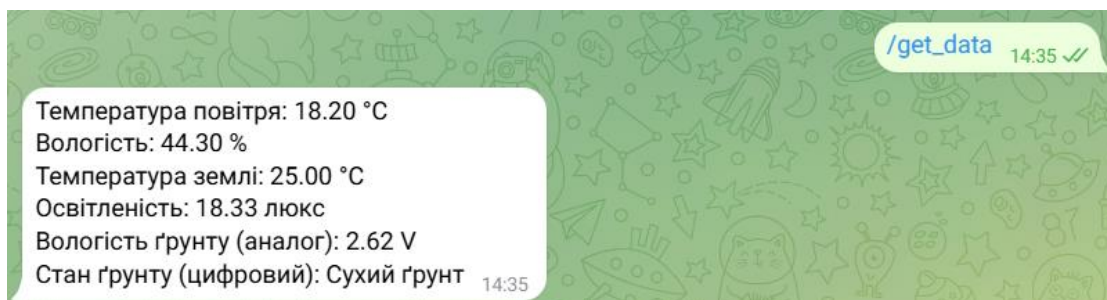


Рисунок 3.4 – Використання команди /get_data

Далі протестуємо реле освітлення, з інформації на рисунку 3.3 випливає що освітленість 18.33 люкса, поставимо командою /set_light_threshold ліміт на 20, таким чином у нас освітленість буде менша чим поріг, тому реле активується, та лампочка загориться. Також має прийти повідомлення про зміну порогу освітлення(Рис. 3.5).



Рисунок 3.5 – Отримання порогу освітленості

Після введення порогу реле активувалось та лампочка загорілась(Рис. 3.6).

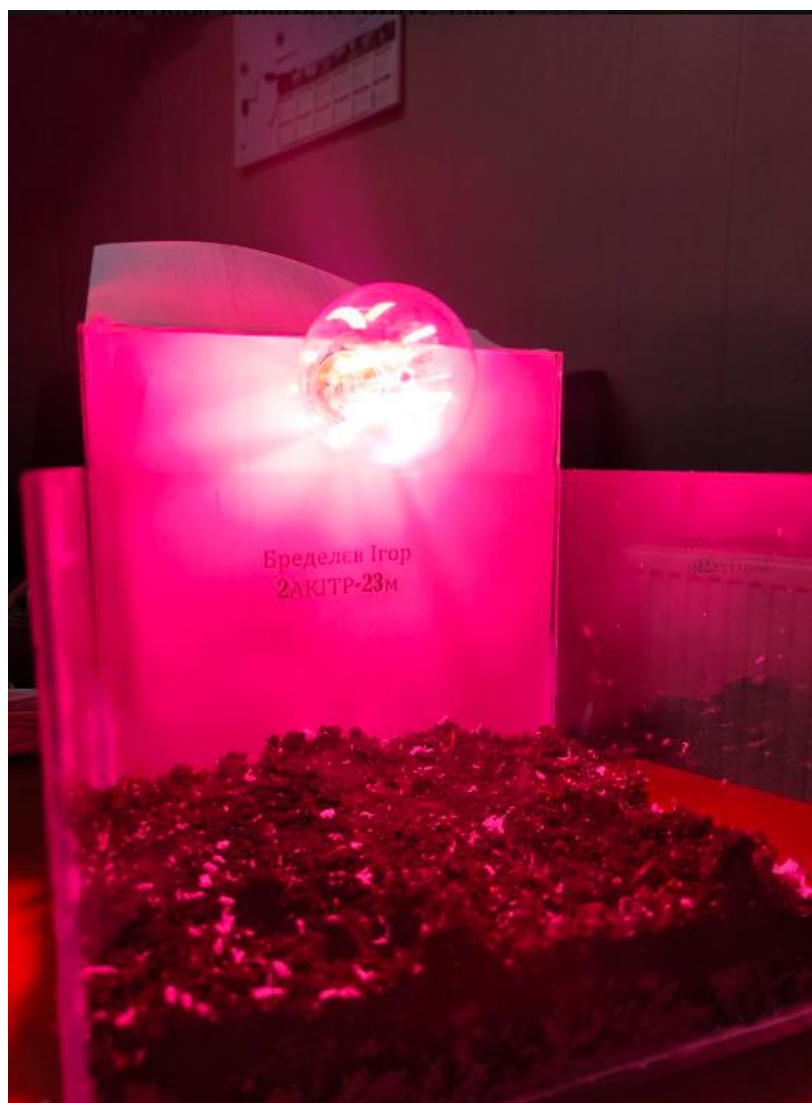


Рисунок 3.6 – Активація лампочки

Система управління освітленням працює за заданим алгоритмом: реле освітлення вмикається при освітленості нижче заданого порогу. Під час тестування освітленість була змінена, і система правильно реагувала на зміни. На фотографіях видно, як реле вмикається в умовах низької освітленості.

Далі протестуємо систему поливу з інформації на рисунку 3.3 впливає що вологість 2.62, поставимо командою `set_moisture_threshold 2.2`, після введення команди був встановлений новий ліміт(Рис. 3.7).



Рисунок 3.7 – Встановлення порогу вологості.

Після введення команди реле поливу ввімкнулось, та почався полив, щоб краще було видно, шланг дістали з землі на поверхню(Рис. 3.8).

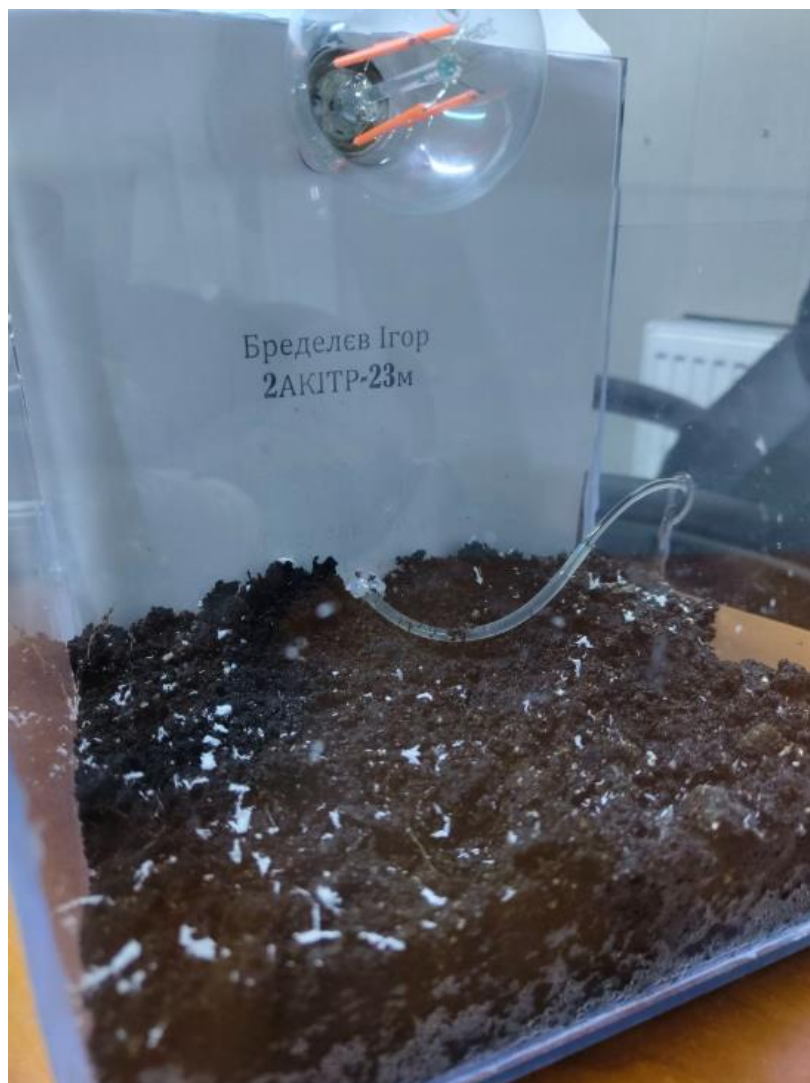


Рисунок 3.8 – Активація поливу.

Полив триває до тих пір поки значення з датчика не буде менше чим поріг, після вимкнення насоса перевіряємо вологість командою `/get_data`(Рис. 3.9).

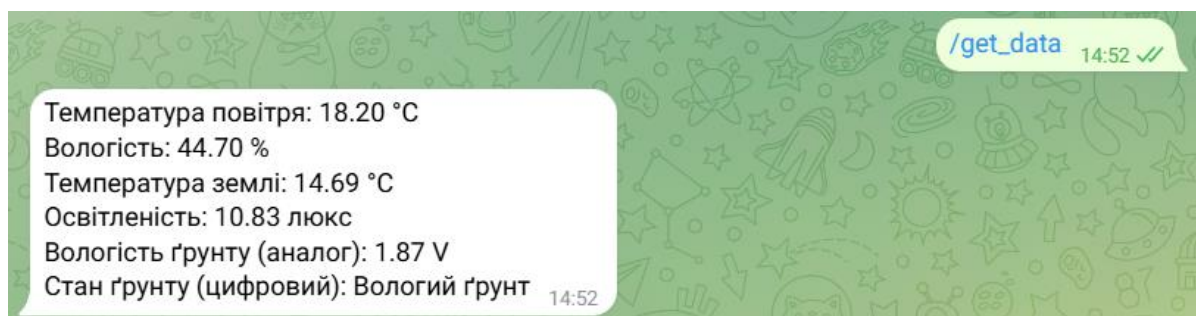


Рисунок 3.9 – Перевірка вологості

З перевірок вище можна зрозуміти що все працює як потрібно, на кінець перевіримо обидві команди одночасно, так як нова вологість ґрунту з рисунку 3.9 - 1.87, установимо ще менший ліміт на 1.6, освітленість поставимо на 15, так як освітленість в кімнаті 10.83 люкса(Рис. 3.10).

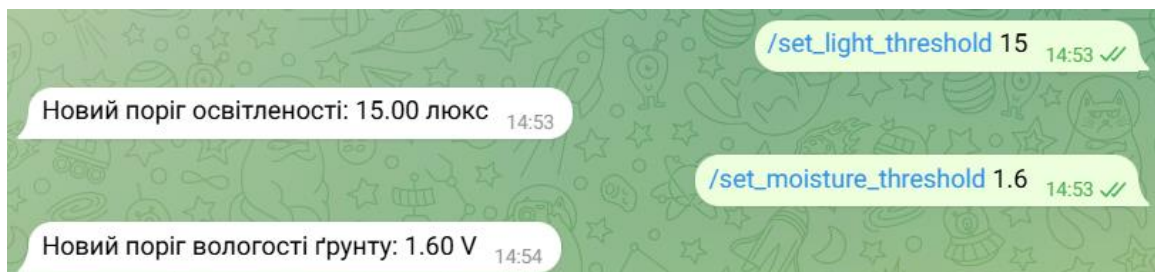


Рисунок 3.10 – Перевірка команд

Після оновлення порогів зразу увімкнулись реле, та спрацювало освітлення та полив(Рис. 3.11).

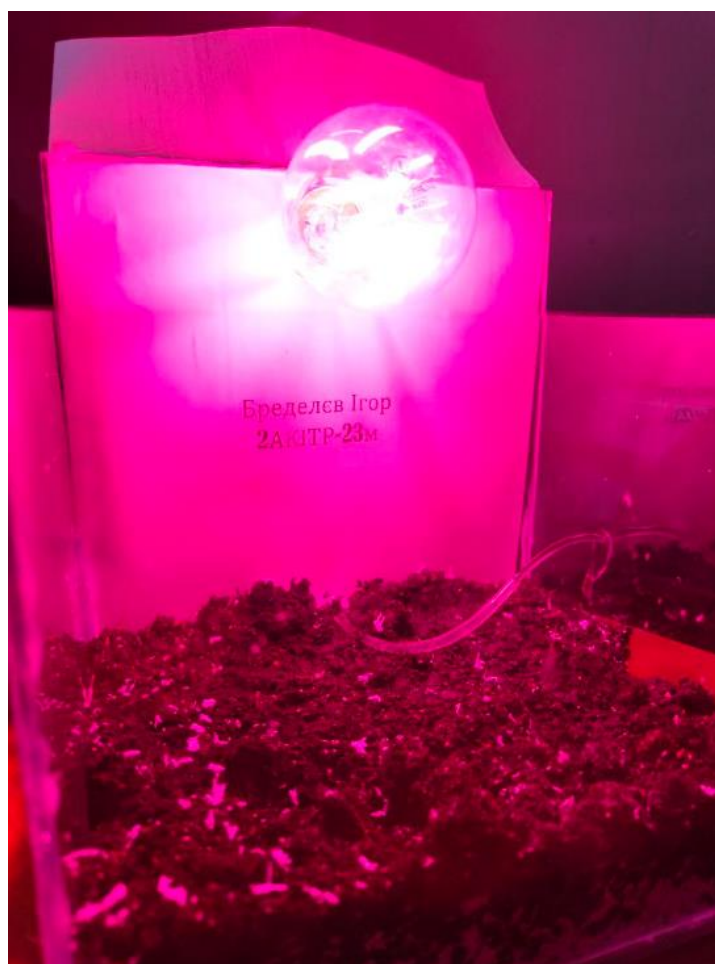


Рисунок 3.11 – Перевірка освітлення та поливу

Після того як вологість впала реле поливу виключилось, лампочка продовжила горіти, перевіримо яка у нас вологість ґрунту командою /get_data(3.12). Після чого бачимо що вологість справді впала нижче значення 1.6, аж до 1.48.

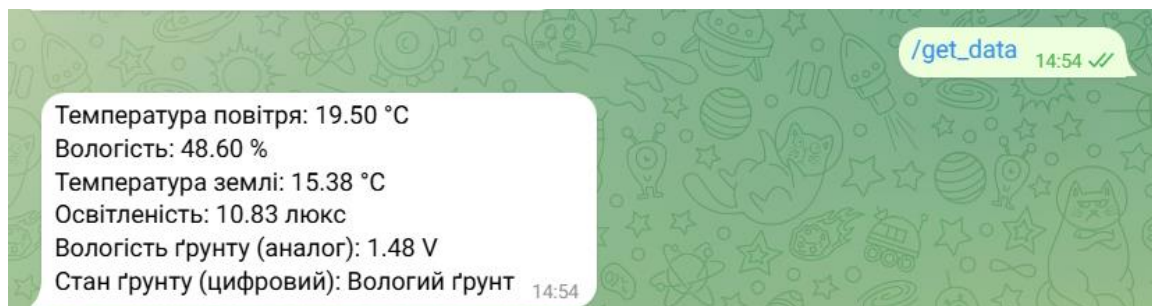


Рисунок 3.12 – Перевірка вологості

Усі дані були актуальними та відповідали умовам тестування. Всі компоненти системи, включаючи датчики, реле, Wi-Fi модуль і Telegram-бот, продемонстрували стабільну та надійну роботу. Тестування показало, що система може ефективно управляти мікрокліматом, автоматично реагуючи на зміни навколишнього середовища.

3.5 Висновки до розділу

Система контролю мікроклімату, побудована на базі NodeMCU, виявилася успішною в реалізації основних функцій. Підключення до Wi-Fi та інтеграція з Telegram дозволили нам отримувати актуальні дані. Використані датчики, такі як DHT22, DS18B20, BH1750 і DA136, показали стабільні результати, забезпечуючи точні вимірювання температури, вологості повітря та ґрунту, а також рівня освітленості.

Система автоматично реагує на зміни навколишнього середовища, контролюючи полив та освітлення через реле. Тестування підтвердило належну роботу всіх компонентів, а результати зафіксовані на фотографіях, які ілюструють процеси поливу та активації освітлення, насоса. Це дозволяє значно спростити догляд за росл

нами, економлячи час і зусилля. У подальшому можливі покращення, такі як додавання графічного інтерфейсу або інтеграція з іншими системами автоматизації, що підвищить функціональність і зручність використання системи.

4 ЕКОНОМІЧНА ЧАСТИНА

Науково-технічна розробка має право на існування та впровадження, якщо вона відповідає вимогам часу, як в напрямку науково-технічного прогресу та і в плані економіки. Тому для науково-дослідної роботи необхідно оцінювати економічну ефективність результатів виконаної роботи.

Магістерська кваліфікаційна робота з розробки та дослідження «Комп'ютерно-інтегрований інтелектуальний інтерфейс для управління мікрофермою» відноситься до науково-технічних робіт, які орієнтовані на виведення на ринок (або рішення про виведення науково-технічної розробки на ринок може бути прийнято у процесі проведення самої роботи), тобто коли відбувається так звана комерціалізація науково-технічної розробки. Цей напрямок є пріоритетним, оскільки результатами розробки можуть користуватися інші споживачі, отримуючи при цьому певний економічний ефект. Але для цього потрібно знайти потенційного інвестора, який би взявся за реалізацію цього проекту і переконати його в економічній доцільності такого кроку.

4.1 Проведення комерційного та технологічного аудиту науково-технічної розробки

Метою проведення комерційного і технологічного аудиту дослідження за темою «Комп'ютерно-інтегрований інтелектуальний інтерфейс для управління мікрофермою» є оцінювання науково-технічного рівня та рівня комерційного потенціалу розробки, створеної в результаті науково-технічної діяльності.

Оцінювання науково-технічного рівня розробки та її комерційного потенціалу рекомендується здійснювати із застосуванням 5-ти бальної системи оцінювання за 12-ма критеріями, наведеними в табл. 4.1 [26].

Таблиця 4.1 – Рекомендовані критерії оцінювання науково-технічного рівня і комерційного потенціалу розробки та бальна оцінка

Бали (за 5-ти бальною шкалою)					
	0	1	2	3	4

Продовження таблиці 4.1

Технічна здійсненність концепції					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено працездатність продукту в реальних умовах
Ринкові переваги (недоліки)					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів
4	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів	Технічні та споживчі властивості продукту значно кращі, ніж в аналогів
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
Ринкові перспективи					
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкурентів немає
Практична здійсненність					
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витратити значні кошти та час на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування

Продовження таблиці 4.1

10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років
12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту

Результати оцінювання науково-технічного рівня та комерційного потенціалу науково-технічної розробки потрібно звести до таблиці.

Таблиця 4.2 – Результати оцінювання науково-технічного рівня і комерційного потенціалу розробки експертами

Критерії	Експерт (ПІБ, посада)		
	1	2	3
	Бали:		
1. Технічна здійсненність концепції	5	5	5
2. Ринкові переваги (наявність аналогів)	2	3	2
3. Ринкові переваги (ціна продукту)	4	5	3
4. Ринкові переваги (технічні властивості)	3	3	3
5. Ринкові переваги (експлуатаційні витрати)	2	2	3
6. Ринкові перспективи (розмір ринку)	3	3	3
7. Ринкові перспективи (конкуренція)	2	2	2
8. Практична здійсненність (наявність фахівців)	4	4	4

Продовження таблиці 4.2

9. Практична здійсненність (наявність фінансів)	2	3	2
10. Практична здійсненність (необхідність нових матеріалів)	3	4	4
11. Практична здійсненність (термін реалізації)	3	4	4
12. Практична здійсненність (розробка документів)	4	4	3
Сума балів	37	42	38
Середньоарифметична сума балів CB_c	39,0		

За результатами розрахунків, наведених в таблиці 4.2, зробимо висновок щодо науково-технічного рівня і рівня комерційного потенціалу розробки. При цьому використаємо рекомендації, наведені в табл. 4.3 [25].

Таблиця 4.3 – Науково-технічні рівні та комерційні потенціали розробки

Середньоарифметична сума балів СБ , розрахована на основі висновків експертів	Науково-технічний рівень та комерційний потенціал розробки
41...48	Високий
31...40	Вище середнього
21...30	Середній
11...20	Нижче середнього
0...10	Низький

Згідно проведених досліджень рівень комерційного потенціалу розробки за темою «Комп'ютерно-інтегрований інтелектуальний інтерфейс для управління мікрофермою» становить 39,0 бала, що, відповідно до таблиці 4.3, свідчить про комерційну важливість проведення даних досліджень (рівень комерційного потенціалу розробки вище середнього).

4.2 Розрахунок узагальненого коефіцієнта якості розробки

Окрім комерційного аудиту розробки доцільно також розглянути технічний рівень якості розробки, розглянувши її основні технічні показники. Ці показники по-різному впливають на загальну якість проектної розробки.

Узагальнений коефіцієнт якості (B_n) для нового технічного рішення розрахуємо за формулою [26]:

$$B_n = \sum_{i=1}^k \alpha_i \cdot \beta_i, \quad (4.1)$$

де k – кількість найбільш важливих технічних показників, які впливають на якість нового технічного рішення;

α_i – коефіцієнт, який враховує питому вагу i -го технічного показника в загальній якості розробки. Коефіцієнт α_i визначається експертним шляхом і при цьому

має виконуватись умова $\sum_{i=1}^k \alpha_i = 1$;

β_i – відносне значення i -го технічного показника якості нової розробки.

Відносні значення β_i для різних випадків розраховуємо за такими формулами:

- для показників, зростання яких вказує на підвищення в лінійній залежності якості нової розробки:

$$\beta_i = \frac{I_{ni}}{I_{ai}}, \quad (4.2)$$

де I_{ni} та I_{na} – чисельні значення конкретного i -го технічного показника якості відповідно для нової розробки та аналога;

- для показників, зростання яких вказує на погіршення в лінійній залежності якості нової розробки:

$$\beta_i = \frac{I_{ai}}{I_{ni}}; \quad (4.3)$$

Використовуючи наведені залежності можемо проаналізувати та порівняти техніко-економічні характеристики аналогу та розробки на основі отриманих наявних та проектних показників, а результати порівняння зведемо до таблиці 4.4.

Таблиця 4.4 – Порівняння основних параметрів розробки та аналога.

Показники (параметри)	Одиниця вимірю- вання	Аналог	Проектований пристрій	Відношення параметрів нової розро- бки до ана- лога	Питома вага показника
--------------------------	-----------------------------	--------	--------------------------	---	--------------------------

Продовження таблиці 4.4

1 Точність регулювання параметрів	%	85	95	1,12	0,25
2 Кількість параметрів регулювання клімату	од	2	5	2,5	0,15
3 Споживана потужність блоку контролю	Вт	80	20	4	0,25
4 Зручність переналаштування	бал	10	8	0,8	0,2
5 Зручність контролю мікроферми	бал	5	10	2	0,15

Узагальнений коефіцієнт якості (B_n) для нового технічного рішення складе:

$$B_n = \sum_{i=1}^k \alpha_i \cdot \beta_i = 1,12 \cdot 0,25 + 2,5 \cdot 0,15 + 4 \cdot 0,25 + 0,8 \cdot 0,2 + 2 \cdot 0,15 = 2,12.$$

Отже за технічними параметрами, згідно узагальненого коефіцієнту якості розробки, науково-технічна розробка переважає існуючі аналоги приблизно в 2,12 рази.

4.3 Розрахунок витрат на проведення науково-дослідної роботи

Витрати, пов'язані з проведенням науково-дослідної роботи на тему «Комп'ютерно-інтегрований інтелектуальний інтерфейс для управління мікрофермою», під час планування, обліку і калькулювання собівартості науково-дослідної роботи групуємо за відповідними статтями.

4.3.1 Витрати на оплату праці

До статті «Витрати на оплату праці» належать витрати на виплату основної та додаткової заробітної плати керівникам відділів, лабораторій, секторів і груп, науковим, інженерно-технічним працівникам, конструкторам, технологам, креслярам, ко-

пiювальникам, лаборантам, робiтникам, студентам, аспiрантам та iншим працiвникам, безпосередньо зайнятим виконанням конкретної теми, обчисленої за посадовими окладами, вiдрядними розцiнками, тарифними ставками згiдно з чинними в органiзацiях системами оплати працi.

Основна заробiтна плата дослiдникiв

Витрати на основну заробiтну плату дослiдникiв (Z_o) розраховуємо у вiдповiдностi до посадових окладiв працiвникiв, за формулою [25]

$$Z_o = \sum_{i=1}^k M_{ni} \cdot t_i / T_p, \quad (4.4)$$

де k – кiлькiсть посад дослiдникiв залучених до процесу дослiджень;

M_{ni} – мiсячний посадовий оклад конкретного дослiдника, грн;

t_i – число днiв роботи конкретного дослiдника, дн.;

T_p – середнє число робочих днiв в мiсяцi, $T_p=21$ днi.

$$Z_o = 18300,00 \cdot 15 / 21 = 13071,43 \text{ грн.}$$

Проведенi розрахунки зведемо до таблицi.

Таблиця 4.5 – Витрати на заробiтну плату дослiдникiв

Найменування посади	Мiсячний посадовий оклад, грн	Оплата за робочий день, грн	Число днiв роботи	Витрати на заробiтну плату, грн
Керiвник	18300,00	871,43	15	13071,43
Інженер-розробник iнтегрованого iнтелектуального iнтерфейсу	16200,00	771,43	15	11571,43
Фахiвець-консультант (агроном)	16000,00	761,90	5	3809,52
Лаборант	8050,00	383,33	15	5750,00
Всього				34202,38

Основна заробiтна плата робiтникiв

Витрати на основну заробiтну плату робiтникiв (Z_p) за вiдповiдними найменуваннями робiт НДР на тему «Комп'ютерно-iнтегрований iнтелектуальний iнтерфейс для управлiння мiкрофермою» розраховуємо за формулою:

$$Z_p = \sum_{i=1}^n C_i \cdot t_i, \quad (4.5)$$

де C_i – погодинна тарифна ставка робітника відповідного розряду, за виконану відповідну роботу, грн/год;

t_i – час роботи робітника при виконанні визначеної роботи, год.

Погодинну тарифну ставку робітника відповідного розряду C_i можна визначити за формулою:

$$C_i = \frac{M_M \cdot K_i \cdot K_c}{T_p \cdot t_{зм}}, \quad (4.6)$$

де M_M – розмір прожиткового мінімуму працездатної особи, або мінімальної місячної заробітної плати (в залежності від діючого законодавства), прийmemo $M_M=8000,00$ грн;

K_i – коефіцієнт міжкваліфікаційного співвідношення для встановлення тарифної ставки робітнику відповідного розряду (табл. Б.2, додаток Б) [25];

K_c – мінімальний коефіцієнт співвідношень місячних тарифних ставок робітників першого розряду з нормальними умовами праці виробничих об'єднань і підприємств до законодавчо встановленого розміру мінімальної заробітної плати.

T_p – середнє число робочих днів в місяці, приблизно $T_p = 21$ дн;

$t_{зм}$ – тривалість зміни, год.

$$C_1 = 8000,00 \cdot 1,35 \cdot 1,15 / (21 \cdot 8) = 73,93 \text{ грн.}$$

$$З_{рл} = 73,93 \cdot 4,80 = 354,86 \text{ грн.}$$

Таблиця 4.6 – Величина витрат на основну заробітну плату робітників

Найменування робіт	Тривалість роботи, год	Розряд роботи	Тарифний коефіцієнт	Погодинна тарифна ставка, грн	Величина оплати на робітника грн
Підготовка робочого місця розробника інтелектуального інтерфейсу	4,80	3	1,35	73,93	354,86
Підготовка імітаційного обладнання мікроферми	5,50	4	1,50	82,14	451,79
Формування бази даних моделей впливу	10,00	3	1,35	73,93	739,29
Монтаж імітаційного обладнання мікроферми	6,25	4	1,50	82,14	513,39

Продовження таблиці 4.6

Налагодження обладнання	2,00	2	1,10	60,24	120,48
Фіксація результатів експерименту	4,00	1	1,00	54,76	219,05
Всього					2398,85

Додаткова заробітна плата дослідників та робітників

Додаткову заробітну плату розраховуємо як 10 ... 12% від суми основної заробітної плати дослідників та робітників за формулою:

$$Z_{\text{доп}} = (Z_o + Z_p) \cdot \frac{H_{\text{доп}}}{100\%}, \quad (4.7)$$

де $H_{\text{доп}}$ – норма нарахування додаткової заробітної плати. Прийmemo 11%.

$$Z_{\text{доп}} = (34202,38 + 2398,85) \cdot 11 / 100\% = 4026,13 \text{ грн.}$$

4.3.2 Відрахування на соціальні заходи

Нарахування на заробітну плату дослідників та робітників розраховуємо як 22% від суми основної та додаткової заробітної плати дослідників і робітників за формулою:

$$Z_n = (Z_o + Z_p + Z_{\text{доп}}) \cdot \frac{H_n}{100\%} \quad (4.8)$$

де H_n – норма нарахування на заробітну плату. Приймаємо 22%.

$$Z_n = (34202,38 + 2398,85 + 4026,13) \cdot 22 / 100\% = 8938,02 \text{ грн.}$$

4.3.3 Сировина та матеріали

До статті «Сировина та матеріали» належать витрати на сировину, основні та допоміжні матеріали, інструменти, пристрої та інші засоби і предмети праці, які придбані у сторонніх підприємств, установ і організацій та витрачені на проведення досліджень за темою «Комп'ютерно-інтегрований інтелектуальний інтерфейс для управління мікрофермою».

Витрати на матеріали (M), у вартісному вираженні розраховуються окремо по кожному виду матеріалів за формулою:

$$M = \sum_{j=1}^n H_j \cdot C_j \cdot K_j - \sum_{j=1}^n B_j \cdot C_{ej}, \quad (4.9)$$

де H_j – норма витрат матеріалу j -го найменування, кг;

n – кількість видів матеріалів;

C_j – вартість матеріалу j -го найменування, грн/кг;

K_j – коефіцієнт транспортних витрат, ($K_j = 1,1 \dots 1,15$);

B_j – маса відходів j -го найменування, кг;

C_{ej} – вартість відходів j -го найменування, грн/кг.

$$M_1 = 4,0 \cdot 196,00 \cdot 1,05 - 0 \cdot 0 = 823,20 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці.

Таблиця 4.7 – Витрати на матеріали

Найменування матеріалу, марка, тип, сорт	Ціна за 1 кг, грн	Норма витрат, кг	Величина відходів, кг	Ціна відходів, грн/кг	Вартість витраченого матеріалу, грн
Папір офісний	196,00	4,0	0	0	823,20
Приладдя канцелярське	106,00	4,0	0	0	445,20
Тонер для картриджа Canon LaserIQ	185,00	1,0	0	0	194,25
Папір для записів	98,00	3,0	0	0	308,70
Органайзер	201,00	1,00	0	0	211,05
Диск оптичний	28,00	4,0	0	0	117,60
USB-пам'ять Kingston 64 GB	239,00	1,0	0	0	250,95
Тека для паперів	99,00	4,000	0	0	415,80
Наповнювач (імітатор ґрунту)	42,00	3,000	0	0	132,30
Термоусадка	150,00	0,500	0	0	78,75
Набір кабелів	64,00	1,000	0	0	67,20
Припій	652,00	0,010	0	0	6,85
Інші матеріали	120,00	1,000	0,000	0,00	126,00
Всього					3177,85

4.3.4 Розрахунок витрат на комплектуючі

Витрати на комплектуючі (K_{ϵ}), які використовують при проведенні НДР на тему «Комп'ютерно-інтегрований інтелектуальний інтерфейс для управління мікроферою», розраховуємо, згідно з їхньою номенклатурою, за формулою:

$$K_{\epsilon} = \sum_{j=1}^n H_j \cdot C_j \cdot K_j \quad (4.10)$$

де H_j – кількість комплектуючих j -го виду, шт.;

C_j – покупна ціна комплектуючих j -го виду, грн;

K_j – коефіцієнт транспортних витрат, ($K_j = 1,1 \dots 1,15$).

$K_{\epsilon} = 1 \cdot 360,00 \cdot 1,05 = 378,00$ грн.

Проведені розрахунки зведемо до таблиці.

Таблиця 4.8 – Витрати на комплектуючі

Найменування комплектуючих	Кількість, шт.	Ціна за штуку, грн	Сума, грн
Контролер NodeMCU	1	360,00	378,00
Датчик освітлення BH1750	1	58,00	60,90
Датчик вологості ґрунту LM393	1	42,00	44,10
Датчик вологості та температури повітря DHT22	1	48,00	50,40
Датчик температури землі DS18B20	1	25,00	26,25
Модуль реле	3	55,00	173,25
Погрузний насос 5в	1	120,00	126,00
Лампа (фітокомпонентна)	1	164,00	172,20
Вентилятор	1	45,00	47,25
Всього			1078,35

4.3.5 Спецустаткування для наукових (експериментальних) робіт

До статті «Спецустаткування для наукових (експериментальних) робіт» належать витрати на виготовлення та придбання спецустаткування необхідного для проведення досліджень, також витрати на їх проектування, виготовлення, транспортування, монтаж та встановлення. Витрати за статтею відсутні.

4.3.6 Програмне забезпечення для наукових (експериментальних) робіт

До статті «Програмне забезпечення для наукових (експериментальних) робіт» належать витрати на розробку та придбання спеціальних програмних засобів і програмного забезпечення, (програм, алгоритмів, баз даних) необхідних для проведення досліджень, також витрати на їх проектування, формування та встановлення.

Балансову вартість програмного забезпечення розраховуємо за формулою:

$$B_{npz} = \sum_{i=1}^k C_{inprz} \cdot C_{npz.i} \cdot K_i, \quad (4.11)$$

де C_{inprz} – ціна придбання одиниці програмного засобу даного виду, грн;

$C_{npz.i}$ – кількість одиниць програмного забезпечення відповідного найменування, які придбані для проведення досліджень, шт.;

K_i – коефіцієнт, що враховує інсталяцію, налагодження програмного засобу тощо, ($K_i = 1,10 \dots 1,12$);

k – кількість найменувань програмних засобів.

$$B_{npz} = 9220,00 \cdot 1 \cdot 1,05 = 9681,00 \text{ грн.}$$

Отримані результати зведемо до таблиці:

Таблиця 4.9 – Витрати на придбання програмних засобів по кожному виду

Найменування програмного засобу	Кількість, шт	Ціна за одиницю, грн	Вартість, грн
Середовище моделювання Proteus (ISIS Proteus)	1	9220,00	9681,00
Arduino IDE	1	1350,00	1417,50

Продовження таблиці 4.9

Arduino IDE	1	1350,00	1417,50
Абонентна плата доступу до мережі Internet (1 місяць)	1	325,00	341,25
Всього			11439,75

4.3.7 Амортизація обладнання, програмних засобів та приміщень

В спрощеному вигляді амортизаційні відрахування по кожному виду обладнання, приміщень та програмному забезпеченню тощо, розраховуємо з використанням прямолінійного методу амортизації за формулою:

$$A_{обл} = \frac{Ц_{б}}{T_{е}} \cdot \frac{t_{вик}}{12}, \quad (4.12)$$

де $Ц_{б}$ – балансова вартість обладнання, програмних засобів, приміщень тощо, які використовувались для проведення досліджень, грн;

$t_{вик}$ – термін використання обладнання, програмних засобів, приміщень під час досліджень, місяців;

$T_{е}$ – строк корисного використання обладнання, програмних засобів, приміщень тощо, років.

$$A_{обл} = (26355,00 \cdot 1) / (2 \cdot 12) = 1098,13 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці.

Таблиця 4.10 – Амортизаційні відрахування по кожному виду обладнання

Найменування обладнання	Балансова вартість, грн	Строк корисного використання, років	Термін використання обладнання, місяців	Амортизаційні відрахування, грн
Персональний комп'ютер розробника програмного забезпечення	26355,00	2	1	1098,13
Робоче місце дослідника	7880,00	5	1	131,33
Пристрої виводу інформації	6875,00	4	1	143,23
Оргтехніка	8675,00	4	1	180,73

Продовження таблиці 4.9

Приміщення лабораторії	299000,00	25	1	996,67
ОС Windows 10	6520,00	2	1	271,67
Прикладний Офісний пакет Microsoft Office 2016	6540,00	2	1	272,50
Всього				3094,25

4.3.8 Паливо та енергія для науково-виробничих цілей

Витрати на силову електроенергію (B_e) розраховуємо за формулою:

$$B_e = \sum_{i=1}^n \frac{W_{yi} \cdot t_i \cdot C_e \cdot K_{eni}}{\eta_i}, \quad (4.13)$$

де W_{yi} – встановлена потужність обладнання на визначеному етапі розробки, кВт;

t_i – тривалість роботи обладнання на етапі дослідження, год;

C_e – вартість 1 кВт-години електроенергії, грн; (вартість електроенергії визначається за даними енергопостачальної компанії), прийmemo $C_e = 10,98$ грн;

K_{eni} – коефіцієнт, що враховує використання потужності, $K_{eni} < 1$;

η_i – коефіцієнт корисної дії обладнання, $\eta_i < 1$.

$$B_e = 0,35 \cdot 120,0 \cdot 10,98 \cdot 0,95 / 0,97 = 461,16 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці.

Таблиця 4.11 – Витрати на електроенергію

Найменування обладнання	Встановлена потужність, кВт	Тривалість роботи, год	Сума, грн
Персональний комп'ютер розробника програмного забезпечення	0,35	120,0	461,16
Робоче місце дослідника	0,25	120,0	329,40
Пристрої виводу інформації	0,12	10,0	13,18
Оргтехніка	0,10	2,0	2,20

Продовження таблиці 4.11

Обладнання моделювання мікроферми	0,10	100,0	109,80
Всього			915,73

4.3.9 Службові відрядження

Витрати за статтею «Службові відрядження» відсутні.

4.3.10 Витрати на роботи, які виконують сторонні підприємства, установи і організації

Витрати за статтею «Витрати на роботи, які виконують сторонні підприємства, установи і організації» відсутні.

4.3.11 Інші витрати

До статті «Інші витрати» належать витрати, які не знайшли відображення у зазначених статтях витрат і можуть бути віднесені безпосередньо на собівартість досліджень за прямими ознаками.

Витрати за статтею «Інші витрати» розраховуємо як 50...100% від суми основної заробітної плати дослідників та робітників за формулою:

$$I_{\text{в}} = (Z_{\text{о}} + Z_{\text{р}}) \cdot \frac{H_{\text{ів}}}{100\%}, \quad (4.14)$$

де $H_{\text{ів}}$ – норма нарахування за статтею «Інші витрати», прийmemo $H_{\text{ів}} = 55\%$.

$$I_{\text{в}} = (34202,38 + 2398,85) \cdot 55 / 100\% = 20130,67 \text{ грн.}$$

4.3.12 Накладні (загальновиробничі) витрати

До статті «Накладні (загальновиробничі) витрати» належать: витрати, пов'язані з управлінням організацією; витрати на винахідництво та раціоналізацію; витрати на підготовку (перепідготовку) та навчання кадрів; витрати, пов'язані з набором робочої

сили; витрати на оплату послуг банків; витрати, пов'язані з освоєнням виробництва продукції; витрати на науково-технічну інформацію та рекламу та ін.

Витрати за статтею «Накладні (загальновиробничі) витрати» розраховуємо як 100...150% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{нзв} = (З_o + З_p) \cdot \frac{H_{нзв}}{100\%}, \quad (4.15)$$

де $H_{нзв}$ – норма нарахування за статтею «Накладні (загальновиробничі) витрати», прийmemo $H_{нзв} = 100\%$.

$$B_{нзв} = (34202,38 + 2398,85) \cdot 100 / 100\% = 36601,23 \text{ грн.}$$

Витрати на проведення науково-дослідної роботи на тему «Комп'ютерно-інтегрований інтелектуальний інтерфейс для управління мікрофермою» розраховуємо як суму всіх попередніх статей витрат за формулою:

$$B_{заг} = З_o + З_p + З_{дод} + З_n + M + K_{\text{в}} + B_{\text{спец}} + B_{\text{прг}} + A_{\text{обл}} + B_e + B_{\text{св}} + B_{\text{сп}} + I_{\text{в}} + B_{нзв}. \quad (4.16)$$

$$B_{заг} = 34202,38 + 2398,85 + 4026,13 + 8938,02 + 3177,85 + 1078,35 + 0,00 + 11439,75 + 3094,25 + 915,73 + 0,00 + 0,00 + 20130,67 + 36601,23 = 126003,21 \text{ грн.}$$

Загальні витрати $ЗВ$ на завершення науково-дослідної (науково-технічної) роботи та оформлення її результатів розраховується за формулою:

$$ЗВ = \frac{B_{заг}}{\eta}, \quad (4.17)$$

де η - коефіцієнт, який характеризує етап (стадію) виконання науково-дослідної роботи, прийmemo $\eta = 0,9$.

$$ЗВ = 126003,21 / 0,9 = 140003,57 \text{ грн.}$$

4.4 Розрахунок економічної ефективності науково-технічної розробки при її можливій комерціалізації потенційним інвестором

Результати дослідження проведені за темою «Комп'ютерно-інтегрований інтелектуальний інтерфейс для управління мікрофермою» передбачають комерціалізацію протягом 4-х років реалізації на ринку.

В цьому випадку майбутній економічний ефект буде формуватися на основі таких даних:

ΔN – збільшення кількості споживачів пристрою, у періоди часу, що аналізуються, від покращення його певних характеристик;

Показник	1-й рік	2-й рік	3-й рік	4-й рік
Збільшення кількості споживачів, осіб	200	400	450	400

N – кількість споживачів які використовували аналогічний пристрій у році до впровадження результатів нової науково-технічної розробки, прийmemo 5200 осіб;

C_o – вартість пристрою у році до впровадження результатів розробки, прийmemo 500,00 грн;

$\pm \Delta C_o$ – зміна вартості пристрою від впровадження результатів науково-технічної розробки, прийmemo 162,50 грн.

Можливе збільшення чистого прибутку у потенційного інвестора $\Delta \Pi_i$ для кожного із 4-х років, протягом яких очікується отримання позитивних результатів від можливого впровадження та комерціалізації науково-технічної розробки, розраховуємо за формулою [25].

$$\Delta \Pi_i = (\pm \Delta C_o \cdot N + C_o \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot \left(1 - \frac{\vartheta}{100}\right), \quad (4.18)$$

де λ – коефіцієнт, який враховує сплату потенційним інвестором податку на додану вартість. У 2024 році ставка податку на додану вартість складає 20%, а коефіцієнт $\lambda = 0,8333$;

ρ – коефіцієнт, який враховує рентабельність інноваційного продукту). Приймемо $\rho = 42\%$;

ϑ – ставка податку на прибуток, який має сплачувати потенційний інвестор, у 2024 році $\vartheta = 18\%$;

Збільшення чистого прибутку 1-го року:

$$\Delta \Pi_1 = (162,50 \cdot 5200,00 + 662,50 \cdot 200) \cdot 0,83 \cdot 0,42 \cdot (1 - 0,18/100\%) = 279420,33 \text{ грн.}$$

Збільшення чистого прибутку 2-го року:

$$\Delta\Pi_2 = (162,50 \cdot 5200,00 + 662,50 \cdot 600) \cdot 0,83 \cdot 0,42 \cdot (1 - 0,18/100\%) = 355171,11 \text{ грн.}$$

Збільшення чистого прибутку 3-го року:

$$\Delta\Pi_3 = (162,50 \cdot 5200,00 + 662,50 \cdot 1050) \cdot 0,83 \cdot 0,42 \cdot (1 - 0,18/100\%) = 440390,74 \text{ грн.}$$

Збільшення чистого прибутку 4-го року:

$$\Delta\Pi_4 = (162,50 \cdot 5200,00 + 662,50 \cdot 1450) \cdot 0,83 \cdot 0,42 \cdot (1 - 0,18/100\%) = 516141,52 \text{ грн.}$$

Приведена вартість збільшення всіх чистих прибутків $\Pi\Pi$, що їх може отримати потенційний інвестор від можливого впровадження та комерціалізації науково-технічної розробки:

$$\Pi\Pi = \sum_{i=1}^T \frac{\Delta\Pi_i}{(1+\tau)^t}, \quad (4.19)$$

$$\begin{aligned} \Pi\Pi &= 279420,33/(1+0,25)^1 + 355171,11/(1+0,25)^2 + 440390,74/(1+0,25)^3 + \\ &+ 516141,52/(1+0,25)^4 = 223536,26 + 227309,51 + 225480,06 + 211411,57 = 887737,40 \text{ грн.} \end{aligned}$$

Величина початкових інвестицій PV , які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки:

$$PV = k_{инв} \cdot 3B, \quad (4.20)$$

де $k_{инв}$ – коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію, приймаємо $k_{инв} = 2$;

$3B$ – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, приймаємо 140003,57 грн.

$$PV = k_{инв} \cdot 3B = 2 \cdot 140003,57 = 280007,13 \text{ грн.}$$

Абсолютний економічний ефект $E_{абс}$ для потенційного інвестора від можливого впровадження та комерціалізації науково-технічної розробки становитиме:

$$E_{абс} = \Pi\Pi - PV \quad (4.21)$$

де $ПП$ – приведена вартість зростання всіх чистих прибутків від можливого впровадження та комерціалізації науково-технічної розробки, 887737,40 грн;

PV – теперішня вартість початкових інвестицій, 280007,13 грн.

$$E_{abc} = ПП - PV = 887737,40 - 280007,13 = 607730,27 \text{ грн.}$$

Внутрішня економічна дохідність інвестицій E_g , які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$E_g = T_{ж} \sqrt[4]{1 + \frac{E_{abc}}{PV}} - 1, \quad (4.22)$$

де E_{abc} – абсолютний економічний ефект вкладених інвестицій, 607730,27 грн;

PV – теперішня вартість початкових інвестицій, 280007,13 грн;

$T_{ж}$ – життєвий цикл науково-технічної розробки, тобто час від початку її розробки до закінчення отримування позитивних результатів від її впровадження, 4 роки.

$$E_g = T_{ж} \sqrt[4]{1 + \frac{E_{abc}}{PV}} - 1 = (1 + 607730,27/280007,13)^{1/4} = 0,33.$$

Мінімальна внутрішня економічна дохідність вкладених інвестицій τ_{min} :

$$\tau_{min} = d + f, \quad (4.23)$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2024 році в Україні $d = 0,1$;

f – показник, що характеризує ризикованість вкладення інвестицій, приймемо 0,15.

$\tau_{min} = 0,1 + 0,15 = 0,25 < 0,33$ свідчить про те, що внутрішня економічна дохідність інвестицій E_g , які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки вища мінімальної внутрішньої дохідності. Тобто інвестувати в науково-дослідну роботу за темою «Комп'ютерно-інтегрований інтелектуальний інтерфейс для управління мікрофермою» доцільно.

Період окупності інвестицій $T_{ок}$ які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$T_{ок} = \frac{1}{E_{\epsilon}}, \quad (4.24)$$

де E_{ϵ} – внутрішня економічна дохідність вкладених інвестицій.

$$T_{ок} = 1 / 0,33 = 2,99 \text{ р.}$$

$T_{ок} < 3$ -х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

4.5 Висновки до розділу

Згідно проведених досліджень рівень комерційного потенціалу розробки за темою «Комп'ютерно-інтегрований інтелектуальний інтерфейс для управління мікрофермою» становить 39,0 бала, що, свідчить про комерційну важливість проведення даних досліджень (рівень комерційного потенціалу розробки вище середнього).

При оцінюванні за технічними параметрами, згідно узагальненого коефіцієнту якості розробки, науково-технічна розробка переважає існуючі аналоги приблизно в 2,12 рази.

Також термін окупності становить 2,99 р., що менше 3-х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

Отже можна зробити висновок про доцільність проведення науково-дослідної роботи за темою «Комп'ютерно-інтегрований інтелектуальний інтерфейс для управління мікрофермою».

ВИСНОВКИ

В процесі виконання магістерської кваліфікаційної роботи були досягнуті наступні результати.

У першому розділі проведено аналіз сучасних методів контролю параметрів мікроклімату, зокрема вологості ґрунту, температури та освітленості. Визначено переваги та недоліки різних підходів, а також вимоги до системи автоматизації для забезпечення оптимальних умов вирощування рослин у мікрофермах. Проведено техніко-економічне обґрунтування доцільності автоматизації та сформульовано критерії вибору обладнання.

У другому розділі розроблено архітектуру автоматизованої системи контролю мікроклімату. На основі порівняльного аналізу обрано апаратні засоби, включаючи датчики вологості ґрунту, температури та освітленості, а також контролери. Визначено оптимальну конфігурацію системи для управління насосами та освітленням.

У третьому розділі створено алгоритми автоматизованого управління та розроблено програмне забезпечення на базі NodeMCU з використанням Arduino IDE. Забезпечено інтеграцію з хмарними платформами для віддаленого моніторингу та управління. Розроблено інструкції користувача для роботи з системою. Проведені експерименти підтвердили працездатність і надійність системи.

В економічній частині роботи виконано розрахунки ефективності впровадження автоматизованої системи, що показали її економічну доцільність для малих і середніх фермерських господарств.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Automated Smart Wick System-Based Microfarm Using Internet of Things. [Електронний ресурс] // Romeo Jr Lumabiang Jorda, C. Alcabasa, A. Buhay, Charlhaine Eunice Dela Cruz, J. P. Mendoza, A. Tolentino, Lean Karlo Tolentino, Edmon Fernandez, August Thio-ac, Jessica Velasco, Nilo Arago. – 2019. – URL: https://www.researchgate.net/publication/337019517_Automated_Smart_Wick_System-Based_Microfarm_Using_Internet_of_Things
2. First you need the farmers: The microfarm system as a critical intervention in the alternative food movement: [Електронний ресурс] // Kent Curtis, Grace Hand. – 2024. – URL: https://www.researchgate.net/publication/379724831_First_you_need_the_farmers_The_microfarm_system_as_a_critical_intervention_in_the_alternative_food_movement
3. Co-Design and Experimentation of a Prototype of Agroecological Micro-Farm Meeting the Objectives Set by Climate-Smart Agriculture: [Електронний ресурс] // Stan Selbonne , Loïc GuindeFrançois Causeret, Thierry Bajazet, Lucienne Desfontaines, Mathieu Duval, Jorge Sierra, Franck Solvar, Régis Tournebize ,Jean-Marc Blazy. -2024 – URL: https://www.researchgate.net/publication/366971229_Coesign_and_Experimentation_of_a_Prototype_of_Agroecological_Micro-arm_Meeting_the_Objectives_Set_by_Climate-Smart_Agriculture
4. Модульні мікроферми. [Електронний ресурс] // ecotech.news, Марія Стецюк - 2019 – URL: <https://ecotech.news/equipment/675-modulni-mikrofermi-perevernut-kharchovu-promislovist.html>
5. Інтернет речей у сільському господарстві: виклики та можливості. [Електронний ресурс] // iot-agriculture.ua. -2023 – URL: <https://iot-agriculture.ua/challenges-opportunities>
6. Інтелектуальні інформаційні системи для керування технологіями у сільському господарстві. [Електронний ресурс] // Прохорова Л. О., Кузьмічев О. В. – 2022 – URL: <http://isystems.agrotech.com.ua/>
7. Інтернет речей (IoT): що це та його використання в сільському господарстві.

[Електронний ресурс] // weagro.com.ua. - 2024 – URL: <https://weagro.com.ua/blog/internet-rechej-iot-shho-cze-ta-jogo-vykorystannya-v-silskomu-gospodarstvi/>

8. Автоматизація теплиці та її реалізація. [Електронний ресурс] // firebox.com.ua. - 2024 – URL: <https://firebox.com.ua/uk/avtomatyzatsiya-teplicity-dostupnymy-sredstvamy/>

9. NodeMCU ESP8266 Detailed Review [Електронний ресурс] // make-it.ca - 2024 – URL: <https://www.make-it.ca/nodemcu-details-specifications/>.

10. Цифровий датчик інтенсивності світла BH1750 [Електронний ресурс] // mikrotik.kpi.ua – 2015 – URL: <https://mikrotik.kpi.ua/index.php/courses-list/category-raspberry/35-digital-sensor-light-intensity-bh1750fvi-i2c>

11. Automated Microclimate Regulation in Agricultural Facilities Using the Air Curtain System [Електронний ресурс] // Nikolay Kiktev, Taras Lendiel, Viktor Vasilenkov, Oksana Kaprallyuk, Taras Hutsol, Szymon Glowacki, Maciej Kuboń, Zbigniew Kowalczyk. – 2021 – URL: <https://www.mdpi.com/1424-8220/21/24/8182>

12. Що таке база даних? [Електронний ресурс] // oracle.com - 2024 – URL: <https://www.oracle.com/ua/database/what-is-database/>

13. Схема підключення ESP8266 та DHT22 [Електронний ресурс] // geekmatic.in.ua - 2019 – URL: https://geekmatic.in.ua/ua/grafik_temperaturi_esp8266

14. Датчик освітленості цифровий [Електронний ресурс] // <https://arduino.ua> - 2024 – URL: <https://arduino.ua/prod1116-datchik-osveshhenosti-cifrovoi-bh1750fvi>

15. AeroFarms. [Електронний ресурс] // aerofarms.com - 2024 – URL: <https://aerofarms.com/>

16. Freight Farms.[Електронний ресурс] // freightfarms.com - 2024 – URL: <https://freightfarms.com/>

17. Tower Garden. [Електронний ресурс] // towergarden.com- 2024 – URL: <https://towergarden.com/>

18. Click and Grow. [Електронний ресурс] // clickandgrow.com - 2024 – URL: <https://clickandgrow.com/>

19. Back to the Roots. [Електронний ресурс] // backtotheroots.com - 2024 – URL:

<https://backtotheroots.com/>

20. Growtainer. [Електронний ресурс] // growtainer.com - 2024 – URL: <https://growtainer.com/>

21. Square Roots. [Електронний ресурс] // squareroostgrow.com - 2024 – URL: <https://squareroostgrow.com/>

22. Arduino. [Електронний ресурс] // geekmatic.in.ua - 2024 – URL: https://geekmatic.in.ua/ua/arduino_summary

23. Датчик температури та вологості DHT22. [Електронний ресурс] // mini-tech.com.ua - 2024 – URL: <https://www.mini-tech.com.ua/ua/datchik-temperature-i-vlazhnosti>

24. Датчик температури DS18B20. [Електронний ресурс] // mini-tech.com.ua - 2024 – URL: <https://www.mini-tech.com.ua/datchik-temperature-ds18b20>

25. Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт / Уклад. : В. О. Козловський, О. Й. Лесько, В. В. Кавецький. – Вінниця : ВНТУ, 2021. – 42 с.

26. Кавецький В. В. Економічне обґрунтування інноваційних рішень: практикум / В. В. Кавецький, В. О. Козловський, І. В. Причепа – Вінниця : ВНТУ, 2016. – 113 с.

27. Бределєв І.В. Розробка персональної системи підтримки прийняття рішень на основі мобільних технологій. LIII Всеукраїнська науково-технічна конференція факультету інтелектуальних інформаційних технологій та автоматизації [Електронний ресурс]. URL: <https://conferences.vntu.edu.ua/index.php/all-fksa/all-fksa-2024/paper/view/20930>.

28. Processing Topologies in IoT. [Електронний ресурс] // Anisha K. – 2024-URL: <https://www.youtube.com/watch?v=CMsHqqYxUiU>

29. Методичні вказівки до виконання магістерських кваліфікаційних робіт для студентів спеціальності 174 «Автоматизація, комп'ютерно-інтегровані технології та робототехніка» (кафедра комп'ютерних систем управління) / Уклад.: В.М. Дубовой – Вінниця : ВНТУ, 2023, 45 с.

ДОДАТКИ

Додаток Б (обов'язковий)

ВНТУ

ЗАТВЕРДЖЕНО

Зав. кафедри КСУ ВНТУ,

д.т.н., професор

В'ячеслав КОВТУН

"4" грудня 2024 р.

ТЕХНІЧНЕ ЗАВДАННЯ

на виконання магістерської кваліфікаційної роботи

Комп'ютерно-інтегрований інтелектуальний інтерфейс для управління
мікрофермою

08-33.МКР.002.00.000 ТЗ

Студент групи 2АКІТР-23мІгор БРЕДЕЛЄВ

Ім'я ПРІЗВИЩЕ

Керівник д.т.н., проф., зав. кафедри КСУ

В'ячеслав КОВТУН

Ім'я ПРІЗВИЩЕ

Вінниця 2024

1. Назва та галузь застосування

1.1 Назва – Комп'ютерно-інтегрований інтелектуальний інтерфейс для управління мікрофермою.

1.2 Галузь застосування – управління мікрофермою.

2 Підстава для проведення розробки.

Тема магістерської кваліфікаційної роботи затверджена наказом по ВНТУ №310 від 17.09.2024 р.

3 Мета та призначення розробки.

Метою магістерської кваліфікаційної роботи є розробка та обґрунтування ефективної моделі комп'ютерно-інтегрованого інтелектуального інтерфейсу для управління мікрофермою

4 Джерела розробки.

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

1. Automated Smart Wick System-Based Microfarm Using Internet of Things https://www.researchgate.net/publication/337019517_Automated_Smart_Wick_System-Based_Microfarm_Using_Internet_of_Things
2. First you need the farmers: The microfarm system as a critical intervention in the alternative food movement: https://www.researchgate.net/publication/379724831_First_you_need_the_farmers_The_microfarm_system_as_a_critical_intervention_in_the_alternative_food_movement
3. Co-Design and Experimentation of a Prototype of Agroecological Micro-Farm Meeting the Objectives Set by Climate-Smart Agriculture: https://www.researchgate.net/publication/366971229_Co-Design_and_Experimentation_of_a_Prototype_of_Agroecological_Micro-Farm_Meeting_the_Objectives_Set_by_Climate-Smart_Agriculture

5 Вимоги до розробки.

5.1 Перелік головних функцій:

- збір інформації;
- керування процесами;
- формування відповідей.

5.2 Основні технічні вимоги до розробки.

5.2.1 Вимоги до програмної платформи:

5.2.1.1 WINDOWS 10;

5.2.1.2 Arduino IDE;

5.2.2 Умови експлуатації системи:

5.2.2.1 робота через Telegram;

5.2.2.2 можливість цілодобового функціонування системи;

5.2.2.3 дані оновлюються і є актуальними.

6 Стадії та етапи розробки.

6.1 Пояснювальна записка:

1. Аналіз методів, принципів, підходів і засобів реалізації задачі автоматизації процесами в об'єкті управління відповідно до теми дипломної роботи. Постановка задач дослідження
«1» листопада 2024 р.
2. Удосконалення технології прийняття рішень при автоматизації об'єкту управління «1» 10 2024 р.
3. Визначення технічних характеристик системи «3» листопада 2024 р.
4. Розробка апаратної частини системи «5» листопада 2024 р.
5. Розробка програмної частини системи «6» листопада 2024 р.

6.2 Графічні матеріали:

1. Розробка моделі системи «12» листопада 2024 р.
2. Розробка діаграм системи «18» листопада 2024 р.
3. Тестування програмного забезпечення «20» листопада 2024 р.

7 Порядок контролю і приймання.

7.1 Хід виконання роботи контролюється керівником роботи. Рубіжний контроль провести до «28» листопада 2024 р.

7.2 Атестація проєкту здійснюється на попередньому захисті. Попередній захист магістерської кваліфікаційної роботи провести до «1» грудня 2024 р.

7.3 Підсумкове рішення щодо оцінки якості виконання роботи приймається на засіданні ЕК. Захист магістерської кваліфікаційної роботи провести до «11» грудня 2024 р.

Додаток В (вибірковий)
Лістинг програми

```
#include <ESP8266WiFi.h>
#include <WiFiClientSecure.h>
#include <UniversalTelegramBot.h>
#include <OneWire.h>
#include <DallasTemperature.h>
#include <Wire.h>
#include <BH1750.h>
#include <DHT.h>

const char* ssid = "...";
const char* password = "...";

#define TELEGRAM_TOKEN "..."
#define CHAT_ID "..."

const int lightRelayPin = D5;
const int moistureRelayPin = D1;
const int moistureSensorPin = D0;

#define ONE_WIRE_BUS D6

OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);
```

```
BH1750 lightMeter;

#define DHTPIN D7

#define DHTTYPE DHT22

DHT dht(DHTPIN, DHTTYPE);

#define SOIL_MOISTURE_ANALOG_PIN A0

#define SOIL_MOISTURE_DIGITAL_PIN D0

WiFiClientSecure wifiClient;

UniversalTelegramBot bot(TELEGRAM_TOKEN, wifiClient);

long lastUpdateId = 0;

unsigned long botRequestDelay = 1000;

unsigned long lastTimeBotRan;

float lightThreshold = 3.0;

float moistureThreshold = 1.0;

void sendMessageToTelegram(String message);

void setupSensors();

void handleTelegramUpdates();

String getDataFromSensors();

float getSoilMoisture();

String getSoilMoistureStatus();
```

```
void checkLight();  
void checkMoisture();  
  
void setup() {  
    Serial.begin(115200);  
  
    pinMode(lightRelayPin, OUTPUT);  
    digitalWrite(lightRelayPin, HIGH);  
  
    pinMode(moistureRelayPin, OUTPUT);  
    digitalWrite(moistureRelayPin, HIGH);  
  
    pinMode(moistureSensorPin, INPUT);  
  
    WiFi.begin(ssid, password);  
    while (WiFi.status() != WL_CONNECTED) {  
        delay(1000);  
        Serial.println("Підключення до WiFi...");  
    }  
    Serial.println("WiFi підключено");  
  
    wifiClient.setInsecure();  
  
    setupSensors();
```

```
moistureThreshold = 3.0;
```

```
int updates = bot.getUpdates(0);
```

```
if (updates > 0) {
```

```
    lastUpdateId = bot.messages[updates - 1].update_id;
```

```
}
```

sendMessageToTelegram("Вітаємо, мікроферму було запущено, для отримання інформації з датчиків скористайтесь командою /get_data , для керування світлом введіть команду /set_light_threshold , для керування вологою - /set_moisture_threshold");

```
delay(5000);
```

```
}
```

```
void loop() {
```

```
    if (millis() - lastTimeBotRan > botRequestDelay) {
```

```
        handleTelegramUpdates();
```

```
        lastTimeBotRan = millis();
```

```
    }
```

```
    checkMoisture();
```

```
    checkLight();
```

```
    delay(3000);
```

```
}
```

```
void handleTelegramUpdates() {
```

```

int updates = bot.getUpdates(lastUpdateId + 1);

if (updates > 0) {
    for (int i = 0; i < updates; i++) {
        String command = bot.messages[i].text;
        lastUpdateId = bot.messages[i].update_id;

        Serial.println("Received command: " + command);
        if (command == "/get_data") {
            String message = getDataFromSensors();
            sendMessageToTelegram(message);
        } else if (command.startsWith("/set_light_threshold ")) {
            String thresholdStr = command.substring(20);
            lightThreshold = thresholdStr.toFloat();
            sendMessageToTelegram("Новий поріг освітленості: " + String(lightThreshold) + " люкс");
        } else if (command.startsWith("/set_moisture_threshold ")) {
            String thresholdStr = command.substring(23);
            moistureThreshold = thresholdStr.toFloat();
            sendMessageToTelegram("Новий поріг вологості ґрунту: " + String(moistureThreshold) + " V");
        }
    }
} else {
    Serial.println("No new updates");
}

```



```
}
```

```
void sendMessageToTelegram(String message) {  
    bool success = bot.sendMessage(CHAT_ID, message, "");  
    if (success) {  
        Serial.println("Message sent successfully");  
    } else {  
        Serial.println("Failed to send message");  
    }  
}
```

```
void setupSensors() {  
    Wire.begin(D3, D2);  
    sensors.begin();  
    dht.begin();  
    lightMeter.begin();  
    pinMode(SOIL_MOISTURE_DIGITAL_PIN, INPUT);  
}
```

```
float getSoilMoisture() {  
    int analogValue = analogRead(SOIL_MOISTURE_ANALOG_PIN);  
    float voltage = analogValue * (3.3 / 1023.0);  
    return voltage;  
}
```

```
String getSoilMoistureStatus() {
    int digitalValue = digitalRead(SOIL_MOISTURE_DIGITAL_PIN);
    if (digitalValue == LOW) {
        return "Вологий ґрунт";
    } else {
        return "Сухий ґрунт";
    }
}
```

```
String getDataFromSensors() {
    float airTemperature = dht.readTemperature();
    float humidity = dht.readHumidity();

    sensors.requestTemperatures();
    float soilTemperature = sensors.getTempCByIndex(0);

    float light = lightMeter.readLightLevel();

    float soilMoisture = getSoilMoisture();
    String soilMoistureStatus = getSoilMoistureStatus();

    String message = "Температура повітря: " + String(airTemperature) + " °C\n" +
        "Вологість: " + String(humidity) + " %\n" +
        "Температура землі: " + String(soilTemperature) + " °C\n" +
        "Освітленість: " + String(light) + " люкс\n" +
```

"Вологість ґрунту (аналог): " + String(soilMoisture) + " V\n" +

"Стан ґрунту (цифровий): " + soilMoistureStatus;

return message;

}

void checkMoisture() {

float soilMoisture = getSoilMoisture();

Serial.print("Аналогове значення вологості: ");

Serial.println(soilMoisture);

if (soilMoisture < moistureThreshold) {

digitalWrite(moistureRelayPin, HIGH);

Serial.println("Реле поливу увімкнено (вологість нижче порогу)");

} else {

digitalWrite(moistureRelayPin, LOW);

Serial.println("Реле поливу вимкнено (вологість вище порогу)");

}

}

void checkLight() {

float light = lightMeter.readLightLevel();

Serial.print("Освітленість: ");

```
Serial.println(light);
```

```
if (light < lightThreshold) {
```

```
    digitalWrite(lightRelayPin, LOW);
```

```
    Serial.println("Реле освітлення увімкнено (освітленість нижче порогу)");
```

```
} else {
```

```
    digitalWrite(lightRelayPin, HIGH);
```

```
    Serial.println("Реле освітлення вимкнено (освітленість вище порогу)");
```

```
}
```

```
}
```

Додаток Г (обов'язковий)

ІЛЮСТРАТИВНА ЧАСТИНА

Комп'ютерно-інтегрований інтелектуальний інтерфейс для управління
мікрофермою

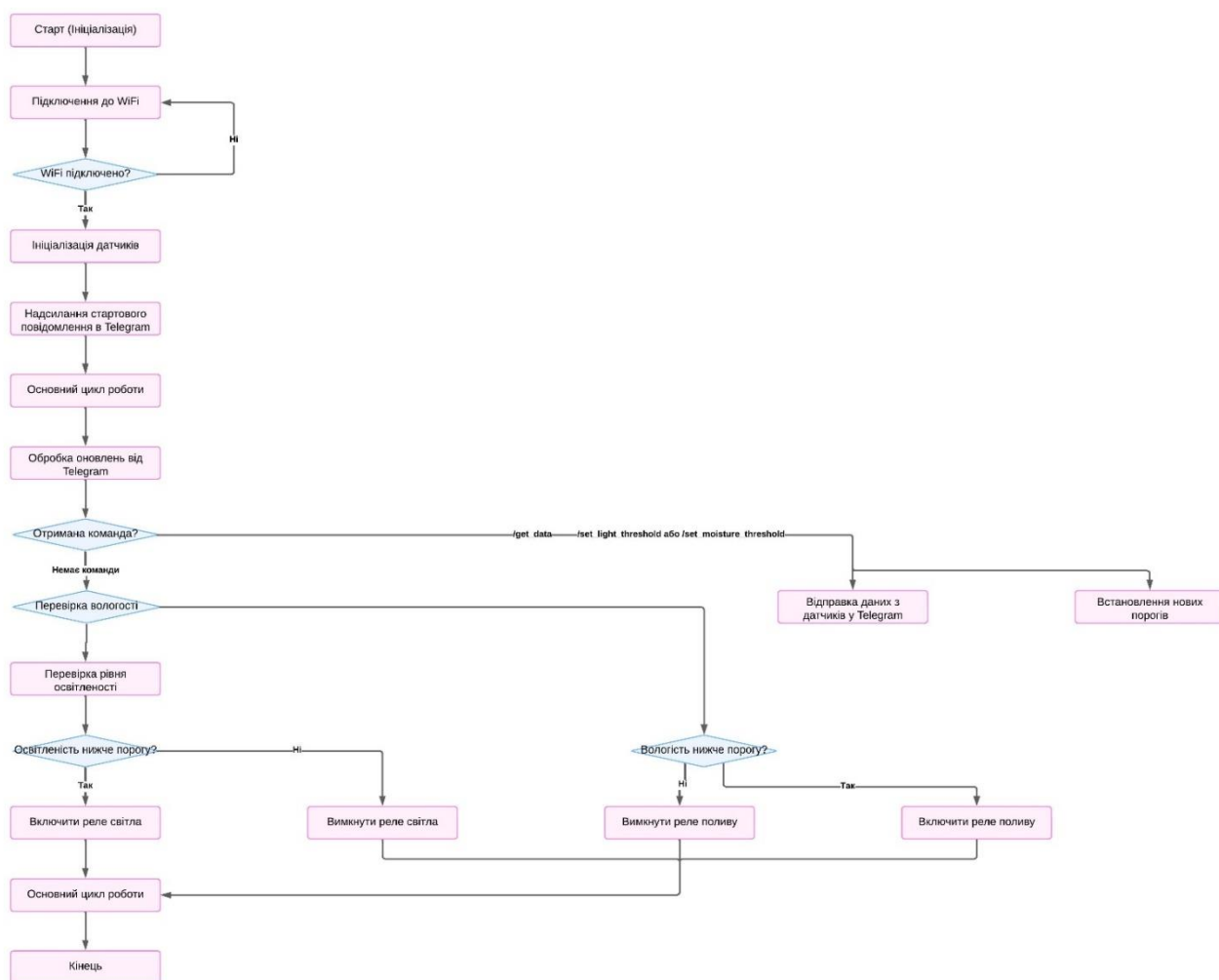
Студент групи 2АКІТР-23м

Підпис Ігор БРЕДЕЛЄВ
Ім'я ПРІЗВИЩЕ

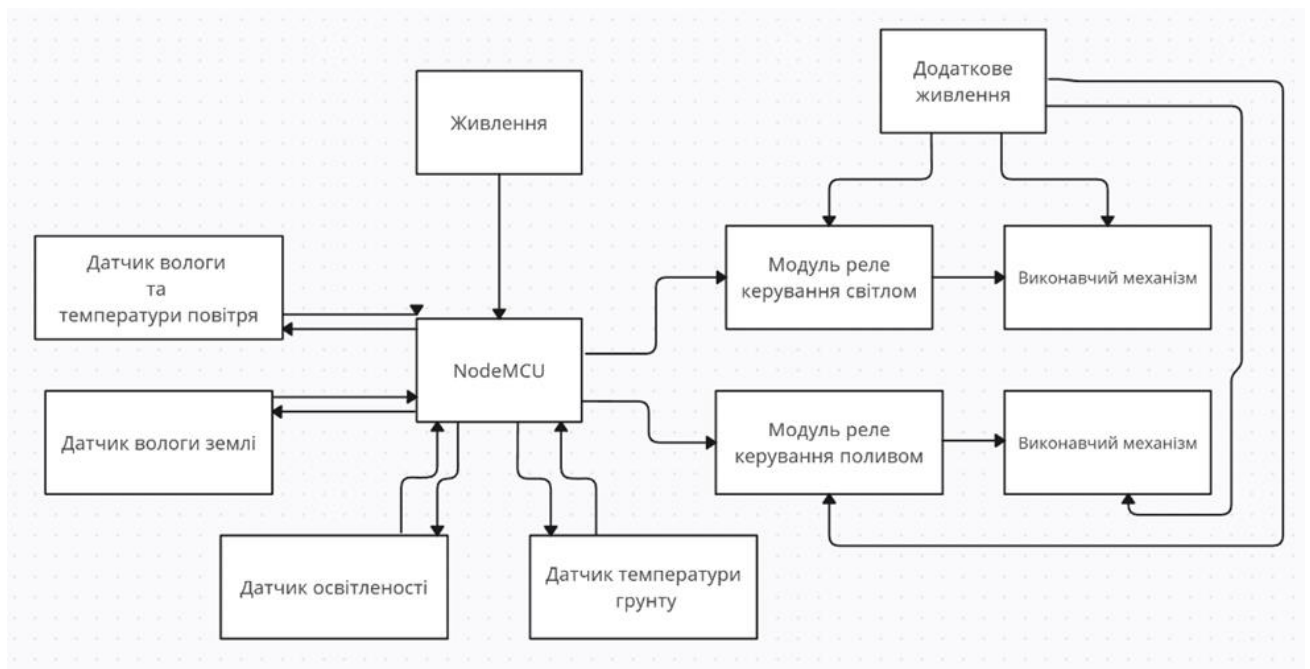
Керівник: професор, завідувач кафедри КСУ

Підпис В'ячеслав КОВТУН
Ім'я ПРІЗВИЩЕ

UML-діаграма діяльності інтелектуальної системи контролю мікроклімату



Структурна схема мікроферми



Апаратна схема мікроферми

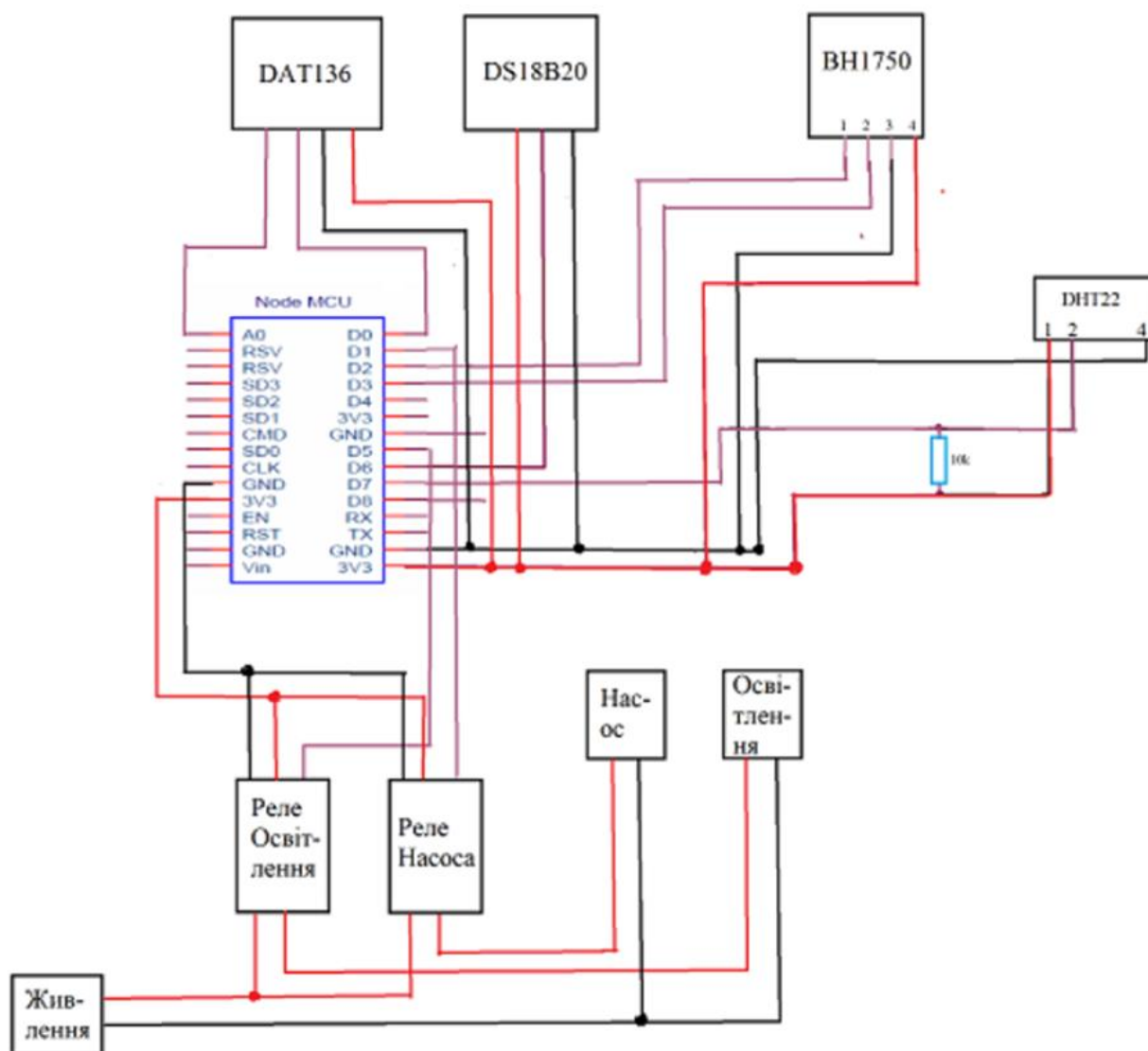


Схема програми контролю мікроферми

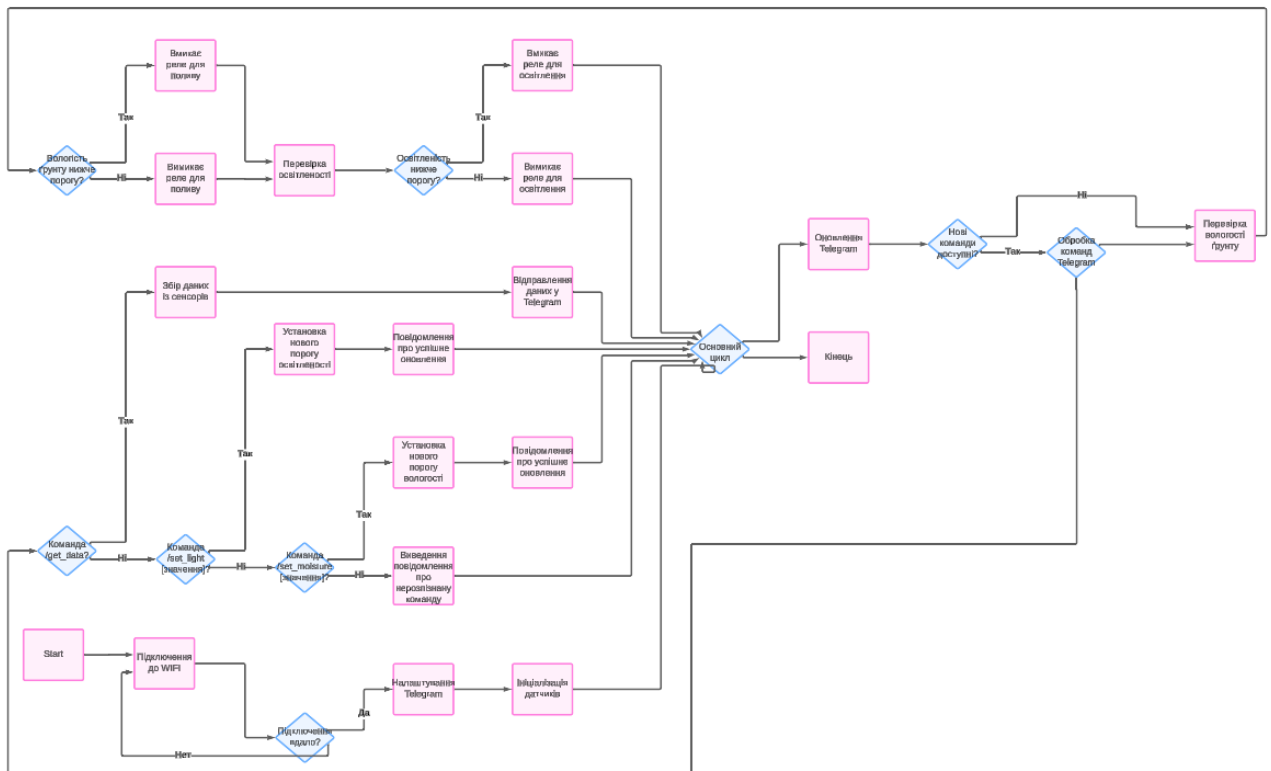


Схема структурна ієрархічна АСУТП

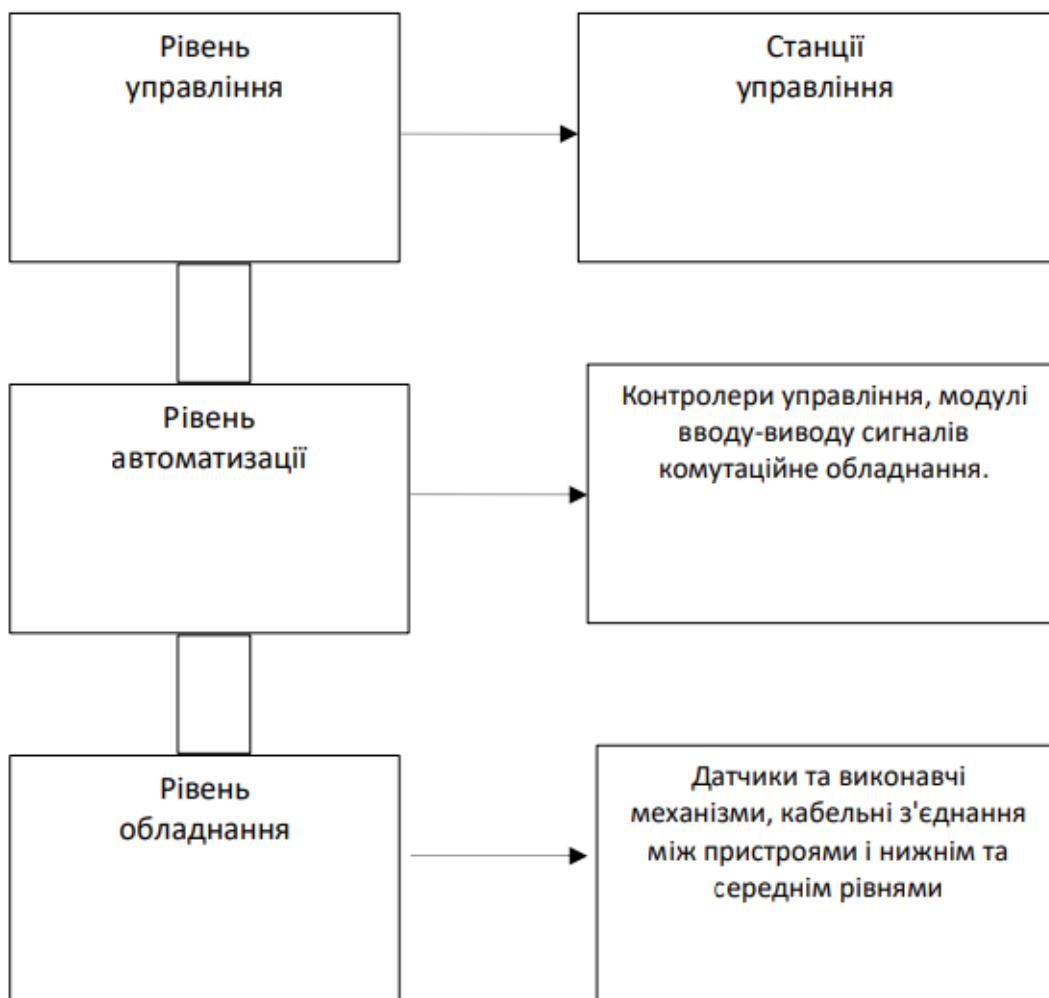


Схема контролю мікрокліматом

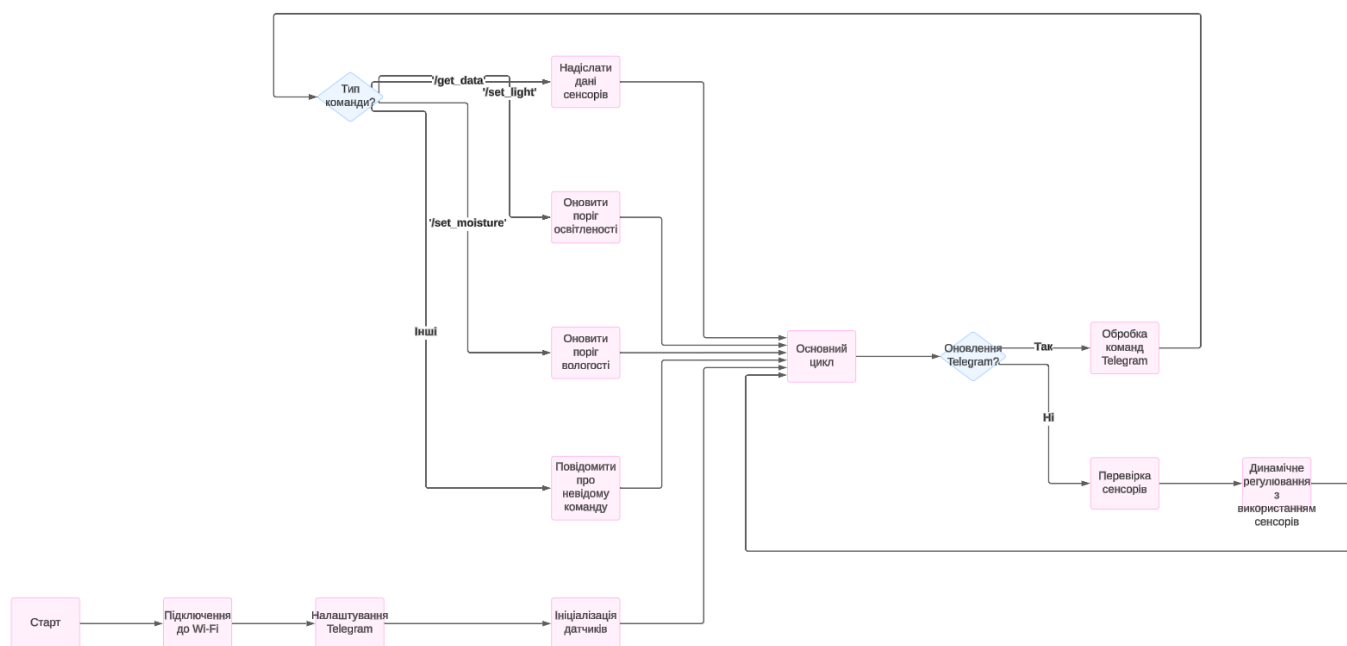
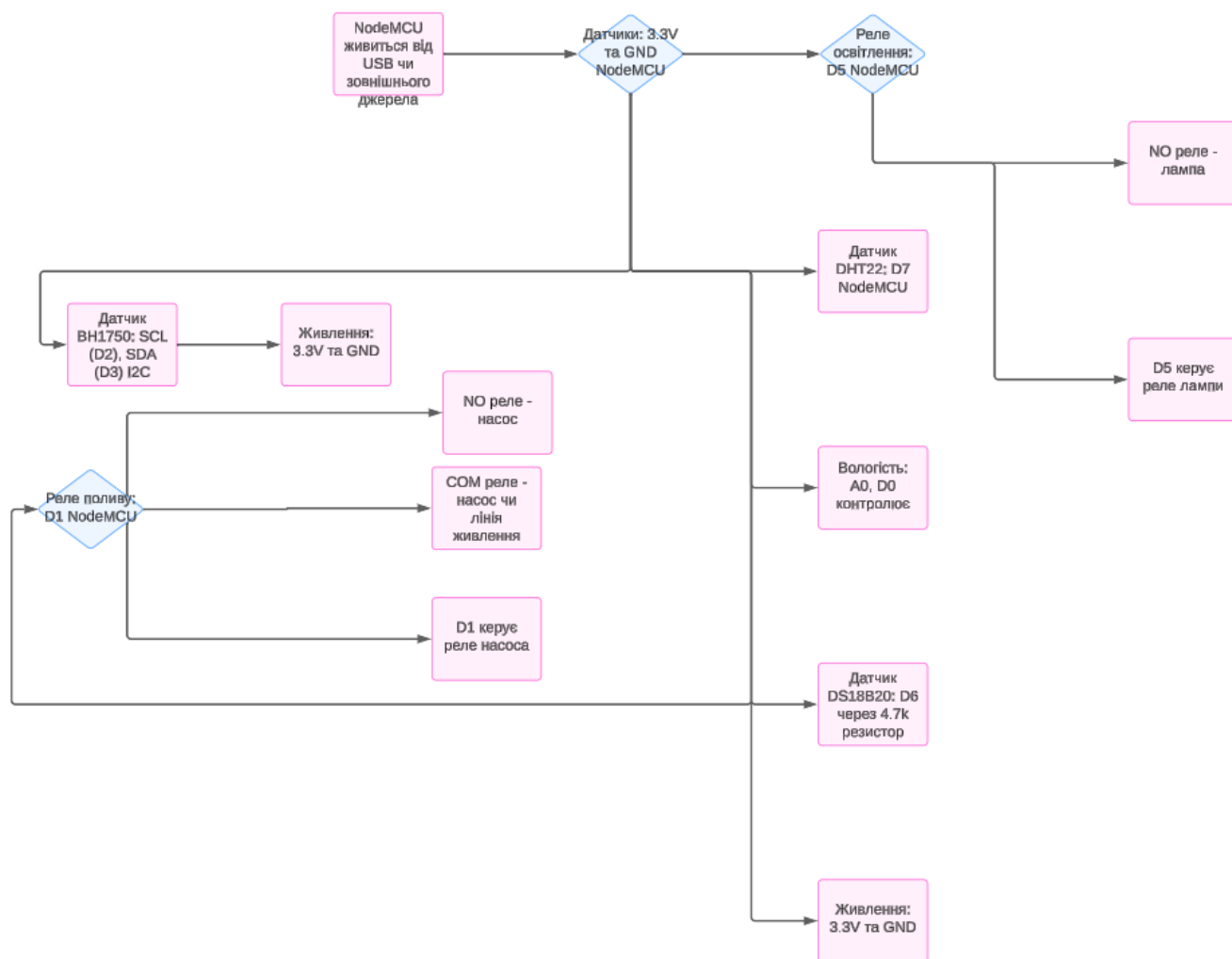


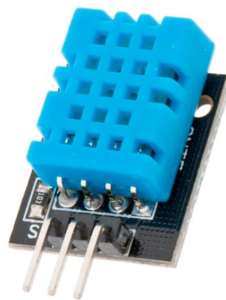
Схема підключення та живлення компонентів



ЕЛЕКТРОННІ КОМПОНЕНТИ СИСТЕМИ АВТОМАТИЗАЦІЇ



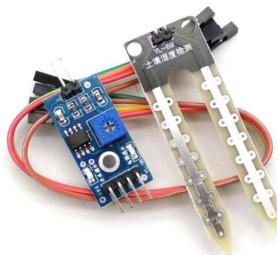
Модуль керування NodeMCU



Датчик вологості та температури повітря DHT22



Датчик температури землі



Датчик вологості землі