

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ
КАФЕДРА ІНФОРМАЦІЙНИХ СИСТЕМ ТА ТЕХНОЛОГІЙ

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «Інтелектуальні системи управління Smart Garden для
автоматизації догляду за рослинами»

на здобуття освітнього ступеня бакалавра
зі спеціальності 126, Інформаційні системи та технології
(код, найменування спеціальності)
освітньо-професійної програми Інформаційні системи та технології
(назва)

*Кваліфікаційна робота містить результати власних досліджень.
Використання ідей, результатів і текстів інших авторів мають посилання
на відповідне джерело*

(підпис) Герман ЧЕТВЕРИК
Ім'я, ПРІЗВИЩЕ здобувача

Виконав: здобувач вищої освіти гр. ІСД-43

Герман ЧЕТВЕРИК
Ім'я, ПРІЗВИЩЕ

Керівник: К.т.н., доцент Андрій АРОНОВ
науковий ступінь, Ім'я, ПРІЗВИЩЕ
вчене звання

Рецензент: _____
науковий ступінь, Ім'я, ПРІЗВИЩЕ
вчене звання

Київ 2025

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ**
Навчально-науковий інститут Інформаційних технологій

Кафедра Інформаційних систем та технологій

Ступінь вищої освіти Бакалавр

Спеціальність Інформаційні системи та технології

Освітньо-професійна програма Інформаційні системи та технології

ЗАТВЕРДЖУЮ

Завідувач кафедру ІСТ

_____ Каміла СТОРЧАК

«_____» _____ 2025р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

Четверику Герману Олексійовичу

(прізвище, ім'я, по батькові здобувача)

1. Тема кваліфікаційної роботи: Інтелектуальні системи управління Smart Garden для автоматизації догляду за рослинами

керівник кваліфікаційної роботи Андрій АРОНОВ, к.т.н., доцент,

(Ім'я, ПРИЗВИЩЕ, науковий ступінь, вчене звання)

затверджені наказом Державного університету інформаційно-комунікаційних технологій від «24» Лютого 2025р. № 56

2. Строк подання кваліфікаційної роботи «30» Травня 2025р.

3. Вихідні дані до кваліфікаційної роботи: _____

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

1. Теоретичні аспекти Smart Garden
2. Методи та моделі автоматизації садівництва
3. Розробка ІОТ-системи для автоматизованого догляду за садом

5. Перелік ілюстративного матеріалу: *презентація*

6. Дата видачі завдання «24» Лютого 2025р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	ТЕОРЕТИЧНІ АСПЕКТИ SMART GARDEN	28 лютого 2025	Виконано
2	МЕТОДИ ТА МОДЕЛІ АВТОМАТИЗАЦІЇ САДІВНИЦТВА	4 березня 2025	Виконано
3	Методи автоматизації садівництва	10 березня 2025	Виконано
4	Моделі автоматизації садівництва	15 березня 2025	Виконано
5	Алгоритм реалізації процесу автоматизації садівництва	25 березня 2025	Виконано
6	РОЗРОБКА ІОТ-СИСТЕМИ ДЛЯ АВТОМАТИЗОВАНОГО ДОГЛЯДУ ЗА САДОМ	1 квітня 2025	Виконано
7	Структурна схема ІоТ-системи для автоматизованого догляду за садом	10 квітня 2025	Виконано
8	Вибір обладнання та програмного забезпечення	15 квітня 2025	Виконано

Здобувач(ка) вищої освіти

(підпис)

Герман ЧЕТВЕРИК

(Ім'я, ПРІЗВИЩЕ)

Керівник
кваліфікаційної роботи

(підпис)

Андрій АРОНОВ

(Ім'я, ПРІЗВИЩЕ)

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня бакалавра (магістра): 61 стор., 9 рис., 3 табл., 20 джерел.

Мета роботи – Метою даного дослідження є розкриття принципів використання Інтернету речей для автоматизації садівництва.

Об'єкт дослідження – Пристрій для автоматизації садівництва

Предмет дослідження – розробка схем підключення систем і програмної моделі керування системою автоматизації садівництва.

Методи дослідження - Під час виконання роботи використовувались теоретичні методи досліджень із застосуванням апарату термодинаміки, електроніки, теорії ймовірності, математичної статистики, похибок.

Короткий зміст роботи:

Дипломна робота присвячена розробці інтелектуальної системи управління Smart Garden, яка базується на технологіях Інтернету речей (IoT) та призначена для автоматизації догляду за рослинами. Основна мета дослідження — проаналізувати та впровадити IoT-рішення, здатне здійснювати моніторинг вологості ґрунту, керування поливом і збором даних про стан навколишнього середовища. Для досягнення цієї мети було створено структурну схему системи, підібрано відповідне апаратне та програмне забезпечення, зокрема мікроконтролери, датчики вологості, бездротові модулі зв'язку та елементи живлення. Запропоноване рішення спрямоване на підвищення ефективності використання води, зменшення ручної праці та забезпечення сталого вирощування рослин як у домашніх умовах, так і в малих господарствах. Практична цінність полягає у можливості широкого впровадження цієї системи в умовах обмеженого простору та часу, з орієнтацією на екологічність та енергоефективність.

КЛЮЧОВІ СЛОВА

ІНТЕРНЕТ РЕЧЕЙ; SMART GARDEN; АВТОМАТИЗАЦІЯ САДІВНИЦТВА; ДАТЧИКИ ВОЛОГОСТІ ҐРУНТУ; МІКРОКОНТРОЛЕР; БЕЗДРОТОВИЙ ЗВ'ЯЗОК; КЕРУВАННЯ ПОЛИВОМ; ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ УПРАВЛІННЯ.

ЗМІСТ

ВСТУП	8
РОЗДІЛ 1 ТЕОРЕТИЧНІ АСПЕКТИ SMART GARDEN	10
1.1 Smart Garden: поняття, сутність, генезис.....	10
1.2 Інтернет речей для автоматизації садівництва.....	15
1.3 Проблематика та постановка завдань дослідження	20
Висновки до розділу	21
РОЗДІЛ 2 МЕТОДИ ТА МОДЕЛІ АВТОМАТИЗАЦІЇ САДІВНИЦТВА.....	22
2.1 Методи автоматизації садівництва	22
2.2 Моделі автоматизації садівництва	32
2.3 Алгоритм реалізації процесу автоматизації садівництва.....	37
Висновки до розділу	40
РОЗДІЛ 3 РОЗРОБКА ІОТ-СИСТЕМИ ДЛЯ АВТОМАТИЗОВАНОГО ДОГЛЯДУ ЗА САДОМ	41
3.1 Структурна схема ІоТ-системи для автоматизованого догляду за садом.....	41
3.2 Вибір обладнання та програмного забезпечення.....	42
3.3 Верифікація результатів дослідження	50
Висновки до розділу	52
ВИСНОВКИ.....	53
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ.....	54
ДОДАТКИ.....	56
Демонстраційні матеріали (Презентація)	61

ВСТУП

Актуальність дослідження. Садівництво було основним продуктом української культури протягом століть, забезпечуючи сім'ї свіжими фруктами та овочами, додаючи дому краси та цінності. Однак традиційні методи домашнього садівництва можуть вимагати багато часу та праці. В останні роки технології розумного садівництва з'явилися як рішення цих проблем. Це дозволило автоматизувати кілька садових завдань, таких як полив, удобрення та навіть косіння газону.

Розумне садівництво – це термін, що означає використання автоматизованих технологій для спрощення процесу вирощування рослин. Що включає використання датчиків, автоматизації та Інтернету речей (IoT), щоб допомогти домогосподарям і садівникам ефективніше доглядати за своїми газонами та рослинами.

Завдяки автоматизації очікується, що глобальний ринок продуктів Smart Gardening швидко зростатиме в найближчі роки. Згідно зі звітом Polaris Market Research, очікується, що глобальний ринок інтелектуальних систем внутрішнього садівництва досягне 3,22 мільярда доларів США до 2030 року, зростаючи на 5,2% у середньому в порівнянні з 2,06 мільярда доларів США у 2021 році. Це зростання зумовлене кількома факторами, зокрема зростаюча популярність домашнього садівництва, зростаюче визнання технологій автоматизації та зростання доступності інноваційних продуктів для садівництва.

Мета та завдання. Метою даного дослідження є розкриття принципів використання Інтернету речей для автоматизації садівництва.

За для досягнення поставленої мети у роботі необхідно виконати низку завдань:

- навести методи та моделі автоматизації садівництва;
- розробити алгоритм реалізації процесу автоматизації садівництва;
- запропонувати структурну схему IoT-системи для автоматизованого догляду за садом;
- здійснити вибір обладнання та програмного забезпечення;
- представити верифікацію результатів дослідження.

Об'єкт дослідження – пристрій для автоматизації садівництва.

Предмет дослідження – розробка схем підключення систем і програмної моделі керування системою автоматизації садівництва.

Методи дослідження. Під час виконання роботи використовувались теоретичні методи досліджень із застосуванням апарату термодинаміки, електроніки, теорії ймовірності, математичної статистики, похибок.

Наукова новизна отриманих результатів.

1) Запропоновано комплексне рішення для автоматизації садівництва на базі Інтернету речей.

2) Розроблені моделі автоматизації садівництва на базі Інтернету речей.

Практичне значення отриманих результатів. Результати дослідження можуть бути впроваджені у рамках реального підприємства для автоматизації садівництва на базі Інтернету речей.

Структура роботи. Дана робота складається зі вступу, трьох розділів, висновків, переліку використаної літератури, що включає в себе 20 літературних джерел. Загальний обсяг роботи становить 64 сторінки.

РОЗДІЛ 1

ТЕОРЕТИЧНІ АСПЕКТИ SMART GARDEN

1.1 Smart Garden: поняття, сутність, генезис

На ринку доступно кілька розумних рішень для садівництва, кожне з яких має свій набір функцій і переваг. Залежно від типу культури, яку потрібно вирощувати, наявності місця, досвіду ведення сільського господарства, наявності часу та зусиль, можна вибрати між простішими та складнішими інтелектуальними системами садівництва, як наведено нижче:

1. Indoor Smart Gardens: внутрішні розумні сади призначені для використання в будинках і квартирах. Ці системи зазвичай використовують світлодіодне освітлення та гідропонні методи вирощування, щоб дозволити вирощувати рослини в приміщенні без прямого сонячного світла.

2. Розумні сади на відкритому повітрі: розумні сади на відкритому повітрі призначені для використання у дворах і внутрішніх двориках. Ці системи зазвичай використовують традиційні методи садівництва, такі як вирощування на ґрунті, але з додаванням автоматизації та розумних функцій.

3. Packaged Smart Gardens: це повні системи садівництва, які включають усе необхідне для початку, наприклад, одиницю для вирощування, горщики для посадок, ґрунт і розсаду. Це чудовий варіант для початківців, які хочуть розпочати садівництво, але не мають часу чи знань, щоб створити традиційний сад.

4. Портативні розумні сади: це невеликі автономні садові системи, які можна переміщати за потреби. Ці системи зазвичай використовуються на балконах, внутрішніх двориках або інших невеликих відкритих просторах. Портативні розумні сади – чудовий варіант для тих, хто хоче гнучкості, щоб переміщати свій сад за потреби, або для тих, хто має обмежений простір.

5. Розумні теплиці. Розумні теплиці – це великі автоматизовані садові системи, розроблені для використання в комерційних умовах, таких як ферми чи розплідники. Ці системи зазвичай використовують передові методи вирощування, такі як гідропоніка, і включають такі функції, як клімат-контроль і автоматичний полив. Розумні теплиці – чудовий варіант для тих, хто хоче вирощувати велику кількість рослин або для тих, кому потрібно контролювати середовище вирощування.

6. Розумні вертикальні сади. Розумні вертикальні сади – це системи садівництва, які використовують вертикально орієнтовані кашпо для вирощування рослин. Ці системи зазвичай призначені для невеликих приміщень, таких як квартири або офіси. Розумні

вертикальні сади – відмінний варіант для тих, хто хоче заощадити простір або для тих, хто хоче вирощувати рослини в більш естетично привабливий спосіб.

7. Розумні віконні сади: розумні віконні сади – це невеликі системи садівництва, призначені для розміщення на підвіконнях. Ці системи зазвичай використовують гідропонні методи вирощування та включають світлодіодне освітлення, щоб рослини могли рости без прямого сонячного світла. Розумні віконні сади підходять для тих, хто хоче вирощувати рослини вдома, але не має місця для традиційного кімнатного саду.

Елементи розумного саду та розуміння системи автоматизації:

Типовий автоматизований сад або сад на основі Інтернету речей використовує шестирівневу структуру; кожен рівень складається з компонентів, які створюють ступінчасту архітектуру для задоволення вимог повного автоматизованого сільськогосподарського рішення. Інфраструктура включає фізичний рівень (обладнання), мережевий рівень (Інтернет і зв'язок), рівень проміжного програмного забезпечення (шлюзи IoT), рівень обслуговування (хмарні продукти), рівень аналітики (великі дані, прогнозовий аналіз тощо) і взаємодію з користувачем (врожайність і досвід фермера).

1. Фізичний рівень (апаратний рівень): складається з датчиків, клапанів, соленоїдів, приводів, мікроконтролерів, шлюзів та іншого обладнання та пристроїв, встановлених на рівні культури. Основна мета цього рівня – перемістити оброблені дані кореневого рівня на вищі рівні.

2. Мережевий рівень: складається з Інтернет-технологій та комунікаційних технологій (таких як LoRa Wi-Fi, стільникових технологій, таких як LTE, CDMA та GSM) для полегшення передачі даних із апаратних пристроїв до хмари або підключених пристроїв.

3. Рівень проміжного програмного забезпечення забезпечує керування пристроями, визначення контексту, взаємодію, портативність платформи та функції безпеки. HYDRA, UBIWARE, UBIROAD і SMEPP чудово забезпечують контекстно-залежні функції; з іншого боку, SOCRADES, GSN, SIRENA та інші ефективно впроваджують безпеку та конфіденційність користувачів у своїй архітектурі.

4. Сервісний рівень. Хмарний сервісний рівень має вирішальне значення для Smart Gardens і надає рішення програмного забезпечення як послуги (SaaS) для сільськогосподарських проблем. Функції зондування, активації та ідентифікації захворювань визначаються за допомогою даних датчиків, розпізнавання обладнання, збереження інформації про хвороби культур і статистичне сортування.

5. Рівень аналітики: рівень використовує аналітичні моделі даних для прогнозування кліматичних змін, аналізу врожайності, прогнозування прибутку та рентабельності інвестицій, а також виявлення хвороб культур, бур'янів і шкідників. Крім розумних

садівників, мультикультурні фермери використовують аналітичний рівень для розуміння різних агрокультур, таких як вермікультура, садівництво, квітникарство тощо.

6. Рівень взаємодії з користувачем: цей рівень надає послуги після врожайності та додаткові послуги, які складаються з об'єктів і ресурсів, таких як поведінка холодного зберігання, зв'язок із фермерами, молочні служби та інструменти видобутку біомаси.

Сучасні сенсорні системи Smart Garden:

1. Bitronics (автоматична гідропоніка): надає сенсорний пристрій, серверну веб-підтримку та бібліотеку налаштованих параметрів росту рослин. Пристрій складається з датчиків води/повітря, температури, вологості, pH і яскравості. Користувач може вибрати

- 1) кількість годин денного світла, необхідну для рослини,
- 2) необхідний оптимальний діапазон pH
- 3) необхідну кількість води та відповідно керувати насосом.

2. Edyn (зовнішній сад і зрошення): Ця система моніторингу на сонячних батареях складається з датчиків температури, вологості ґрунту, вологості ґрунту, освітлення та кислотності та надає користувачам рекомендації щодо профілактичних заходів для кращого росту культур. Система самонавчання визначає оптимальні умови та керує водними клапанами, щоб регулювати pH і рівень вологості та підтримувати здоровий ріст рослин.

Подібно до вищезгаданих пакетних сенсорних систем, існують інші комерційні варіанти, такі як Botanicalls (застосування для внутрішнього чи зовнішнього саду), PlantLink (застосування для внутрішнього чи зовнішнього зрошення) та Open Garden (застосування для внутрішнього чи зовнішнього саду, зрошення та аквапоніки). Це готові до розгортання системи, які забезпечують безпроблемне встановлення.

Крок 3 – Вибір датчика(ів): Замість того, щоб вибирати готовий пакет датчиків, якщо користувач має обмежені або специфічні вимоги, такі як керування водним клапаном з електричним керуванням або надання рекомендацій щодо здоров'я рослин, або моніторинг додаткових параметрів (крім тих, надається разом із пакетом послуг), користувачі можуть вибрати автономні датчики, необхідні лише для цих функцій. Наступні датчики зазвичай використовуються для внутрішнього/зовнішнього застосування Smart Garden:

1. Температура (RTD): Для моніторингу/контролю температури навколишнього середовища.

2. Навколишнє освітлення (датчик яскравості): моніторинг/контроль сонячного світла або світлодіодного освітлення для оптимального фотосинтезу.

3. Вологість ґрунту: для моніторингу рівня вологості ґрунту та регулювання потоку води до необхідних рівнів.

4. pH ґрунту: ці датчики кислотності забезпечують якість ґрунту та рекомендації щодо добрив, необхідних для здорового росту.

5. Вологість повітря: надає дані про вологість погоди та контролює потік води.

6. Датчик CO₂: Цей датчик визначає наявність парникових викидів і регулює необхідну вентиляцію.

Користувачі можуть вибрати одну або кілька комбінацій датчиків (зі згаданих вище), залежно від програми та параметрів, необхідних для моніторингу.

Крок 4. Апаратні платформи: кілька підтримуваних IoT апаратних платформ є комерційно доступними. На основі їхніх ключових специфікацій (процесор, графічний процесор, робоча напруга, тактова частота, ширина шини, система та флеш-пам'ять, підтримка різних протоколів бездротового зв'язку, підключення вводу/виводу, а також розробка кодів і мова програмування) користувачі можуть вибрати апаратне забезпечення. Серед популярних варіантів апаратних платформ для внутрішнього або зовнішнього сільського господарства – Arduino (Uno, Yun і Nano), Intel (Galileo, Edison), Beagle Bone і Raspberry Pi (3, B+ або Zero).

Крок 5. Зв'язок між обладнанням і хмарою. Вибір належного каналу зв'язку (для передачі даних) є критичним для системи автоматизації, тому користувачі повинні бути обережними під час вибору. Комунікація визначає формати передачі даних, шифрування, керування потоком і послідовністю, а також передачу втрачених пакетів даних. Популярні протоколи зв'язку включають WiFi, WiMax, LoRa, LR-WPAN, мобільні мережі (2G, 3G, 4G і 5G) і Bluetooth. Користувачі можуть вибрати один із наведених вище протоколів на основі частоти передачі даних, швидкості передачі, діапазону передачі, енергоспоживання та операційних витрат.

Крок 6. Хмарні платформи: після передачі даних від датчиків рівня поля, аналіз даних, інтерпретація, моніторинг і етапи регулювання виконуються за допомогою хмари. Незважаючи на те, що доступно кілька хмарних платформ, перед вибором правильної хмарної платформи для програми слід переглянути важливі параметри, такі як збір даних у реальному часі, формати візуалізації даних, тип хмарної служби, вартість розробки та сповіщень, а також моделі аналізу даних.

Крок 7. Джерело живлення: залежно від робочої потужності кожного датчика та апаратного забезпечення, для роботи датчиків можна використовувати перетворювачі живлення (SMPS) або джерела живлення. Переважно можна вибрати датчики з однаковими вимогами до потужності (12 В постійного струму або 24 В постійного струму).

Крок 8. Клапани та встановлення водопровідних ліній: нормально закриті електромагнітні клапани є стандартним і найбільш бажаним варіантом для регулювання

потоків води до кожної рослини в системі Smart Garden. Електромагнітні клапани вибираються на основі типу, МОС, витрати нагнітання та кількості параметрів керування (2- або 3-ходове керування).

Дотримуючись кроків, згаданих вище, користувачі можуть розробити та впровадити автоматизовану систему сільського господарства в приміщенні або на відкритому повітрі, яка відповідно контролює та регулює середовище для кожної рослини

Основними мотивами розробки Smart Garden є досягнення вищої врожайності з меншим споживанням води та добрив, а також створення якомога більшої частини автоматизованого процесу. Рішення можна застосовувати в різних приміщеннях і на відкритому повітрі, щоб зменшити витрати на оплату праці та заощадити час виробництва. Нижче наведено кілька переваг і сфер застосування сільського господарства на основі Інтернету речей:

1. Моніторинг посівів у режимі реального часу: встановлюючи різні датчики в полі, фермери можуть постійно контролювати стан своїх посівів і ґрунту. Ця інформація може допомогти їм прийняти своєчасне рішення щодо зрошення, удобрення та боротьби зі шкідниками.

2. Система управління зрошенням: за допомогою автоматизованих датчиків, які вимірюють рівень вологості ґрунту, фермери можуть спростити процес зрошення. Це економить воду та гарантує, що культури отримають потрібну кількість води в потрібний час.

3. Боротьба зі шкідниками та хворобами (та зведення до мінімуму використання пестицидів): Завдяки постійному моніторингу посівів фермери можуть на ранній стадії виявити зараження шкідниками та вжити заходів щодо їх усунення. Це допомагає звести до мінімуму використання пестицидів, що корисно не лише для навколишнього середовища, але й для здоров'я людей, які споживають ці культури.

4. Зниження витрат на робочу силу: системи на основі Інтернету речей можуть автоматизувати різні завдання, такі як зрошення, внесення добрив та боротьба зі шкідниками. Це зменшує потребу в ручній праці та експлуатаційні витрати.

5. Ефективне управління ланцюгом поставок: системи на основі Інтернету речей допомагають ефективніше керувати великими фермами, надаючи інформацію в реальному часі про різні аспекти сільського господарства. Це дозволяє фермерам приймати проактивні рішення, які підвищують ефективність і продуктивність.

6. Зменшення втрат (води та відходів урожаю): Завдяки постійному контролю за умовами посівів фермери можуть зменшити втрати через надмірний полив або неправильне зрошення, надмірне внесення добрив та зараження шкідниками.

7. Збільшення врожайності, прозорості та задоволеності клієнтів: системи на основі Інтернету речей генерують багато даних, які можна використовувати для відстеження прогресу сільськогосподарських операцій. Це сприяє збільшенню виробництва сільськогосподарської продукції, якості, прозорості та підзвітності, що зрештою призводить до кращого задоволення клієнтів, що є важливим для будь-якого бізнесу.

1.2 Інтернет речей для автоматизації садівництва

Інтернет речей (IoT) – це концепція мережі взаємопов'язаних фізичних пристроїв, що мають вбудовані датчики та засоби обміну даними. Ці пристрої, які часто називають «речами», підключені до інтернету та можуть взаємодіяти один з одним і з користувачами. IoT охоплює широкий спектр застосувань, від розумних будинків та міст до промислового контролю та охорони здоров'я. Головна ідея полягає в тому, щоб зробити фізичні об'єкти «розумними» та здатними генерувати та обмінюватися інформацією, що полегшує автоматизацію та прийняття рішень. [1].

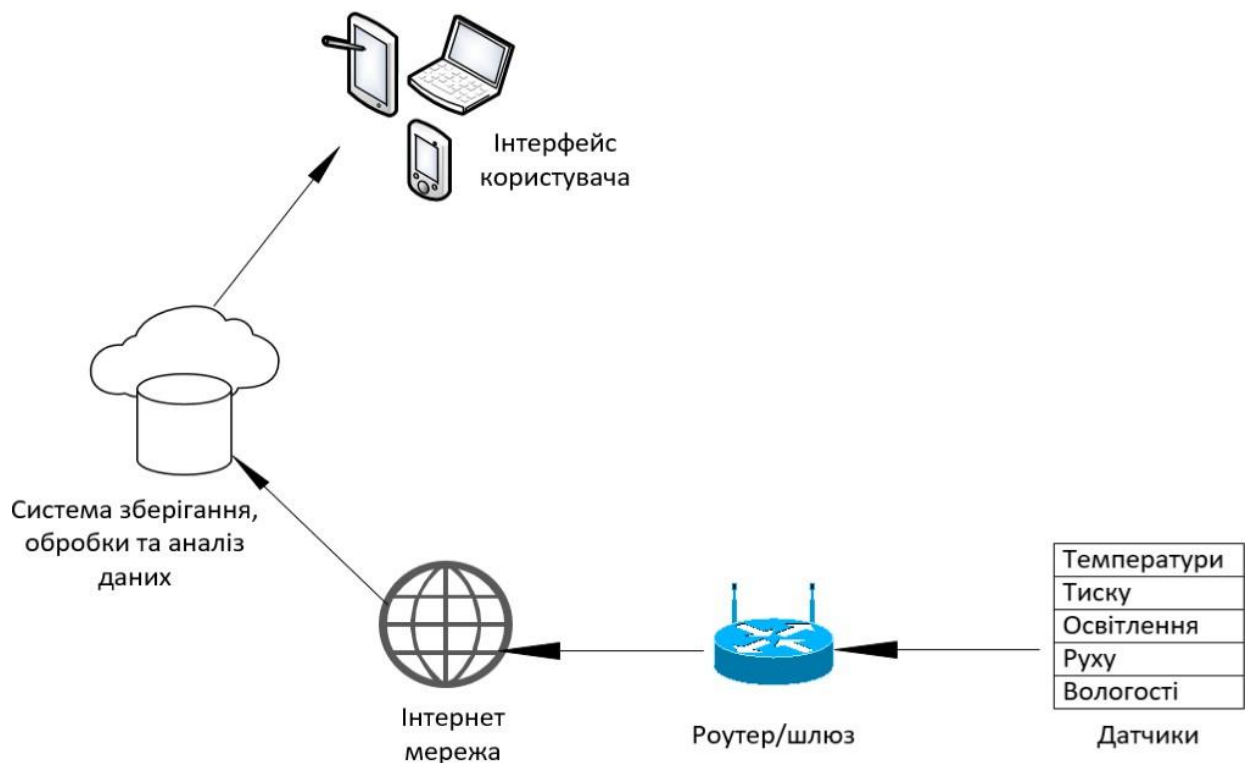


Рисунок 1.1 - Принцип роботи IoT

Архітектура IoT пристроїв наведена на рисунку 1.1.

Принцип роботи IoT (Інтернету речей) включає чотири основні рівні: фізичний рівень (датчики), мережевий рівень, рівень обробки даних та рівень додатків. Ці рівні працюють разом, щоб збирати дані з фізичних пристроїв, передавати їх, обробляти та представляти

користувачам. Існують також вертикальні рівні управління та безпеки, які забезпечують ефективне функціонування всієї системи та усунення можливих проблем. Датчики вимірюють різноманітні параметри, такі як вологість, температура, рух, тощо.

1) Рівень датчиків

Основним елементом IoT є рівень датчиків, які виконують ключову роль у зборі даних. Цей рівень включає різні типи датчиків та активаторів, які вимірюють фізичні параметри і передають отримані дані для подальшої обробки. Ключові аспекти, пов'язаних із рівнем датчиків у контексті IoT:

Основні категорії датчиків для збору даних:

Фізичні датчики:

Температурні датчики: Вимірюють температуру навколишнього середовища (наприклад, термометри).

Вологісні датчики: Вимірюють рівень вологи у повітрі або землі.

Датчики тиску: Вимірюють атмосферний тиск або тиск у рідинах.

Датчики руху: Виявляють рух об'єктів (наприклад, PIR-датчики).

Хімічні датчики:

Вимірюють склад повітря, води або інших середовищ. Наприклад, газоаналізатори можуть реєструвати наявність шкідливих газів.

Оптичні датчики:

Використовують світло для збору даних. Наприклад, камери відеоспостереження або датчики світла.

Активаційні датчики:

Усуваються для автоматизації процесів, наприклад, реле або моторизовані елементи, які активують або деактивують пристрої на підставі отриманих даних.

Збір і передача даних:

Збір даних: Датчики постійно моніторять навколишнє середовище та збирають дані, які здатні вказувати на зміни в стані об'єктів або середовища.

Передача даних: Зібрані дані передаються через бездротові (Wi-Fi, Bluetooth, LoRaWAN) або дротові (Ethernet) мережі до центральної системи або в "хмару" для аналізу.

Застосування:

Розумні будинки: Датчики, що контролюють температуру, освітлення, безпеку.

Промисловий IoT: Системи моніторингу та контролю виробництва.

Охорона здоров'я: Wearable devices, які відстежують здоров'я пацієнтів (наприклад, монітори серцевого ритму).

Таким чином, рівень датчиків у системах IoT є критично важливим для збору, обробки та аналізу даних, що робить можливим автоматизоване управління та прийняття рішень на основі отриманої інформації [2].

2) Мережевий рівень

Мережевий рівень IoT є критично важливим для забезпечення зв'язку між "речами", які збирають дані, і системами, які їх обробляють і аналізують.

Основні принципи мережевого рівня IoT:

Передача даних: Мережевий рівень відповідає за передачу зібраних даних від датчиків до серверів, "хмари" або центрів обробки даних. Це включає як бездротові, так і дротові технології.

Інтероперабельність: Забезпечення узгодженості і здатності різних пристроїв і систем працювати разом, незважаючи на те, що вони можуть бути вироблені різними компаніями або підтримувати різні протоколи.

Масштабованість: Мережеві технології повинні мати змогу підтримувати зростаючу кількість підключених пристроїв та обробляти величезний обсяг даних.

Безпека: Забезпечення конфіденційності, цілісності та доступності даних під час передачі, що є критично важливим у системах IoT.

Основні технології мережевого рівня IoT

Протоколи зв'язку:

MQTT (Message Queuing Telemetry Transport): Легкий протокол передачі повідомлень, що орієнтований на малосильні пристрої та невеликі обсяги даних.

CoAP (Constrained Application Protocol): Протокол, спеціально розроблений для обмежених пристроїв і мереж, який є ефективним для використання в IoT.

HTTP/HTTPS: Використовується для звичайного зв'язку через веб, але може бути менш ефективним для обмежених пристроїв.

Бездротові технології:

Wi-Fi: Поширена технологія для підключення пристроїв до інтернету в домогосподарствах та офісах.

Bluetooth/Bluetooth Low Energy (BLE): Використовується для короткочасного зв'язку між пристроями.

Zigbee: Технологія для бездротових мереж, що споживає мало енергії, зазвичай використовується в розумних будинках.

LoRaWAN: Низька потужність, довгі відстані, широкосмуговий зв'язок, що ідеально підходить для сільськогосподарських та промислових IoT-аплікацій.

NB-IoT (Narrowband IoT): Технологія мобільного зв'язку, що дозволяє підключати величезну кількість пристроїв в мережах.

Дротові технології:

Ethernet: Стандартна технологія для підключення пристроїв до локальних мереж.

RS-485: Використовується для з'єднання промислових пристроїв.

Основні виклики на мережевому рівні

Забезпечення безпеки: Вразливості в мережах можуть призвести до витоку даних або атаки на системи.

Масштабованість: Зі збільшенням кількості пристроїв потрібно гарантувати стабільність та продуктивність мережі.

Управління затримками: Для одних застосувань, таких як автономні автомобілі, потрібна низька затримка, що може бути складно реалізувати у великих мережах.

Мережева інфраструктура IoT є основою для успішного зв'язку і взаємодії між різними пристроями та системами. Справжній потенціал IoT реалізується лише тоді, коли мережевий рівень ефективно підтримує передачу даних, забезпечує безпеку та дозволяє різним пристроям працювати разом.

3) Рівень обробки даних

Рівень обробки даних в IoT є критично важливим для перетворення зібраної інформації на корисні знання та дії. Цей етап охоплює аналіз, зберігання і візуалізацію даних, щоб забезпечити ефективність і автоматизацію прийняття рішень.

Основні принципи рівня обробки даних в IoT

Збір даних: Інформація, що надходить з IoT-пристроїв, обробляється у реальному часі або із затримкою для подальшого аналізу.

Аналіз даних: Використовуються різні алгоритми і моделі, включаючи статистичні методи, машинне навчання (ML) і штучний інтелект (AI), щоб виявити шаблони, аномалії чи тенденції в даних.

Зберігання даних: Дані зберігаються в структурованих (реляційні бази даних) або неструктурованих (NoSQL) форматах для забезпечення легкого доступу до них для подальшої обробки.

Візуалізація даних: Візуалізаційні інструменти допомагають представити результати аналізу у зрозумілому вигляді, що полегшує прийняття рішень користувачами.

Основні технології рівня обробки даних в IoT

Аналіз даних:

Машинне навчання (ML) та штучний інтелект (AI): Алгоритми, що використовуються для прогнозування, класифікації та відстеження змін у даних. Наприклад, моделі для передбачення нештатних ситуацій на виробництві.

Обробка природної мови (NLP): Аналіз текстових даних, які можуть надходити з коментарів користувачів, відгуків чи інших джерел.

Бази даних:

Реляційні бази даних: MySQL, PostgreSQL, використовуються для зберігання структурованих даних.

NoSQL бази даних: MongoDB, Cassandra, які пропонують більшу гнучкість при зберіганні неструктурованих або напівструктурованих даних.

Time-series бази даних: InfluxDB, Prometheus, спеціалізовані на зберіганні та обробці даних, які змінюються з плином часу (наприклад, дані з сенсорів).

Хмарні сервіси:

Сервіси, вбудовані в хмарні платформи, такі як AWS, Microsoft Azure або Google Cloud, пропонують потужні інструменти для обробки, зберігання та аналізу даних.

IoT платформи:

Спеціалізовані платформи, такі як ThingSpeak, Particle або Blynk, що надають готові рішення для управління та обробки даних з IoT-пристроїв.

Застосування рівня обробки даних

Розумні будинки: Автоматизація систем керування (освітлення, опалення) на основі аналізу звичок користувачів.

Промисловий IoT: Моніторинг та прогнозування збоїв обладнання, що підвищує безпеку та знижує витрати.

Охорона здоров'я: Аналіз даних збору з медичних пристроїв для моніторингу стану пацієнтів у реальному часі.

Виклики рівня обробки даних

Обробка великих обсягів даних: Зростання кількості підключених пристроїв призводить до значних обсягів даних, що потребує потужних засобів обробки.

Забезпечення безпеки і конфіденційності: Необхідно захищати дані на всіх етапах - від збору до зберігання та обробки.

Точність аналізу: Забезпечити, щоб аналітичні алгоритми давали якомога точніші і коректні результати.

Рівень обробки даних у структурах IoT є серцем системи, що перетворює необроблені дані в цінну інформацію. Завдяки новим технологіям, таким як AI, ML і потужним

інфраструктурам для зберігання даних, організації можуть отримувати глибокі інсайти і приймати обґрунтовані рішення на основі даних.

4) Рівень додатків

Рівень додатків презентує результати роботи рівню обробки даних за допомогою різного роду додатків IoT пристроїв. Рівень додатків є рівнем орієнтованим на користувачів, який виконує різні завдання для користувача. Існують різні IoT додатки, які включають розумний будинок, розумний транспорт тощо.

1.3 Проблематика та постановка завдань дослідження

У суспільстві можна спостерігати тенденцію зростання використання органічних зелених продуктів, вирощених у домашніх умовах. Такий тренд стає популярним, оскільки у традиційному виробництві фермери використовують хімічно шкідливі речовини, що руйнують корисні елементи, що входять до складу продуктів харчування.

Незадоволені таким способом вирощування продуктів або люди які мають проблеми зі здоров'ям все частіше звертаються до органічних ферм або домашнього садівництва, де використовуються лише природні ресурси. Органічні ферми уникають додавання небезпечних хімічних добавок, таких як штучні інгредієнти, іонізуюче випромінювання, стічні води та генна інженерія, а також не використовують методи виробництва, які шкідливі для навколишнього середовища і, зокрема, для здоров'я людини. Даний метод допомагає знизити кількість відходів, тому вирішує проблему утилізації та переробки сміття, і споживає менше енергії, таким чином люди можуть легко вирощувати овочі та фрукти в домашніх умовах. Деякі люди можуть займатися садівництвом як хобі, в інших є фізичні обмеження, які не дозволяють їм доглядати свій сад поза домом. Міні ферми можна також використовувати у навчальних лабораторіях. Однак використання домашнього саду накладає деякі обмеження, наприклад, відсутність достатнього простору для комфортного життя рослин поруч із людиною, відсутність сонячного світла та своєчасного поливу, а також відсутність часу для забезпечення правильного догляду за рослинами.

Інтелектуальне садівництво та розведення в контейнерах – це спеціальні методи, адаптовані для пом'якшення наслідків цих обмежень, тому завдяки новим технологіям вирішується низка проблем, які раніше виконувались за безпосередньої участі людини, а тепер стали автоматизованими. Зараз існує багато технологій, що дозволяють вирощувати понад 100 видів зелених овочів та інших рослин на невеликій площі, але розміри таких установок усе ще досить великі.

Метою даного дослідження є розкриття принципів використання Інтернету речей для автоматизації садівництва.

За для досягнення поставленої мети у роботі необхідно виконати низку завдань:

- навести методи та моделі автоматизації садівництва;
- розробити алгоритм реалізації процесу автоматизації садівництва;
- запропонувати структурну схему IoT-системи для автоматизованого догляду за садом;
- здійснити вибір обладнання та програмного забезпечення;
- представити верифікацію результатів дослідження.

Висновки до розділу

У першому розділі даної роботи описано поняття Smart Garden, сутність та генезис, також представлено характеристику Інтернету речей для автоматизації садівництва, окреслено проблематику дослідження та здійснено постановку завдань.

РОЗДІЛ 2

МЕТОДИ ТА МОДЕЛІ АВТОМАТИЗАЦІЇ САДІВНИЦТВА

2.1 Методи автоматизації садівництва

Передача даних від датчиків Інтернет-речей (також, як і при прийомі даних) здійснюється за допомогою контролерів.

Контролери Інтернет-речей можуть бути реалізовані за допомогою різних апаратних засобів: мікропроцесорів, мікроконтролерів, уніфікованих наборів (конструкторів) типу Arduino або ін., одноплатних ЕОМ або одноплатних ПК (типу RaspberryPi)[3].

Мікроконтролер (англ. Micro Controller Unit, MCU) – це компактний комп'ютер, призначений для автоматизації управлінських задач в електронних пристроях. Він містить процесор, пам'ять, засоби введення/виведення (I/O) та інші елементи в одному чіпі.

Види мікроконтролерів

8-розрядні мікроконтролери:

Приклад: 8051, PIC1.

Використовуються в простих проектах, таких як управління світлодіодами або простими сенсорами.

16-розрядні мікроконтролери:

Приклад: PIC16, MSP430.

Застосовуються для більш складних завдань, де потрібна більша обчислювальна потужність, наприклад, в автоматизації побутових пристроїв.

32-розрядні мікроконтролери:

Приклад: ARM Cortex-M, PIC32.

Використовуються в складних проектах, таких як розумні пристрої, смартфони, інтернет речей (IoT).

FPGA (Field-Programmable Gate Array):

Доступні для програмування, що надає високу гнучкість і можливість реалізації різних логічних функцій.

SoC (System on Chip):

Інтегрує мікроконтролер, пам'ять і інші компоненти в одному чіпі. Використовується, наприклад, в мобільних телефонах та інших мобільних пристроях.

Принципи роботи мікроконтролерів

Процесор:

Основний компонент, який виконує команди і обробляє дані. Наприклад, арифметичні та логічні операції, обробка вхідних сигналів.

Пам'ять:

ROM (Read-Only Memory): Неперезаписувана пам'ять, що містить програму.

RAM (Random Access Memory): Оперативна пам'ять для зберігання тимчасових даних під час виконання програми.

Введення/виведення (I/O):

Порти, які відповідають за взаємодію з іншими пристроями. Включають цифрові та аналогові порти для роботи з сенсорами, двигунами, та іншою периферією.

Внутрішні таймери і перетворювачі:

Таймери для планування подій або створення затримок.

Аналогово-цифрові перетворювачі (ADC) для перетворення аналогових сигналів у цифрові.

Програмне забезпечення:

Мікроконтролери зазвичай програмуються мовами, такими як C, C++, або специфічними мовами високого рівня (наприклад, для Arduino).

Застосування мікроконтролерів

Бутікові електронні проекти: Виготовлення спеціалізованих або саморобних пристроїв.

Промислові контролери: Використання в автоматизації виробничих процесів.

Розумні будинки: Керування освітленням, опаленням, безпекою.

Автомобільна промисловість: Управління системами безпеки, двигунами, електронікою.

Медичні пристрої: Вимірювання життєвих показників пацієнтів, моніторинг стану здоров'я.

Мікроконтролери є основними компонентами сучасних електронних пристроїв. Їхня енергоефективність, компактні розміри і можливість програмування дозволяють реалізувати широкий спектр застосувань.

Мікропроцесор - процесор, реалізований у вигляді однієї мікросхеми або комплекту з декількох спеціалізованих мікросхем. Він містить в собі арифметико-логічний пристрій, блок синхронізації і управління, що запам'ятовують пристрій, регістри і шину. Тобто МП містить в собі тільки те, що безпосередньо знадобиться для виконання арифметичний і логічних операцій. Всі інші комплектуючі (ОЗУ, ПЗУ, пристрої введення / виводу, інтерфейси) потрібно підключати ззовні. Перевагою є те, що до МП можна на вибір підключати різну периферію з різними характеристиками (що не у всіх випадках можна на МК). Друге основна

відмінність мікропроцесора від мікроконтролера в тому, що МП мають більше обчислювальної потужності.

Плюси:

- Продуктивність - мікропроцесори мають більшу продуктивність, ніж мікроконтролери.

- Вибір - в разі МП у вас є можливість підібрати комплектуючі. Це дозволить поставити більш відповідну під ваші цілі периферію.

Arduino – це популярна платформа для створення електронних проектів. Вона складається з апаратної і програмної частини, що дозволяє легко розробляти прототипи та реалізовувати різноманітні рішення в галузі електроніки, робототехніки і автоматизації.

Принцип роботи Arduino

Апаратна частина:

Мікроконтролер: Основний компонент, що виконує програми. Найпоширеніші мікроконтролери на базі Arduino – це ATmega (для Arduino Uno) та ARM Cortex (для Arduino Due).

Порти введення/виведення (I/O): Arduino має цифрові та аналогові порти, через які можна підключати різноманітні сенсори, двигуни, світлодіоди та інші пристрої.

Живлення: Arduino може житися через USB або зовнішні джерела живлення (батареї, адаптери).

Програмна частина:

IDE Arduino (Integrated Development Environment): Програмне забезпечення, що використовується для написання, компіляції та завантаження програм на плату. Воно підтримує мову програмування, схожу на C/C++.

Бібліотеки: Arduino має безліч бібліотек, які полегшують роботу з різними компонентами та функціями, такими як дисплеї, датчики, двигуни тощо.

Цикл роботи:

Програма (скетч) на Arduino запускається, коли плата увімкнена. Вона перебуває в циклі, що складається з функцій `setup()` (виконується один раз при старті) і `loop()` (виконується безперервно).

Види Arduino

Arduino Uno:

Найпопулярніша та найпоширеніша плата, базується на ATmega328. Ідеально підходить для початківців.

Arduino Mega:

Має більше входів/виходів (54 цифрових I/O порти), більше пам'яті, що робить її чудовим вибором для складних проектів.

Arduino Nano:

Компактна версія Arduino Uno, призначена для проектів, де простір є обмеженим.

Arduino Due:

Використовує ARM Cortex-M3, пропонує потужнішу обробку, відповідну для більш складних завдань.

Arduino Leonardo:

Має можливість діяти як USB-імітатор (клавіатура, миша), що може бути корисним для інтерактивних проектів.

Arduino MKR:

Серія плат, орієнтованих на IoT, що включає бездротові модулі для підключення до інтернету (наприклад, Wi-Fi, LoRa, GSM).

Технології Arduino

Програмування:

Arduino використовує C/C++-подібну мову програмування, надаючи прості конструкції для роботи з hardware.

Бібліотеки:

Величезна кількість готових бібліотек для роботи з сенсорами, дисплеями, моторами та іншими компонентами.

Наприклад, бібліотеки `Wire.h` для I2C зв'язку, `Servo.h` для сервоприводів, `WiFi.h` для з'єднання з бездротовими мережами.

Зворотний зв'язок:

Arduino пропонує можливості для моніторингу та керування приладами, що забезпечує інтерактивність в проектах.

Спільнота:

Є активна спільнота розробників, що обмінюється проектами, кодом та порадами. Є багато ресурсів, форумів і навчальних матеріалів.

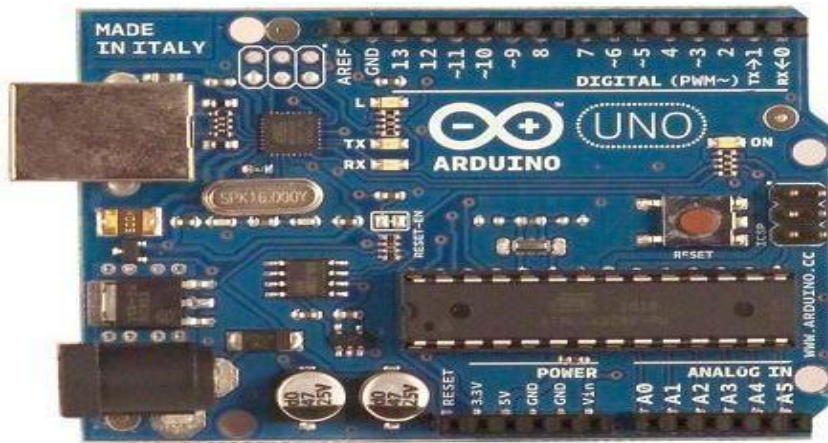


Рисунок 2.1 – Arduino UNO

Інтегроване середовище розробки Arduino (рис.2.1) - це кроссплатформний додаток на Java, що включає в себе редактор коду, компілятор і модуль передачі прошивки в плату. Середовище розробки заснована на мові програмування Processing, а мова програмування мікроконтролерів аналогічний мови, що використовується в проєкті Wiring. Тут програми обробляються за допомогою препроцесора, а потім компілюється за допомогою AVR-GCC. Програмне забезпечення включає багато бібліотек для роботи з зовнішніми пристроями[5].

Raspberry Pi 3 Model B + – це мікрокомп'ютер від Raspberry Pi Foundation, який працює на основі оновленого 4-ядерного 64-розрядної SoC Broadcom BCM2837B0 і збільшеною тактовою частотою 1,4 ГГц.

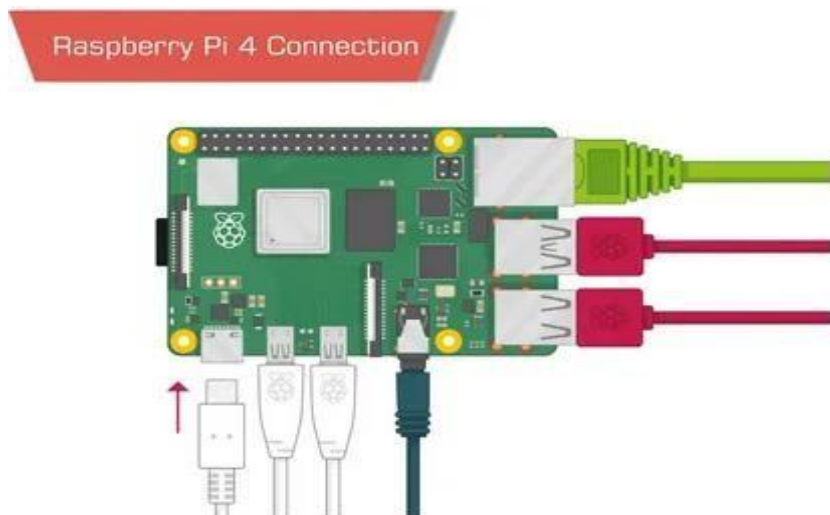


Рисунок 2.2– Raspberry Pi 4

Зв'язок контролерів може здійснюватися різними способами. Для віддаленого управління пристроями, зняття параметрів з датчиків найчастіше використовуються контролерів AVR, MSP430 та STM32.

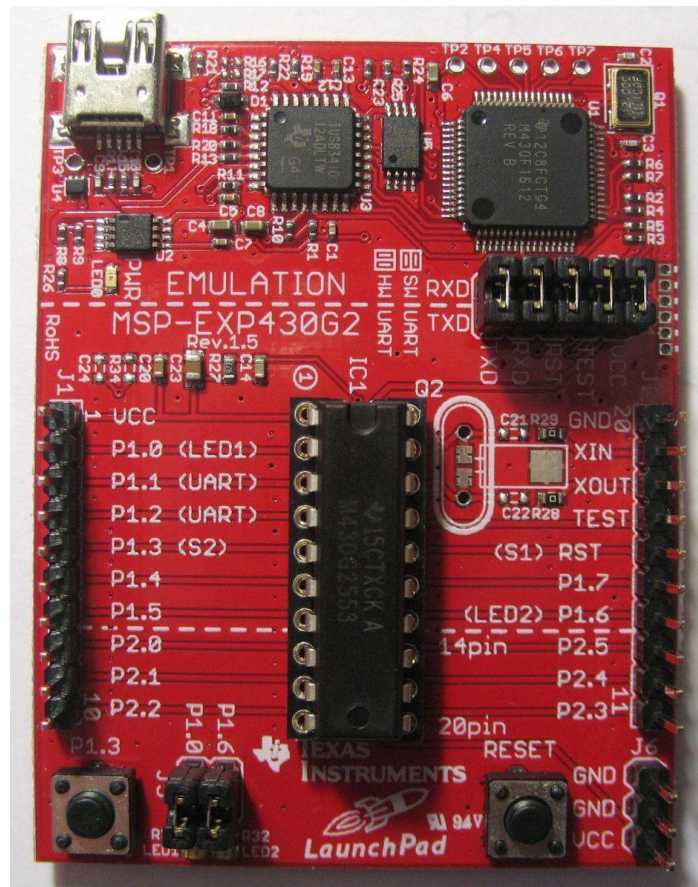


Рисунок 2.4— MSP-EXP430G2 LaunchPad

Для підключення контролерів до мережі найбільш популярні мікросхеми W5100(компанії Wiznet) і ENC28J60.

Технічні параметри Ethernet Shield W5100:

- Напруга живлення: 5 В
- Внутрішній буфер: 16 кбіт
- Швидкість підключення: 10 і 100 Мбіт / с (RJ45)
- Роз'єм для карт пам'яті: micro-SD
- Габарити: 73 мм x 54 мм x 27 мм

Щоб підключити Ethernet Shield W5100, на платі передбачені довгі штирові роз'єми, так само поверх W5100 можна встановити інші Shields. Для зберігання великих файлів на платі встановлений роз'єм micro-SD, для установки micro-SD карт. Плата Arduino взаємодіє як з W5100, так і з SD-картою, використовуючи шину SPI, на обох платах висновки 10 використовується для вибору W5100 і висновки 4 для вибору SD-карти, ці виведення можна використовувати. Так-так W5100 і SD-карта використовують одну шину SPI, одночасно може бути активно тільки один пристрій. для підключення до маршрутизатора або комп'ютера, встановлений стандартний роз'єм RJ45.

ENC28J60 – це Ethernet-контролер, який дозволяє мікроконтролерам підключатися до Ethernet-мережі, забезпечуючи можливість передачі даних через Інтернет. Цей чип часто використовується в проектах, де потрібна можливість підключення до Інтернету або локальної мережі, особливо у проектах на базі Arduino та інших мікроконтролерів.

Основні характеристики ENC28J60

Шина SPI:

Використовує SPI (Serial Peripheral Interface) для зв'язку з мікроконтролером, що забезпечує швидку передачу даних.

Потужність:

Працює з напругою 3.3V, але може витримувати напругу до 5V на входах.

Швидкість передачі:

Максимальна швидкість передачі даних до 10 Мбіт/с.

Вбудована пам'ять:

Має інтегровану пам'ять (до 8К байт) для зберігання Ethernet-кадру.

Протоколи:

Підтримує різні комунікаційні протоколи на рівні мережі, включаючи ARP, IP, UDP, TCP, ICMP, і PING.

Діапазон температур:

Підходить для використання в умовах з різними температурами.

Принцип роботи ENC28J60

Спілкування через SPI:

ENC28J60 з'єднується з мікроконтролером через SPI, що дозволяє швидко обмінюватися даними. Мікроконтролер може читати чи записувати регістри ENC28J60 через цю інтерфейс.

Обробка Ethernet-кадрів:

ENC28J60 автоматично обробляє Ethernet-кадри. Це означає, що мікроконтролеру не потрібно реалізовувати обробку низькорівневих шапок даних.

Конфігурація:

Хоча чип вже має вбудовану логіку, він все ще вимагає налаштування, яке виконується через SPI. Мікроконтролер повинен ініціювати мережеві протоколи та приймати або відправляти дані.

Взаємодія з програмним забезпеченням:

Для спрощення роботи з ENC28J60 існують готові бібліотеки для Arduino, Наприклад, бібліотека `UIPEthernet`, яка надає високий рівень абстракції для роботи з мережевими протоколами.

Застосування ENC28J60

ІоТ-проекти: Використовується для підключення мікроконтролерів до Інтернету, що дозволяє реалізовувати проекти в області Інтернету речей (ІоТ).

Розумні датчики: Здатність надсилати дані з датчиків на сервер або приймати команди від нього.

Веб-сервери: Може бути використаний для створення веб-серверів на базі мікроконтролерів, які можуть обробляти запити і надавати інформацію.

Віддалене управління: Використовується в системах, де потрібен віддалений моніторинг і управління пристроями.

Підключення ENC28J60 до Arduino

Модулі:

Багато модулів ENC28J60 вже доступні на ринку та оформлені у вигляді готових плат, що значно спрощує підключення.

Включення:

Необхідно підключати модулі до Arduino через SPI порти (MISO, MOSI, SCK, CS) та забезпечити живлення (3.3V).

До провідних інтерфейсів передачі даних відносять:

– SPI (послідовний периферійний інтерфейс) - це синхронний протокол послідовної передачі даних, який використовується для зв'язку мікроконтролера з одним або декількома периферійними пристроями. Інтерфейс SPI відрізняється відносно високою швидкістю і призначений для зв'язку близько розташованих пристроїв. Він також може використовуватися для взаємодії двох мікроконтролерів.;

– UART – (Universal Asynchronous Receiver-Transmitter, або Універсальний Асинхронний приймач) використовується для організації зв'язку комп'ютера з різними цифровими пристроями в електроніці. Інтерфейс перетворює дані, що передаються в послідовний код так, щоб була можлива їх передача по одній цифровій лінії іншому електронному пристрою. Прямого опису протоколу UART все таки немає, але його непрямий опис можна побачити в стандартах на протоколи RS-232, RS-422, RS-423, RS-485.

– I2C (Стенди для міжінтегральної схеми) - це протокол послідовного зв'язку, аналогічний UART, який не використовується для зв'язку між ПК та пристроями, застосовується з модулями та датчиками. Це проста двостороння синхронізована послідовна шина, для якої потрібно всього два проводи для передачі інформації між пристроями, які підключені до шини. Вони корисні для проектів, що вимагають спільної роботи безлічі різноманітних класів (наприклад, датчиків, виводів, розширень та драйверів), оскільки вони можуть підключити матеріальну плату до 128 пристроїв, та зберегти при цьому чіткий

зв'язок, I2C використовує адресну систему та загальну шину для безлічі різноманітних пристроїв, які можуть бути підключені з використанням одних і тих же дротів, а також всі дані передаються по одному дроту і мають невелику кількість контактів. Швидкість I2C також залежить від швидкості передачі даних.

– Протокол I2C також використовується для двопровідного інтерфейсу для підключення низькошвидкісних пристроїв, таких як мікроконтролери, EEPROM, аналогово-цифрові та цифрово-аналогові перетворювачі, інтерфейси вводу-виводу та інші подібні периферійні пристрої у вбудованих системах.

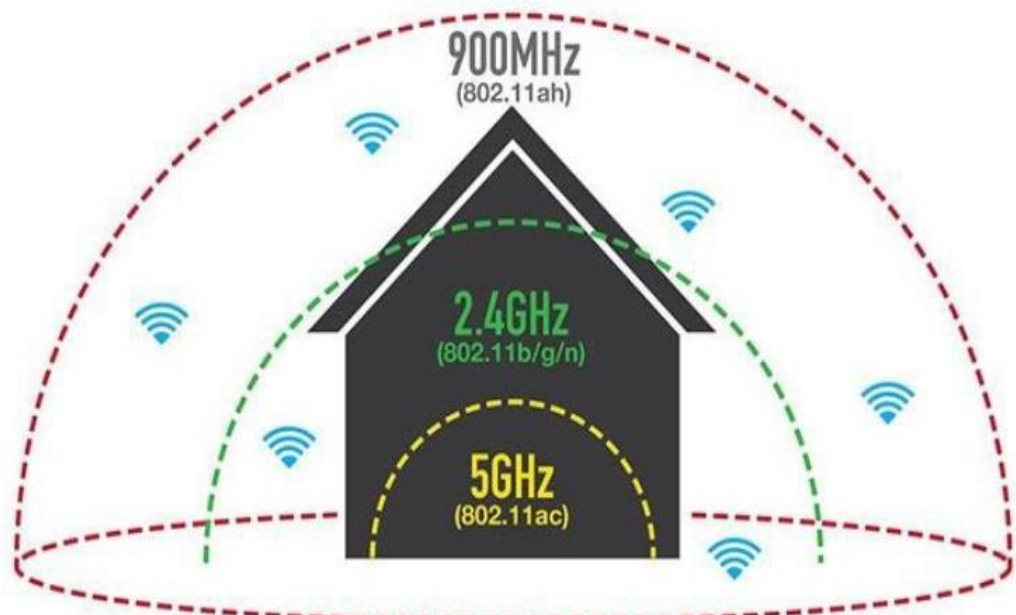


Рисунок 2.5– Порівняння різних протоколів Wi-Fi за зоною покриття

Таблиця 2.1– Різновиди стандарту Wi-Fi

Стандарт	802.11b	802.11g	802.11ac	802.11n	802.11ah
Частотний діапазон, ГГц	2,4	2,4	5	2,4/5	0,9
Максимальна швидкість передачі даних в радіоканалі, Мбіт/с	11	54	54	150-600	7000

Плюси та мінуси.

Деякі ключові особливості стандарту Wi-Fi HaLow. До його переваг відносяться:

- мале енергоспоживання і можливість роботи протягом тривалого часу від автономних джерел живлення;
- великий радіус передачі даних;
- компактність;
- захист переданих даних.

Недоліки:

- відносно висока вартість (в порівнянні з Bluetooth і ZigBee);
- слабка потужність сигналу.

2.2 Моделі автоматизації садівництва

Автоматизація садівництва – це використання різних технологій і систем для покращення процесів вирощування рослин, контролю за умовами середовища та управління ресурсами. Це може включати автоматичні системи поливу, моніторинг стану рослин, управління освітленням, а також системи збору та аналізу даних. Ось кілька основних моделей автоматизації садівництва.

1. Системи автоматичного поливу

Крапельний полив: Технологія, що забезпечує подачу води безпосередньо до кореневої системи рослин. Автоматизовані системи можуть використовувати датчики вологості, щоб визначити, коли поливати рослини.

Системи спринклерного поливу: Використовують розпилювальні пристрої для рівномірного розподілу води по всій ділянці. Можуть бути автоматизовані для роботи за графіком або на основі даних з погодних станцій.

2. Моніторинг стану рослин

Датчики вологи: Встановлюються в ґрунті для вимірювання рівня вологості. Вони допомагають визначити, коли потрібно поливати рослини.

Датчики світла: Використовуються для моніторингу рівня освітлення. Це може допомогти автоматично регулювати освітлення для рослин в закритих приміщеннях або теплицях.

Температурні і вологісні датчики: Використовуються для контролю умов навколишнього середовища. Можна автоматизувати вентиляцію та опалення в теплицях.

3. Системи управління

Централізовані системи контролю: Використовують програмне забезпечення або мобільні додатки для моніторингу та управління всіма аспектами автоматизації садівництва з одного місця. Це може включати контроль систем поливу, освітлення, вентиляції тощо.

Розумні контролери: Можуть бути використані для програмування таймерів, запуску і зупинки поливу або освітлення в залежності від умов навколишнього середовища або потреб рослин.

4. Системи обробки та аналізу даних

IoT-датчики: Використовуються для збору даних про різні параметри (вологість, температура, освітленість). Зібрані дані можуть бути надіслані на сервер або в "хмару" для тривалого зберігання і аналізу.

Аналіз даних: Використання алгоритмів машинного навчання для прогнозування потреб у воді, добривах, і діагностики стану рослин на основі аналізу зібраних даних.

5. Роботизовані системи

Автономні роботи для садівництва: Роботи, які можуть автоматично виконувати такі завдання, як посадка рослин, поливання, збирання урожаю. Вони здатні працювати самостійно, використовуючи сенсори та штучний інтелект для навігації та виконання робочих завдань.

6. Технології вертикального садівництва

Системи гідропоніки: Автоматизовані системи, що вирощують рослини без ґрунту, використовуючи поживний розчин. Вода і поживні речовини автоматично подаються до коренів рослин.

Аераційні системи: Застосовуються для забезпечення кореневої системи достатньою кількістю кисню, що є критично важливим для здоров'я рослин у закритих системах.

До основних моделей автоматизації садівництва варто віднести LoRaWAN і SIGFOX – це два протоколи для бездротового зв'язку, які використовують технології для малопотужних, низькошвидкісних IoT (Internet of Things) пристроїв. Обидва рішення націлені на використання в ситуаціях, де необхідна довга дистанція зв'язку, низьке споживання енергії та тривала автономність пристроїв. Давайте розглянемо їх характеристики, переваги та недоліки.

LoRaWAN (Long Range Wide Area Network)

LoRaWAN – це протокол, що базується на технології LoRa (Long Range), призначений для передачі даних на великі відстані з малим споживанням енергії.

Основні характеристики:

Довга дальність зв'язку: LoRaWAN може забезпечувати зв'язок на дистанції від кількох кілометрів у міських районах до понад 15 км у сільській місцевості.

Низьке споживання енергії: Підходить для пристроїв з тривалим терміном служби (багато років), оскільки пристрої можуть працювати на батареях та автоматично переходити у сплячий режим.

Топологія: Використовує архітектуру з базовими станціями (гейтвеями), які передають дані до центрального сервера. Пристрої зв'язуються з базовою станцією за допомогою енергоефективного радіозв'язку.

Скаленість: Легко масштабувати для тисяч пристроїв у мережі.

Переваги:

Висока ймовірність доставки даних на великі відстані.

Підтримка великих кількостей пристроїв (кілька тисяч на одну базову станцію).

Відкриті специфікації для реалізації та інтеграції.

Недоліки:

Низька швидкість передачі даних (від кількох біт до кількох кілобіт на секунду).

Користувач може зустрітися з різними технічними питаннями, пов'язаними з реалізацією.

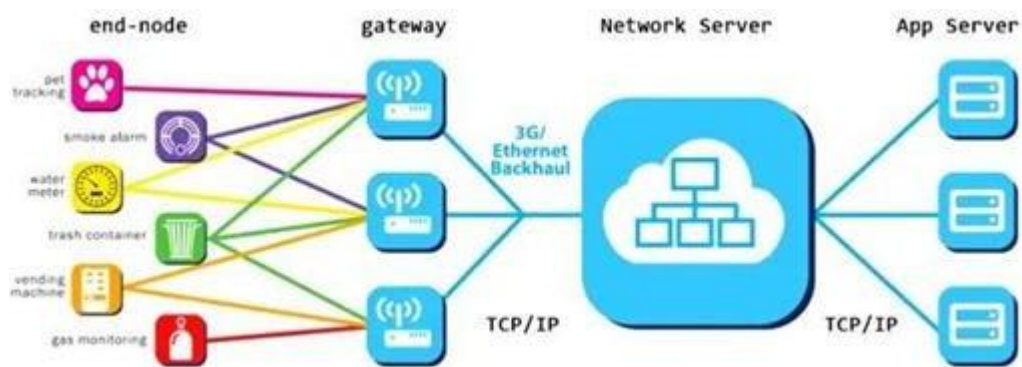


Рисунок 2.6 – архітектура LoRaWAN

SIGFOX

SIGFOX – це також технічне рішення для низькошвидкісного зв'язку в IoT, яке забезпечує глобальну мережу для обміну даними між пристроями.

Основні характеристики:

Досить довга дистанція: SIGFOX здатний забезпечити зв'язок на відстані до 30 км у сільських районах і до 10 км у міських умовах.

Низька швидкість передачі: Передача даних обмежена (до 100 байтів за повідомлення й до 140 повідомлень на день на пристрій).

Централізована інфраструктура: SIGFOX власники мають контроль над мережею, що забезпечує єдину точку доступу до сервісів.

Низьке споживання енергії: Подібно до LoRaWAN, пристрої SIGFOX можуть працювати на батареях протягом декількох років.

Переваги:

Простота реалізації і використання з комфортною архітектурою.

Висока енергоефективність та довгий термін служби батарей.

Недоліки:

Обмеження на обсяги даних, які можна передавати (обмежене використання).

Закрита архітектура, що може обмежити варіанти для інтеграції та налаштування.

Залежність від розгортання мережі SIGFOX по всьому світу.

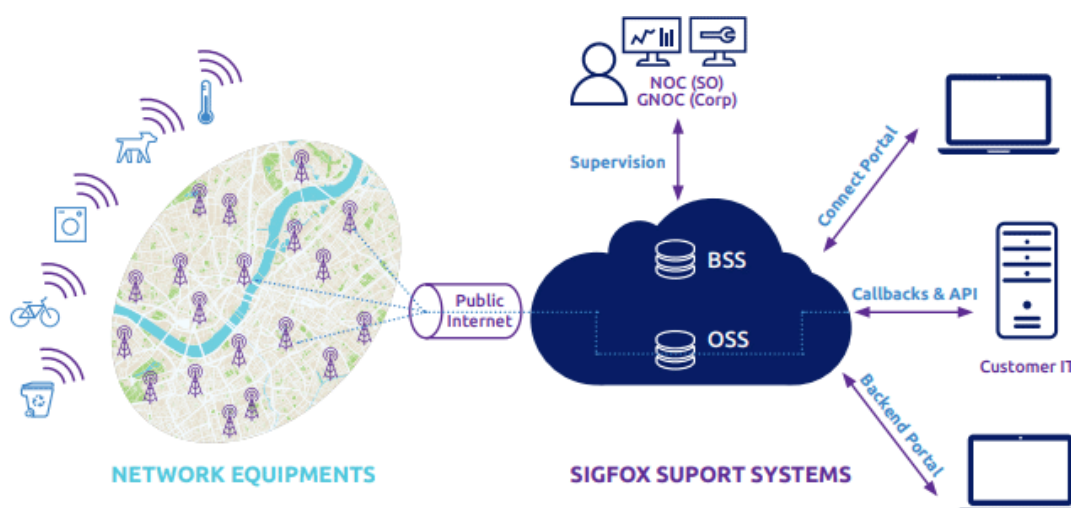


Рисунок 2.7– Топологія мережі SIGFOX

Таблиця 2.2 – Порівняння LoRaWAN і SIGFOX

Характеристика	LoRaWAN	SIGFOX
Дальність	До 15 км в сільській місцевості	До 30 км в сільських районах
Швидкість передачі	До 50-300 кбіт/с	До 100 байт за повідомлення
Інфраструктура	Відкрита	

NB-IoT, який також відомий як стандарт LTE-CAT.M2, має низку значних переваг: наднизьке енергоспоживання, що дозволяє працювати на одній батареї до 10 років, широку зону покриття, можливість швидкої модернізації існуючих мереж та високу надійність зв'язку.

Одним із варіантів розміщення NB-IoT є використання виділеного частотного каналу шириною 200 кГц. Цей підхід забезпечує найкращі умови для роботи технології, однак потребує значних витрат спектру — від 300 до 600 кГц з урахуванням захисних інтервалів. У такому випадку ризик взаємних перешкод з іншими технологіями мінімізується (рис. 2.8).

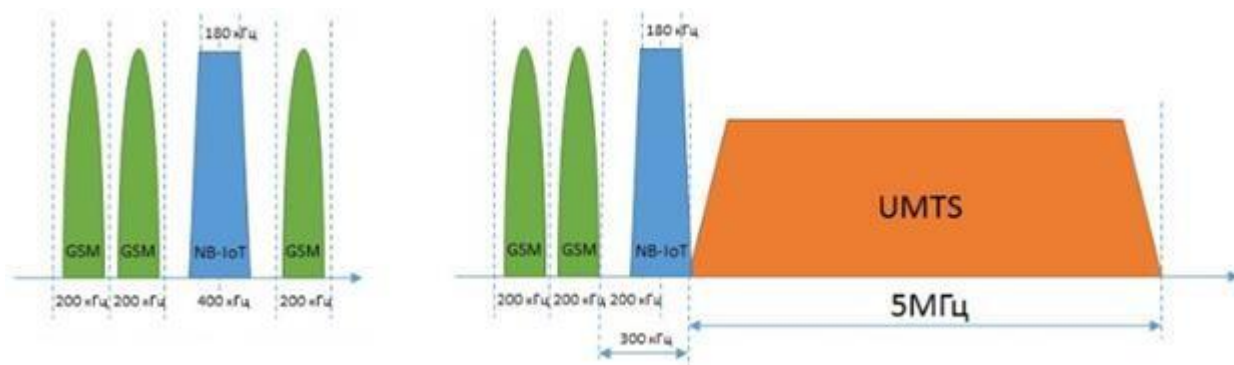


Рисунок 2.8 - Варіанти розміщення NB-IoT у режимі Stand-Alone

1) In-band. В цьому випадку для NB-IoT виділяються ресурси всередині існуючої LTE несучої, але NB-IoT несуча має підвищену потужність на 6дБ в порівнянні з ресурсними блоками LTE. Цей варіант добре підходить для економії частотного ресурсу, але при цьому є проблема взаємного впливу з LTE-мережею (рис. 2.9).

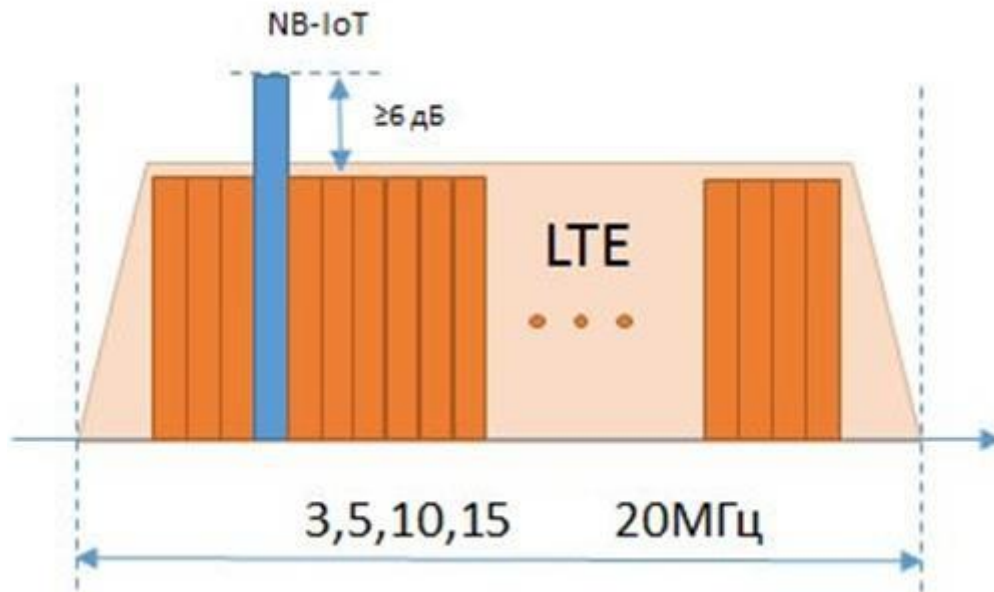


Рисунок 2.9 - Розміщення Nb IoT в режимі in Band

2) Guard-Band

В цьому випадку NB-IoT запускається в так званому захисному інтервалі. Наприклад, в смузі LTE 10 МГц, по 500 кГц вільного спектру, що використовується в якості захисного інтервалу. Так само як і в режимі in-band для більшої дальності NB-IoT-несуча має підвищену потужність на 6-9 дБ в порівнянні з ресурсними блоками LTE (рис. 2.10).

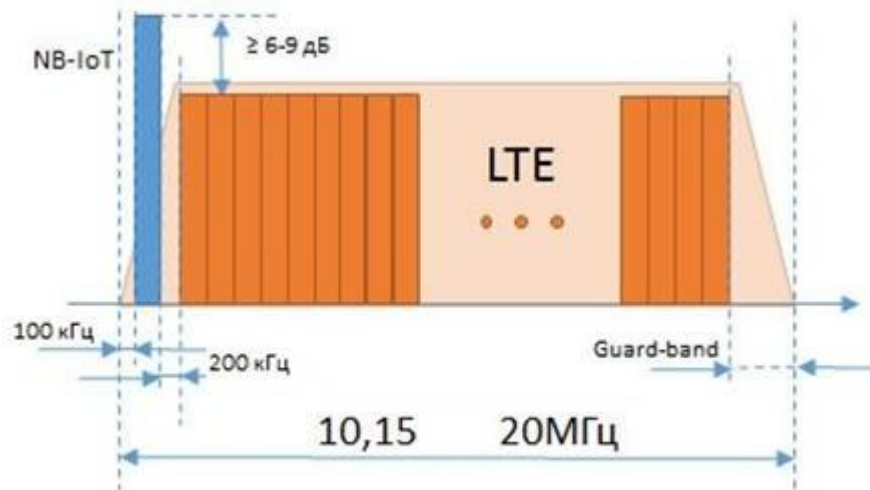


Рисунок 2.10 - Розміщення NB-IoT в режимі guard-band

Таким чином, мережі NB-IoT можуть бути розгорнуті у частотних діапазонах, які на даний момент використовуються для стандарту GSM, після їх рефармінгу під LTE, а також у захисних інтервалах між мережами GSM та LTE. Максимальна швидкість передачі даних у мережах NB-IoT становить до 200 кбіт/с, що є цілком достатнім для обслуговування пристроїв, які періодично передають невеликі обсяги однотипної інформації.

2.3 Алгоритм реалізації процесу автоматизації садівництва

Алгоритм роботи Smart Garden представлений на рис. 2.11.

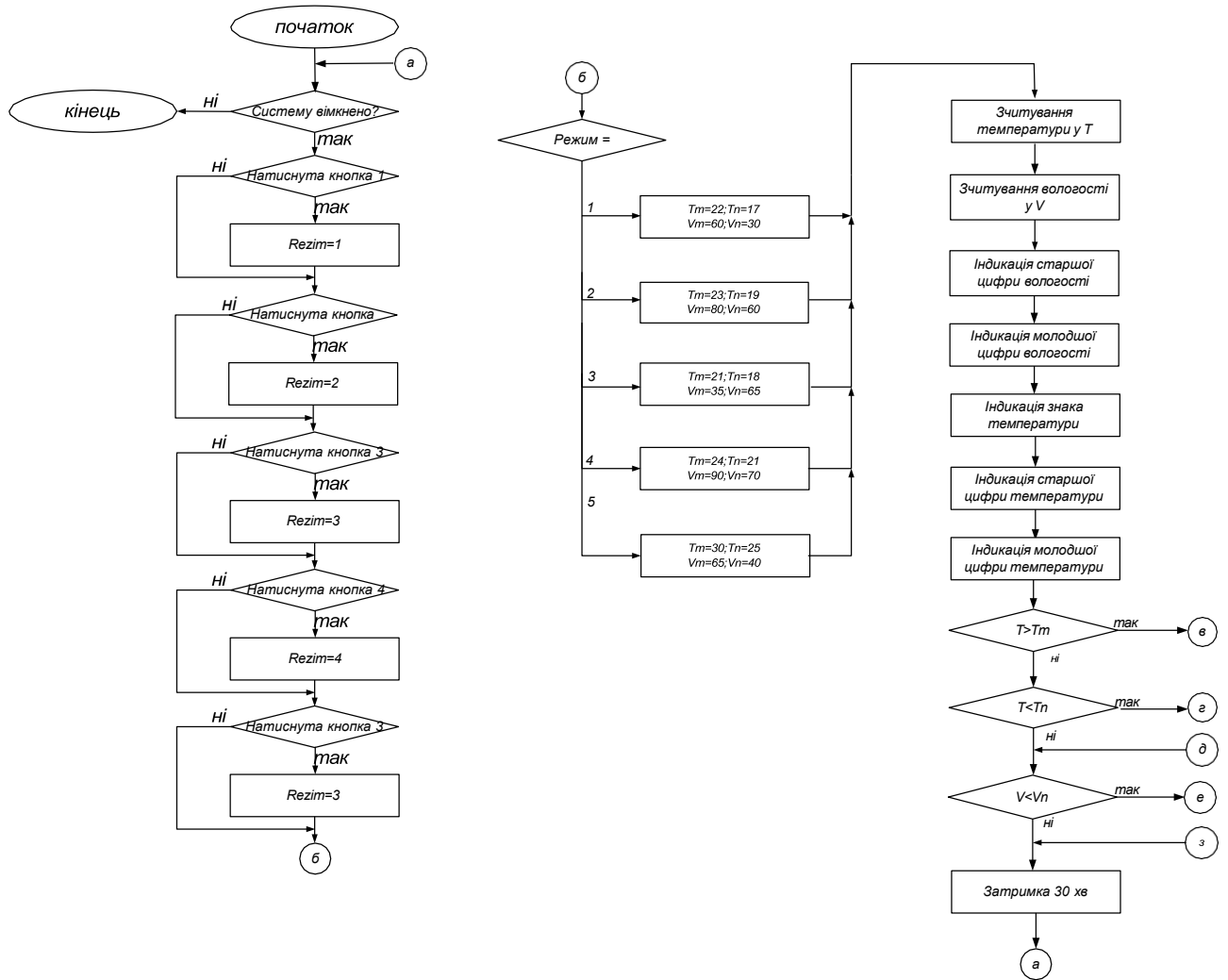
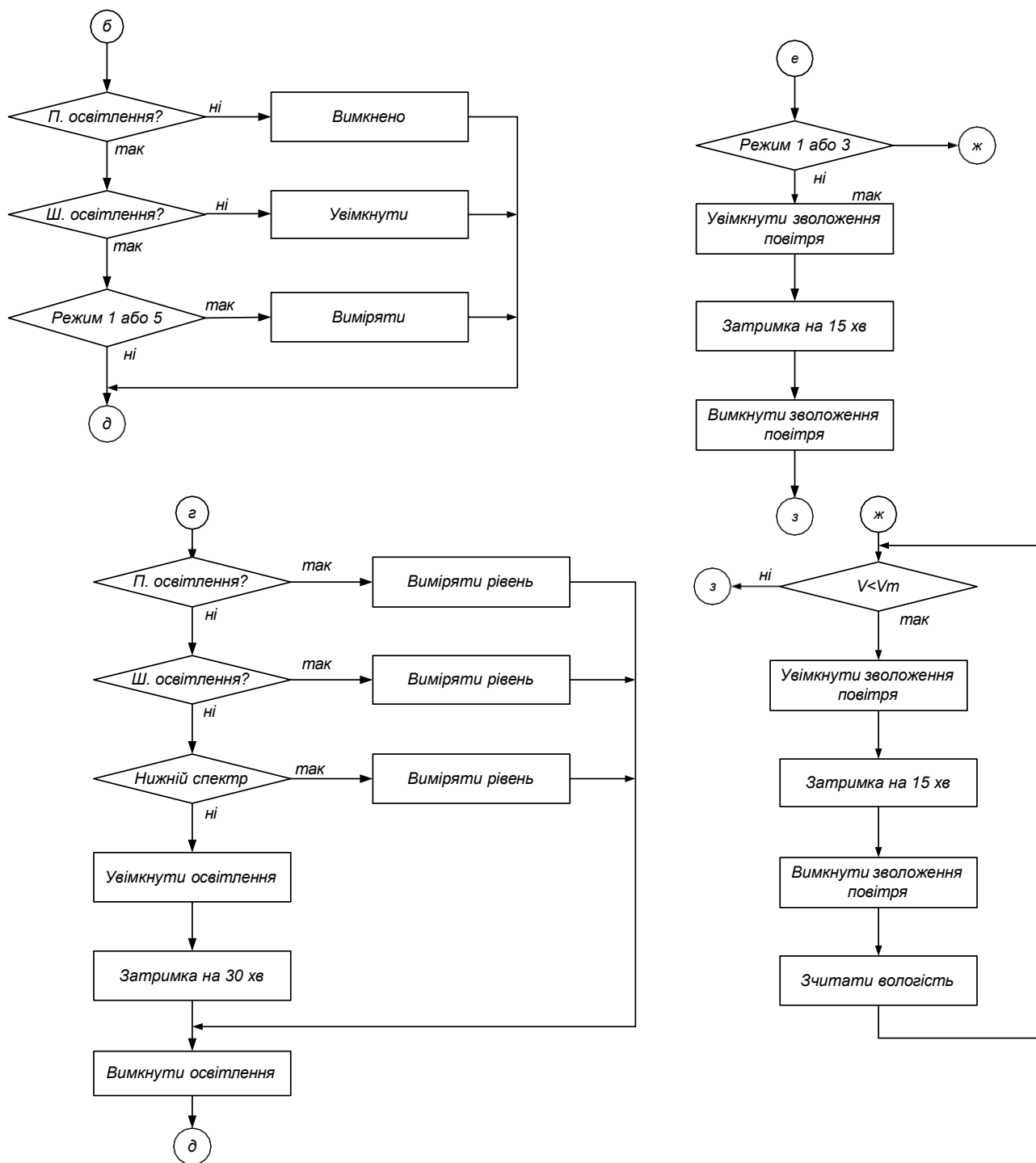


Рисунок 2.11 – Алгоритм роботи Smart Garden



Продовження рисунку 2.11

ІоТ-система для автоматизованого догляду за садом дозволяє встановлювати критичні параметри відповідно до обраного режиму, регулює температуру і вологість, враховуючи особливості реалізації.

Висновки до розділу

У рамках другого розділу розроблено алгоритм роботи IoT-системи для автоматизованого догляду за садом, яка дозволяє встановлювати критичні параметри відповідно до обраного режиму, регулює температуру і вологість, враховуючи особливості реалізації.

РОЗДІЛ 3

РОЗРОБКА ІОТ-СИСТЕМИ ДЛЯ АВТОМАТИЗОВАНОГО ДОГЛЯДУ ЗА САДОМ

3.1 Структурна схема ІоТ-системи для автоматизованого догляду за садом

Для розробки структурної схеми ІоТ-системи для автоматизованого догляду за садом, коротко опишемо функції, які повинна виконувати система, що розробляється:

1. Початковий запуск системи
2. Вибір необхідного для підтримки типу мікроклімату.
3. Прийом даних з датчиків і обробка цих даних відповідно до алгоритму.
4. Виведення поточних параметрів мікроклімату середовища.
5. Формування вихідних сигналів для запуску виконавчих пристроїв освітленості, нагріву, зволоження.

Виходячи з вимог технічного завдання та функцій, які повинна виконувати система, що розробляється, можна виділити основні модулі, з яких повинна складатися обчислювальна система.

Датчики – є невід'ємною частиною системи, вони використовуються для того, щоб система могла в реальному часі реагувати на зміни зовнішніх параметрів по заздалегідь розробленому алгоритму. В даний час на ринку представлена велика різноманітність різних типів датчиків, деякі з них є вузькоспеціалізованими. У межах даної системи використано датчики вологості ґрунту, датчики температури та вологості повітря, фоторезистори/датчики освітленості, насоси.

Система складатиметься з мікроконтролера, сенсорів і виконавчих механізмів.

Відповідно до визначених вище функцій можна визначити загальну структуру системи. Структурна схема ІоТ-системи для автоматизованого догляду за садом представлена на рис. 3.1.

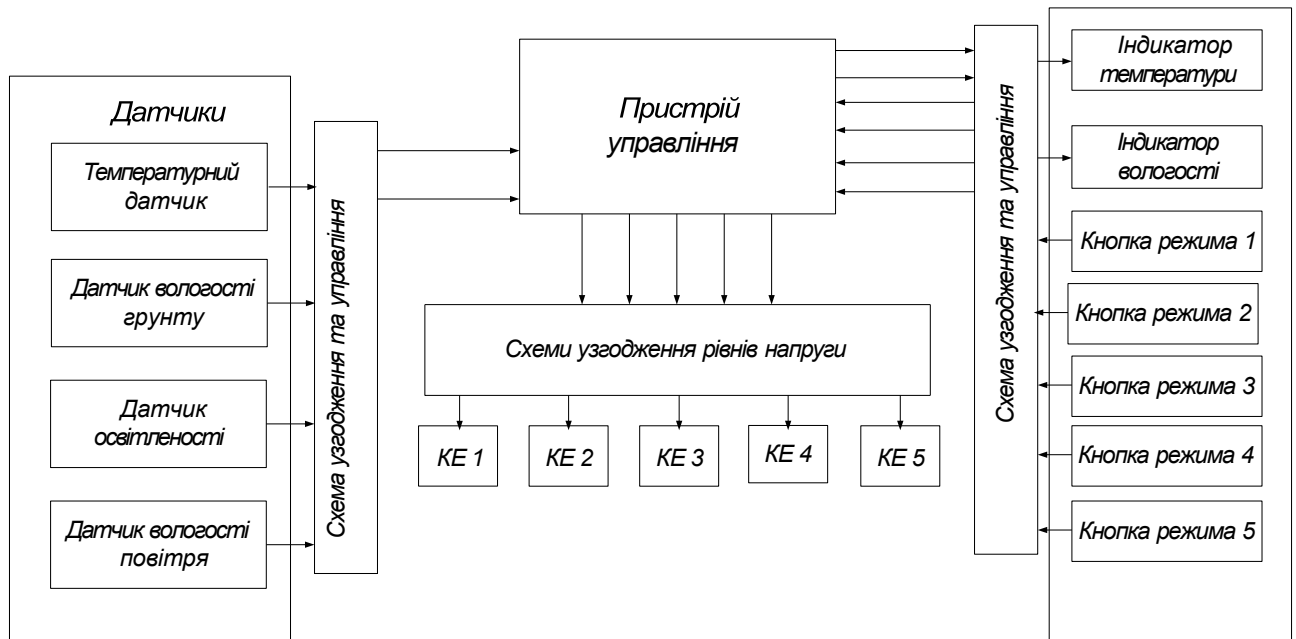


Рисунок 3.1 – Структурна схема IoT-системи для автоматизованого догляду за садом

Пристрій управління отримує від датчиків освітленості, температури, вологості і кнопок управління дані, перетворює їх відповідно до алгоритму роботи і видає дані на індикатори для відображення температури і вологості, а також при необхідності сигнали на ключові елементи. Ключові елементи дозволяють вмикати / вимикати виконавчі пристрої в тому порядку, в якому встановлено в алгоритмі.

3.2 Вибір обладнання та програмного забезпечення

У якості плати обираємо Raspberry Pi. Raspberry Pi – це недорогий одноплатний комп'ютер, який був розроблений для цілей навчання програмування та комп'ютерних наук. Проте з роками Raspberry Pi здобув популярність у багатьох інших сферах, зокрема, в електроніці, IoT (Internet of Things), робототехніці, медіа-центрах, проектах автоматизації та навіть у розробці промислових рішень.

Апаратна архітектура:

Процесор: Raspberry Pi зазвичай використовує ARM-процесори, які є енергоефективними та потужними для невеликих комп'ютерних завдань. Наприклад, Raspberry Pi 4 оснащений чотирьохядерним процесором ARM Cortex-A72.

Оперативна пам'ять: Оперативна пам'ять варіюється в залежності від моделі (від 1 ГБ до 8 ГБ у Raspberry Pi 4). Вона використовується для зберігання даних, які обробляються процесором під час роботи.

Пам'ять: Raspberry Pi використовує MicroSD картки для зберігання операційної системи і даних. У нових моделях Pi також можна використовувати USB-накопичувачі.

Графіка: Вбудовані графічні процесори забезпечують можливість виводу зображення на екрани через HDMI.

Введення/виведення (I/O):

GPIO (General Purpose Input/Output): Raspberry Pi має набір універсальних I/O пінів, які можуть використовуватися для підключення різноманітних сенсорів, виконавчих механізмів (моторів, реле, світлодіодів тощо) і інших периферійних пристроїв.

USB-порти: Використовуються для підключення різних пристроїв, таких як клавіатури, миші, флеш-накопичувачі, веб-камери.

HDMI: Для підключення до монітора або телевізора, передача візуальних сигналів.

Аудіовиходи: Raspberry Pi може передавати аудіо через HDMI або 3,5 мм роз'єм.

Програмне забезпечення:

Raspberry Pi використовує різні операційні системи, однак найпопулярніша – Raspberry Pi OS (раніше Raspbian), що базується на Debian. Існують також інші варіанти ОС, такі як Ubuntu, OSMC (для медіа-плеєрів), LibreELEC, Windows 10 IoT Core та інші.

У Raspberry Pi є змога використовувати мови програмування, такі як Python, Java, C/C++, Scratch та інші. Python, зокрема, став стандартом у навчанні програмування завдяки своїй простоті.

Технології Raspberry Pi

IoT (Internet of Things):

Raspberry Pi може використовуватися для створення IoT-проектів завдяки своїм можливостям підключення до мережі (Ethernet, Wi-Fi). Це дозволяє контролювати і моніторити віддалені пристрої.

Мультимедійні рішення:

Raspberry Pi може бути перетворений на медіа-центр, використовуючи такі платформи, як Kodi або Plex. Це дозволяє відтворювати відео та музику.

Робототехніка:

Raspberry Pi часто використовується в робототехнічних проектах як "мозок" робота, де він управляє сенсорами, двигунами та іншими компонентами.

Офісні та навчальні застосування:

Raspberry Pi може виконувати багато з офісних завдань, таких як обробка текстів, електронних таблиць, переглядання веб-сторінок і навіть програми для організації.

Системи управління:

Можливість контролювати технологічні процеси в садівництві, автоматизації будинку.

На рисунку 3.2 зображена остання на момент написання кваліфікаційної роботи модифікація Raspberry Pi 4B, запущена в продаж у червні 2019 р.

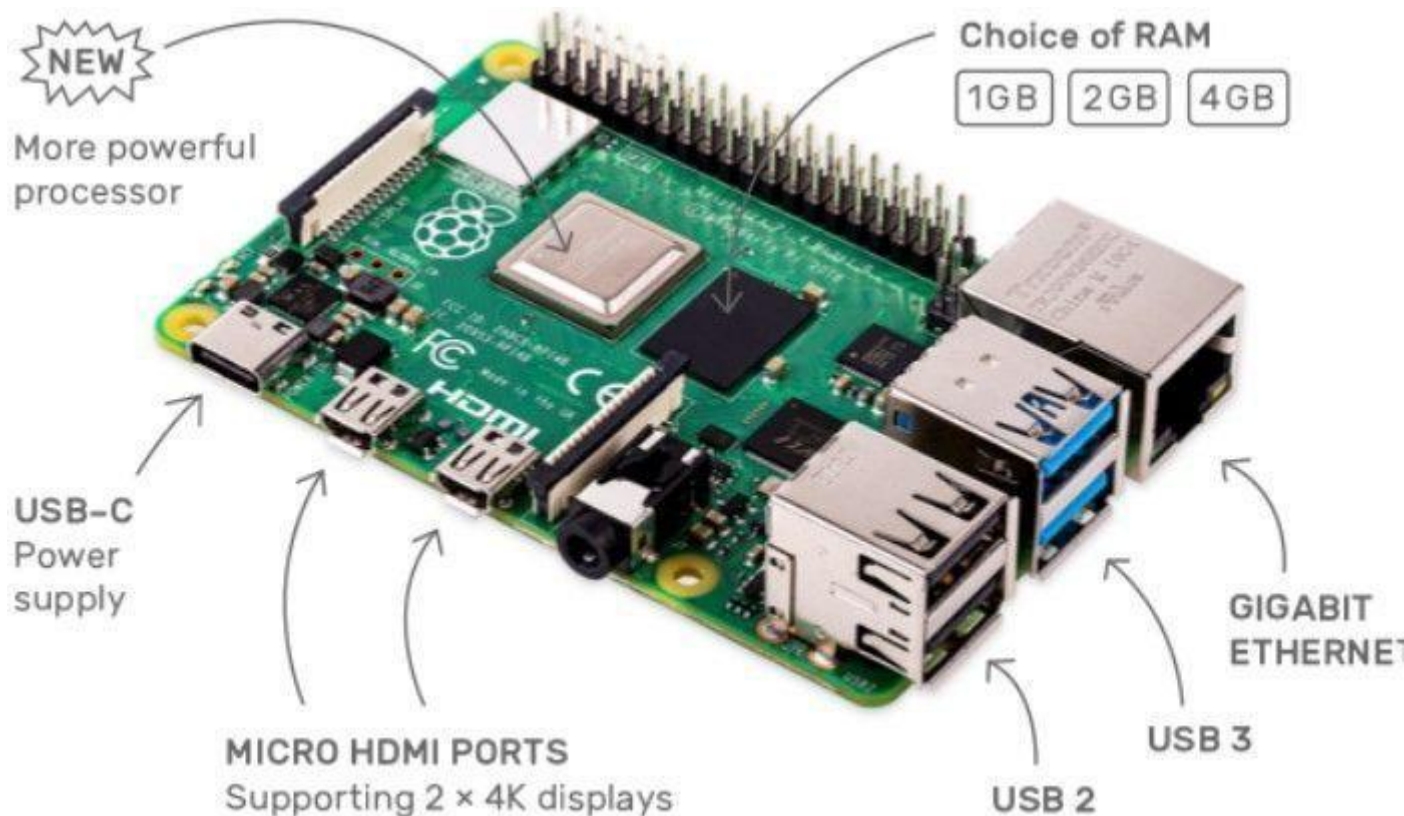


Рисунок 3.2 – Зовнішній вигляд Raspberry Pi 4B

У мережі Інтернет, крім перелічених операційних систем, можна знайти ще безліч модифікацій для різних призначень.

У якості температурного датчика обрано датчик температури DS1621 фірми Dallas, представлений на рисунку 3.3.

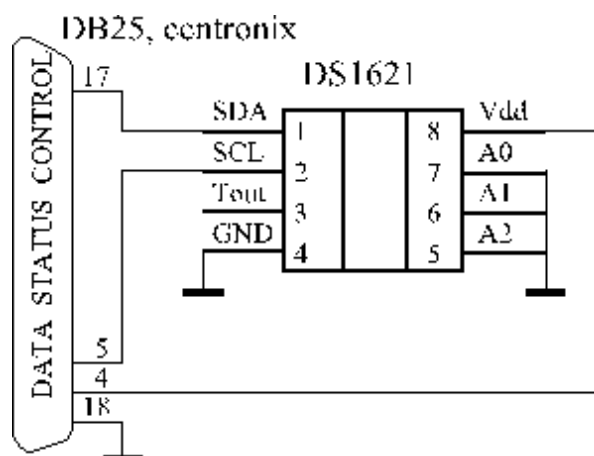


Рисунок 3.3 – Датчик температури DS1621 фірми Dallas

Особливості:

Для вимірювання температури не потрібні зовнішні компоненти;

Вимірює температуру від -55°C до $+125^{\circ}\text{C}$ з кроком $0,5^{\circ}\text{C}$. Еквівалент за Фаренгейтом становить від -67°F до 257°F з кроком $0,9^{\circ}\text{F}$;

Температура зчитується як 9-бітне значення (2-байтова передача);

Широкий діапазон живлення (від 2,7 В до 5,5 В);

Перетворює температуру в цифрове слово менш ніж за 1 секунду;

Налаштування термостата визначається користувачем і є енергонезалежними;

Дані зчитуються/записуються через 2-провідний послідовний інтерфейс (лінії введення/виведення з відкритим стоком);

Застосування включають термостати, промислові системи, споживчі товари, термометри або будь-які термочутливі системи;

8-контактний корпус DIP або SO (150mil і 208mil).

Цифровий термометр і термостат DS1621 забезпечує 9-бітне вимірювання температури, яке вказує на температуру пристрою. Вихід термічної тривоги, TOUT, активний, коли температура пристрою перевищує температуру TH, визначену користувачем. Вихід залишається активним, доки температура не впаде нижче заданої користувачем температури TL, враховуючи будь-який необхідний гістерезис.

Визначені користувачем параметри температури зберігаються в енергонезалежній пам'яті, тому деталі можна запрограмувати перед вставленням у систему. Параметри температури та показники температури передаються до/з DS1621 через простий 2-провідний послідовний інтерфейс.

DS1621 вимірює температуру за допомогою датчика температури на основі забороненої зони. Дельта-сигма аналого-цифровий перетворювач (АЦП) перетворює виміряну температуру в цифрове значення, яке відкаліброване в $^{\circ}\text{C}$; для додатків $^{\circ}\text{F}$ необхідно використовувати таблицю пошуку або програму перетворення.

Зчитування температури надається у 9-бітному вигляді з двома доповненнями шляхом видачі команди READ TEMPERATURE. Дані передаються через 2-провідний послідовний інтерфейс, спочатку MSB. DS1621 може вимірювати температуру в діапазоні від -55°C до $+125^{\circ}\text{C}$ з кроком $0,5^{\circ}\text{C}$.

Оскільки дані спочатку передаються через 2-провідну шину MSB, дані про температуру можуть бути записані в/прочитані з DS1621 як один байт (з роздільною

здатністю температури 1 С) або як два байти. Другий байт міститиме значення молодшого (0,5 С) біта показання температури.

DS1621 можна запрограмувати на виконання безперервних послідовних перетворень (режим безперервного перетворення) або на виконання одиничних перетворень за командою (режим однократного перетворення). Режим перетворення програмується через біт 1SHOT у регістрі конфігурації. У режимі безперервного перетворення DS1621 починає безперервне перетворення після видачі команди Start Convert T. Послідовні перетворення продовжують виконуватися, доки не буде видано команду Stop Convert T, після чого пристрій переходить у стан очікування з низьким енергоспоживанням. Безперервні перетворення можна перезапустити в будь-який час за допомогою команди Start Convert T.

У одноразовому режимі DS1621 виконує одне перетворення температури, коли видається команда Start Convert T. Після завершення перетворення пристрій переходить у стан очікування з низьким енергоспоживанням і залишається в цьому стані, доки знову не буде ініційовано єдине перетворення температури командою Start Convert T.

У своєму робочому режимі DS1621 функціонує як термостат із програмованим гістерезисом. Вихідний сигнал термостата оновлюється, щойно перетворення температури завершується. Коли температура DS1621 досягає або перевищує значення, збережене в регістрі відключення високої температури (TH), вихід стає активним і залишатиметься активним, доки температура не впаде нижче температури, збереженої в регістрі тригера низької температури (TL). Таким чином можна отримати будь-яку величину гістерезису. Активний стан виходу програмується користувачем так, що активним станом може бути або логічна «1» (VDD), або логічний «0» (0 В). Це робиться за допомогою біта POL у регістраторі конфігурації.

У якості датчика вологості було обрано НІН 4000-003 [16]. Датчики вологості НІН-4000 розроблені спеціально для промислових застосувань. Датчики можуть бути безпосередньо підключені до мікроконтролера або іншого пристрою для обробки лінійного сигналу за напругою. Маючи типовий струм споживання 200 мкА, датчики вологості ідеальні для застосування у пристроях з автономним живленням. Повна взаємозамінність датчиків дозволяє значно скоротити витрати на калібрування. Крім того, можливе постачання датчиків з індивідуальним калібруванням.

Датчики вологості НІН-4000 випускаються у плоских корпусах SIP з кроком 1,27 або 2,54 мм. Кристал датчика оброблений лазером і поєднує чутливий ємнісний елемент термореактивного полімеру з мікросхемою посилення сигналу. Багатошарова структура чутливого елемента забезпечує відмінні характеристики приладів навіть у важких умовах

експлуатації: підвищена вологість, забрудненість, присутність масляних речовин та інших хімічних реактивів.

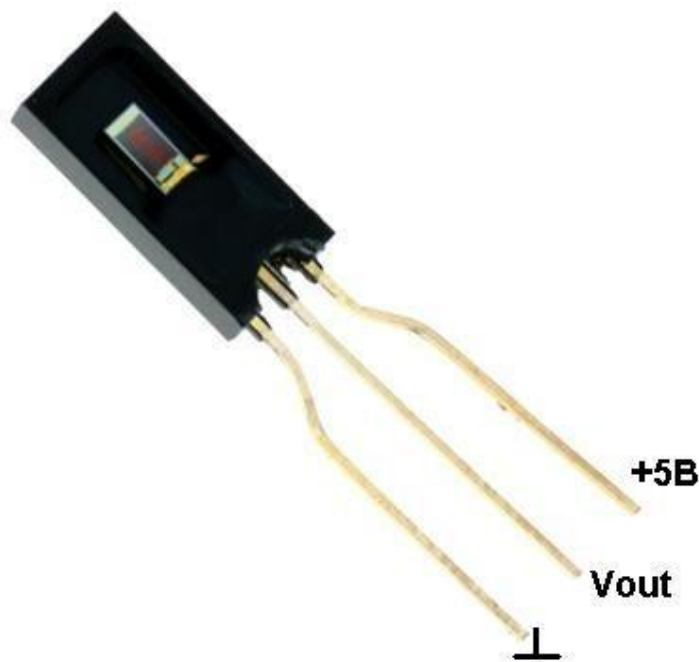


Рисунок 3.4 – Датчик вологості НІН 4000-003

Таблиця 3.1 – Технічні характеристики НІН-4000-003

Параметр	Значення
Виробник	HONEYWELL
Серія	НІН-4000
Діапазон відносної вологості, %	0-100
Вихідний сигнал	напруга
Номінальний вихідний струм, мА	0.05
Час відгуку, с	15
Діапазон вихідної напруги	0.80...3.90
Напруга живлення	5
Корпус SIP	2.54 мм
Вага	0.60 гр.

Датчик вологості ґрунту є важливим компонентом для автоматизації садівництва, оскільки дозволяє вимірювати рівень вологості в ґрунті і автоматично контролювати процеси поливу. Такі датчики широко використовуються в проектах з використанням Arduino та інших мікроконтролерів для моніторингу стану рослин.

Типи датчиків вологості ґрунту

1. Аналогові датчики: Вимірюють опір електричного струму в ґрунті. Чим вищий рівень вологості, тим нижча опірність. Аналогові датчики зазвичай мають простий вихідний сигнал, який можна безпосередньо підключити до аналогового входу Arduino.

2. Цифрові датчики: Відомі як вологостійкі датчики, які зазвичай використовують протоколи, як I2C або 1-Wire для виводу даних. Вони зазвичай є більш стійкими до корозії та мають вищу точність.

Приклад використання аналогового датчика вологості ґрунту з Arduino

Необхідні матеріали

- Arduino (Uno, Mega, Nano тощо)
- Аналоговий датчик вологості ґрунту
- Резистор (звичайно 10 кОм, для підключення до ВК)
- Проводи для з'єднання
- (Опційно) Браслет або щиток для зручності підключення

Схема підключення

1. Підключіть живлення (VCC) датчика до 5V на Arduino.
2. Підключіть землю (GND) датчика до GND на Arduino.
3. Аналігровий вихід (A0) датчика підключіть до аналогового входу A0 на Arduino.

Код для Arduino

```
cpp
// Задаємо змінні для зберігання значень
int sensorPin = A0; // Пін для підключення датчика
int sensorValue = 0; // Змінна для зберігання показників вологості

void setup() {
  Serial.begin(9600); // Ініціалізуємо серійний монітор
}

void loop() {
  sensorValue = analogRead(sensorPin); // Зчитуємо значення з датчика
  // Виводимо значення на серійний монітор
  Serial.print("Вологість ґрунту: ");
  Serial.println(sensorValue);
  delay(1000); // Затримка в 1 секунду перед наступним зчитуванням
}
```

Пояснення коду

- `analogRead(sensorPin)`: Функція зчитує аналоговий сигнал з пін, де підключений датчик. Значення буде в діапазоні від 0 до 1023.
- Значення змінної `sensorValue` відображатиме рівень вологості: чим нижче число, тим вологіший ґрунт, і навпаки.
- Серійний монітор відображає значення вологості кожну секунду.

Використання датчика вологості ґрунту в поєднанні з Arduino дозволяє створити автоматизовані системи поливу, що функціонують на основі реальних даних про вологість ґрунту. Використання таких систем може зменшити витрати води і покращити здоров'я рослин. За потреби, можна доповнити проект сервоприводом або електромагнітним клапаном для автоматичного управління поливом на основі показників вологості.

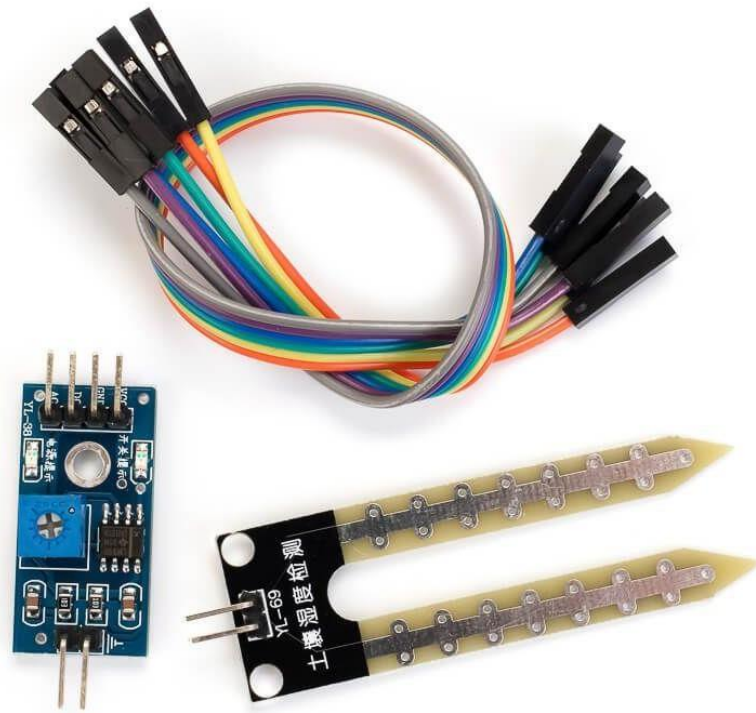


Рисунок 3.5 – Датчик вологості ґрунту

Датчик світла працює у великому діапазоні освітлення і може використовуватися без Arduino, його можна зв'язати цифровим виходом безпосередньо з реле або іншим пристроєм.

Характеристики:

Фоторезистор: GL5516 (MLG5516B)

Напруга живлення: 3.3V - 5V

Вихід датчика: цифровий бінарний ("1" або "0")

Пік спектральної чутливості: 560nm (видиме світло)

Діапазон робочих температур: -30 ° C ... +70 ° C

Розмір плати: 32 x 14 x 7 мм

На платі модуля знаходяться фоторезистор, здвоєний компаратор LM393, підстроювальний резистор, світлодіод живлення і світлодіод спрацьовування датчика.

Піни:

VCC, GND – живлення модуля

DO – цифровий бінарний вихід

3.3 Верифікація результатів дослідження

У сучасних обчислювальних системах команди машинного коду або асемблера класифікуються на кілька груп відповідно до їх функцій. Кожна група команд реалізує певні операції та завдання. Нижче наведено опис кожної з груп команд.

1. Команди логічних операцій

Ця група команд використовується для виконання логічних операцій над бітами даних. До них належать:

AND: Логічна операція "І", що виконує бітове "І" між двома операндами.

OR: Логічна операція "АБО", що виконує бітове "АБО" між двома операндами.

XOR: Логічна операція "Виключне АБО", що виконує бітове "Виключне АБО".

NOT: Логічне заперечення, яке інвертує біти даного операнда.

2. Команди арифметичних операцій і команди зсуву

Ця група містить команди, які виконують арифметичні обчислення, а також операції зсуву:

ADD: Додавання двох операндів.

SUB: Віднімання одного операнда від іншого.

MUL: Множення двох операндів.

DIV: Ділення одного операнда на інший.

INC: Збільшення значення операнда на одиницю.

DEC: Зменшення значення операнда на одиницю.

SHL (Shift Left): Зсув бітів вліво, що множить число на 2.

SHR (Shift Right): Зсув бітів вправо, що ділить число на 2 (для беззнакових чисел).

3. Команди операцій з бітами

Ця група охоплює команди для роботи з окремими бітами даних:

BIT SET: Встановлює певний біт в 1.

BIT CLEAR: Скидає певний біт у 0.

BIT TOGGLE: Інвертує значення певного біта.

4. Команди пересилання даних

Ці команди відповідають за переміщення даних між регістрами, пам'яттю, а також пристроями вводу/виводу:

MOV: Переміщення даних з одного місця в інше.

PUSH: Поміщає дані в стос (stack).

POP: Витягує дані з стосу.

LOAD: Завантажує дані з пам'яті в регістр.

STORE: Зберігає дані з регістра в пам'ять.

5. Команди передачі управління

Ця група команд визначає напрямок виконання програми:

JMP: Непряма передача управління на зазначену адресу.

CALL: Виклик підпрограми, передача управління до специфікованого місця і збереження поточної адреси повернення.

RET: Повернення з підпрограми, повертаючи управління на адресу, що зберігається у стосі.

Conditional Jumps (наприклад, JZ, JNZ): Умовні переходи, які виконуються якщо певна умова виконується (наприклад, "перейти, якщо нуль").

6. Команди управління системою

Ця група включає команди, які управляють системою загалом та її ресурсами:

NOP: "Нічого не робити", команда, яка нічого не виконує — використовується для заповнення місця в коді.

HLT: Зупиняє виконання програми.

INT: Генерує переривання, що дозволяє контролювати обробку подій (наприклад, підключення пристроїв).

CLI: Вимикає обробку переривань (clear interrupts).

STI: Включає обробку переривань (set interrupts).

У таблиці 3.2 представлені коди для відображення цифр і знака «-»:

Таблиця 3.2 – Коди для відображення цифр і знака «-»

	d0	d1	d2	d3	d4	d5	d6	код
цифра 1	0	1	1	0	0	0	0	0x82
цифра 2	1	1	0	1	1	0	1	0x3e
цифра 3	1	1	1	1	0	0	1	0xae
цифра 4	0	1	1	0	0	1	1	0x87
цифра 5	1	0	1	1	0	1	1	0xad
цифра 6	1	0	1	1	1	1	1	0xbd
цифра 7	1	1	1	0	0	0	0	0x22
цифра 8	1	1	1	1	1	1	1	0xbf
цифра 9	1	1	1	1	0	1	1	0xaf
цифра 0	1	1	1	1	1	1	0	0xbb
знак	0	0	0	0	0	0	1	0x1
	A	B	C	D	E	F	G	

Тестування IoT-системи для автоматизованого догляду за садом необхідно проводити на реальному об'єкті, обов'язковою умовою є встановлення та підключення датчиків описаних у роботі.

Налаштування здійснюється за допомогою коду.

Висновки до розділу

У рамках третього розділу розроблено структурну схему IoT-системи для автоматизованого догляду за садом, здійснено вибір обладнання та програмного забезпечення та наведено верифікацію результатів дослідження.

ВИСНОВКИ

У рамках даної кваліфікаційної роботи здійснено розробку системи на базі Інтернету речей для автоматизації садівництва. На основі вищевикладеного варто зробити наступний висновок:

Сучасні технології розумного садівництва включають різні продукти, які можуть допомогти садівникам автоматизувати численні завдання, такі як полив, удобрення та боротьба зі шкідниками. Ці продукти стають все більш доступними та простими у використанні, що робить їх доступними для більш широкого кола споживачів. Оскільки ринок продуктів для розумного садівництва продовжує зростати, є можливість очікувати ще більше інноваційних рішень, які спростують, ніж будь-коли, вирощування здорових рослин. Технології Smart Garden мають великий потенціал, але деякі проблеми ще потребують вирішення. Економічна ефективність, стандартизація, простота, точність і відмовостійкість – все це питання, які зараз вивчаються.

Розроблена IoT-система для автоматизованого догляду за садом дозволяє встановлювати критичні параметри відповідно до обраного режиму, регулює температуру і вологість, враховуючи особливості реалізації.

Тестування IoT-системи для автоматизованого догляду за садом необхідно проводити на реальному об'єкті, обов'язковою умовою є встановлення та підключення датчиків описаних у роботі.

Налаштування здійснюється за допомогою коду.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. T.Thamaraimanalan, S.P.Vivekk, G.Satheeshkumar, P.Saravanan, «Smart Garden Monitoring System Using IOT» Asian Journal of Applied Science and Technology (AJAST) (Open Access Quarterly International Journal) Volume 2, Issue 2, Pages 186-192, April-June 2018
2. S.N. Ishak, N.N.N.Abd Malik, N.M. Abdul Latiff, N. Effiyana Ghazali, M. A. Baharudin, «Smart Home Garden Irrigation System Using Raspberry Pi», Faculty of Electrical Engineering, Universiti Teknologi Malaysia, 81310 Johor Bahru, Johor, Malaysia, 2017.
3. All about circuits. URL: <https://www.allaboutcircuits.com/technical-articles/understanding-arduino-uno-hardware-design/> (дата звернення 20.10.2024)
4. Atmega328P – Microchip Technology. URL: http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf
5. The Rheingold Heavy. URL: <https://rheingoldheavy.com/build-an-arduino-uno-from-scratch-part-1/> (дата обращения 03.10.2024)
6. Borade, Ms. (2024). Blooms & Bytes Nurturing Garden with Smart Irrigation – Planning. INTERANTIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING AND MANAGEMENT. 08. 1-5. 10.55041/IJSREM31164.
7. Geng, Xiuting & Chun, Ou. (2022). Multiscale Layout and Planning of Smart Gardens in the Environment of Agricultural Internet of Things. Computational Intelligence and Neuroscience. 2022. 1-12. 10.1155/2022/4746629.
8. Yang, Yanan. (2024). Smart Garden Management System Based on the combination of Internet of Things and Geographic Information System Technology. International Journal of Communication Networks and Information Security (IJCNIS). 16. 10.17762/ijcnis.v16i1.6471.
9. Zia, Hina & Khan, Nisar & Khatri, Anand & Dogne, Nitesh & Ibraheem, Anam. (2023). Urban Mobility - Wayfinding Signage and Vertical Gardens in Jammu Smart City.
10. Kanhe, Ms & Kharde, Ms & Kankariya, Mr. (2021). Garden Watering System. International Journal of Advanced Research in Science, Communication and Technology. 129-134. 10.48175/IJARSCT-1180.
11. Alfaatihah Alvaran, Muhammad & Permanasari, Maharani & Sudrajat, Alia & Kurniatillah, Aniq & Shavira, Mirza & Afiff, Diza. (2022). Modular Vertical Garden Sebagai Solusi Praktis Urban Gardening Institut Teknologi Nasional. Jurnal Rekayasa Hijau. 5. 207-217. 10.26760/jrh.v5i3.207-217.
12. Han, Gwangho & Ro, Seunghan. (2022). A Study on the Sustainable Urban Redevelopment Structure Based on the Garden City: Focused on the Battery Park City and Hudson

Yards Project in U.S.. Residential Environment Institute Of Korea. 20. 85-98. 10.22313/reik.2022.20.2.85.

13. Elangovan, Ramkumar & Santhanakrishnan, Nagarani & Rozario, Roger & Banu, Arjuman. (2018). Tomen:A Plant monitoring and smart gardening system using IoT. International Journal of Pure and Applied Mathematics. Volume 119.

14. Osita, Kenneth. (2022). IOT BASED SMART GREENHOUSE: Using packet tracer Software.

15. Penzenstadler, Birgit & Khakurel, Jayden & Plojo, Carl & Sanchez, Marinela & Marin, Ruben & Tran, Lam. (2018). Resilient Smart Gardens—Exploration of a Blueprint. Sustainability. 10. 2654. 10.3390/su10082654.

16. Sinatti, Eugenia & Weckert, Simon & Dobrzalski, Ewelina. (2022). Image Garden: Curating Collections and Designing Smart Exhibitions with AI-Based Tools. i-com. 21. 83-94. 10.1515/icom-2022-0010.

17. Ratnayake, Sujith. (2020). Integration of Climate-Smart City Features for Urban Planning and Development: Case Study from Kurunegala Climate-Smart City of Sri Lanka. 10.13140/RG.2.2.26635.44321.

18. Mallick, Biswajit. (2024). Development of a Smart Garden. 7. 8-22. 10.5281/zenodo.10478352.

19. Robert, Osei-Kyei & Xiaohua, Jin & Akomea-Frimpong, Isaac. (2024). Smart Construction in Infrastructure Project Development. 10.13140/RG.2.2.22772.97929.

20. Lu, Mingzhen. (2024). Intelligent design and realization of sustainable development-oriented garden. Journal of Intelligent & Fuzzy Systems. 1-14. 10.3233/JIFS-234540.

```

<?php
    include "../model/conn.php";
    $db = new DataBase();
    $soilLow = 0;
    $soilUp = 0;
    include "../php/checkCookie.php";
    if(isset($_GET['id'])){
        $id = $_GET['id'];
        $checkID = "SELECT *
        FROM microbits
        WHERE microbit_id = '$id'";
        $num = $db->num( $checkID);
        if($num == 0){
            header("Location: dashboard.php");
        }else{
            $sqlcheck = $db->send( $checkID);
            $row = $sqlcheck->fetch_assoc();
            if($row['microbit_owner'] != $_COOKIE['user-id']){
                header("Location: dashboard.php");
            }
        }
        //////////////////////////////////////
        $adaUserName = $row['ada_username'];
        $m_name = $row['microbit_name'];

    }
    }else{
        header("Location: dashboard.php");
    }
}
?>
<!DOCTYPE html>
<html lang="en">

<head>
<meta charset="UTF-8" />
<meta http-equiv="X-UA-Compatible" content="IE=edge" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC"
crossorigin="anonymous" />
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"
integrity="sha384-
MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"
crossorigin="anonymous"></script>
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-
icons@1.7.2/font/bootstrap-icons.css" />
<link rel="preconnect" href="https://fonts.gstatic.com" />
<link href="https://fonts.googleapis.com/css2?family=Roboto+Condensed:wght@300;400;700&display=
swap" rel="stylesheet" />
<link rel="stylesheet" href="../asset/css/main.css" />
<link rel="shortcut icon" href="../favicon.png" type="image/x-icon" />

```

```

<script src="../asset/js/jquery.min.js"></script>
<title>SmartGaden</title>
</head>

<body>
<input type="hidden" id="hide-value-user-id" value="<?php echo $_COOKIE['user-
id'];?>">
<input type="hidden" id="hide-value" value="<?php echo $_GET['id'];?>">
<input type="hidden" id="hide-value-adaName" value="<?php echo $adaUserName;?>">

<div id="wr-header">
<div id="header-1" class="container justify-content-between d-flex">
<div>
<a href="dashboard.php"></a>
</div>

<div class="mobile-look">
<strong>SMART GARDEN PROJECT - TEAM A</strong>
</div>

<div class="d-flex">
<div class="dropdown">
<button class="btn btn-secondary dropdown-toggle" type="button"
id="dropdownMenuButton1" data-bs-toggle="dropdown" aria-expanded="false">
    Обліковий запис
</button>
<ul class="dropdown-menu" aria-labelledby="dropdownMenuButton1">
<li><a class="dropdown-item" href="info.php">Інформація</a></li>
<li><a class="dropdown-item" href="dashboard.php">Приладова панель</a></li>
<li><a class="dropdown-item" href="login.php?logout=true">Вийти</a></li>
</ul>
</div>
</div>
</div>
</div>
<div id="wr-body" class="container">
<a href="info.php" class="link-history"><?php echo $_COOKIE["user-name"]; ?></a>
<span class="link-history">></span>
<a href="dashboard.php" class="link-history">Приладова панель</a>
<span class="link-history">></span>
<a href="detail.php?id=<?php echo $id?>" class="link-history">Інформаційна
сторінка</a>
<span class="link-history">> Історія</span>
<h2>Графік вологості ґрунту протягом доби</h2>
<div class="canvas" id="canvas">
<canvas id="ctx" style="width: 100%; height: 100%;"></canvas>
</div>
<div style="display: flex;margin-top:20px">
<div style="width: 20px;height:20px;background:blue"></div>
<span>: Вологість ґрунту.</span>
</div>
<div style="display: flex;margin-top:20px">
<div style="width: 20px;height:20px;background:red"></div>

```

```

<span><?php
    if(!isset($_GET['y']) || !isset($_GET['m']) || !isset($_GET['d']))($_GET['y'] == "" ||
$_GET['m'] == "" || $_GET['d'] == "")){
        $y = date("Y");
        $m = date("m");
        $d = date("d");
        header("Location: history.php?id=$id&y=$y&m=$m&d=$d");
    }
    if(isset($_GET['y'])){
        $y = $_GET['y'];
        $year = "AND year = '$y'";
    }
    if(isset($_GET['m'])){
        $m = $_GET['m'];
        $month = "AND month = '$m'";
    }
    if(isset($_GET['d'])){
        $d = $_GET['d'];
        $day = "AND day = '$d'";
    }
    ?>
<input type="hidden" id="input-year" value="<?php echo $y?>">
<input type="hidden" id="input-month" value="<?php echo $m?>">
<input type="hidden" id="input-day" value="<?php echo $d?>">
<div style="margin-top: 20px;border-radius:20px;border:1px solid black;padding:20px
10px;">
<h2>Добова історія вологості ґрунту</h2>
<span><?php echo $d.' / '.$m.' / '.$y;?></span>

<div class="d-flex">
<select id="select-day" class="form-select" aria-label="Default select example">
<?php
    for ($i=1; $i <= 31; $i++) {
        if(isset($_GET['d']) && $i == $_GET['d']) {
            if($i < 10){
                ?><option value="<?php echo '0'.$i; ?>" selected><?php echo '0'.$i;
?></option><?php
            }else{
                ?><option value="<?php echo $i; ?>" selected><?php echo $i;
?></option><?php
            }
        }else {
            if($i < 10){
                ?><option value="<?php echo '0'.$i; ?>"><?php echo '0'.$i;
?></option><?php
            }else{
                ?><option value="<?php echo $i; ?>"><?php echo $i;
?></option><?php
            }
        }
    }
?>

```

```

    }
  }
?>
</select>
<select id="select-month" class="form-select" aria-label="Default select example">
<?php
    for ($i=1; $i <= 12; $i++) {
        if(isset($_GET['m']) && $i == $_GET['m']) {
            if($i < 10){
                ?><option value="<?php echo '0'.$i; ?>" selected><?php echo '0'.$i;
?></option><?php
                }else{
                    ?><option value="<?php echo $i; ?>" selected><?php echo $i;
?></option><?php
                }
            }else {
                if($i < 10){
                    ?><option value="<?php echo '0'.$i; ?>"><?php echo '0'.$i;
?></option><?php
                    }else{
                        ?><option value="<?php echo $i; ?>"><?php echo $i;
?></option><?php
                    }
                }
            }
        }
    }
?>
</select>
<select class="form-select" aria-label="Default select example">
<option value="2024" selected>2024</option>
</select>
<button onclick="find()" type="button" class="btn btn-primary">Фільтр</button>
</div>
<script>
    find()=>{
        let d = $("#select-day").val();

        let m = $("#select-month").val();
        location.href= "history.php?id=<?php
$хid?>&d="+d+"&m="+m+"&y=2022";
        }
    }
</script>
<table class="table table-hover">
<thead>
<tr>
<th scope="col">#</th>
<th scope="col">Температура</th>
<th scope="col">Вологість</th>
<th scope="col">Час</th>
</tr>
</thead>
<tbody>
<?php
    $query_1 = "SELECT *

```

```

FROM general_infor
WHERE microbit_id = '$id' $year$month$day";
if($db->num($query_1) != 0 ){
    $sql_1 = $db->send($query_1);
    $i=1;
    while($row_1 = $sql_1->fetch_array()){
        ?><tr>
<th scope="row"><?php echo $i ?></th>
<td><?php echo $row_1['temperature']; ?></td>
<td><?php echo $row_1['humidity']; ?></td>
<td><?php echo $row_1['hour'].': '.$row_1['min']; ?></td>
</tr><?php
        $i++;
    }
    }else echo "<tr>На цю дату немає даних.</tr>";
?>

</tbody>
</table>
</div>
</div>
<script src="../../asset/js/canvas.js"></script>
<script src="../../asset/js/app.js"></script>
</body>

</html>

```

Демонстраційні матеріали (Презентація)

АКТУАЛІСТЬ ДОСІДЖЕННЯ

Садівництво є важливою частиною української культури, проте традиційні методи потребують значних зусиль.

Завдяки автоматизації та інноваційним технологіям очікується стрімке зростання глобального ринку Smart Gardening.

Об'єкт, предмет, методи

- **Об'єкт вивчення** — жорсткий диск автоматизації садівництва.
- **Предмет вивчення** — розробка схем підключення систем і програмної моделі керування системою автоматизації садівництва.
- **Методи вивчення.** Під час виконання роботи використовувались теоретичні методи досліджень із застосуванням апарату термодинаміки, електроніки, теорії ймовірності, математичної статистики, тощо.

1

МЕТА ДОСІДЖЕННЯ

Метою даного дослідження є розробити жорсткий диск використання Інтернету речей для автоматизації садівництва.

Завдання

1. Навести методи та моделі автоматизації садівництва;
2. розробити алгоритм реалізації процесу автоматизації садівництва;
3. Запропонувати структурну схему IoT-системи для автоматизованого догляду за садом;
4. Здійснити вибір обладнання та програмного забезпечення;
5. представити верифікацію результатів дослідження.

2

ФУНКЦІЇ ІОТ-СИСТЕМИ ДЛЯ АВТОМАТИЗОВАНОГО ДОГЛЯДУ ЗА САДОМ

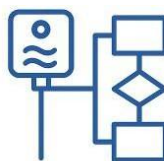
1. Початковий запуск системи
2. Вибір необхідного для підтримки типу мікроклімату.
3. Прийом даних з датчиків і обробка цих даних відповідно до алгоритму.



Початковий
запуск системи



Вибір необхідного
для підтримки
типу мікроклімату

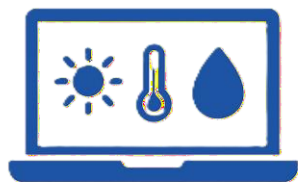


Прийом даних з
датчиків і обробка
цих даних відповідно
до алгоритму

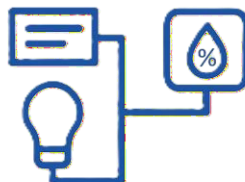
3

ФУНКЦІЇ ІОТ-СИСТЕМИ ДЛЯ АВТОМАТИЗОВАНОГО ДОГЛЯДУ ЗА САДОМ

4. Виведення поточних параметрів мікроклімату середовища.
5. Формування вихідних сигналів для запуску виконавчих пристроїв освітленості, нагріву, зволоження.



Виведення поточних
параметрів мікроклімату
середовища

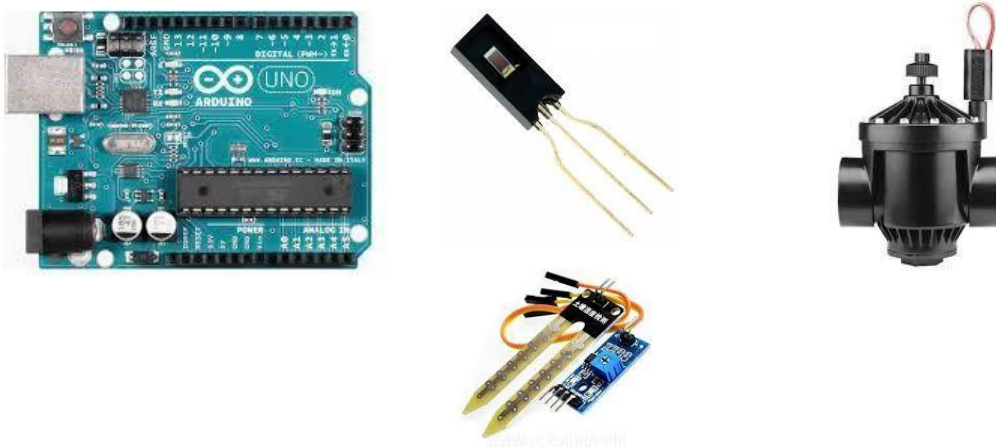


Формування вихідних сигналів
для запуску виконавчих
пристроїв освітленості, нагріву,
зволоження

4

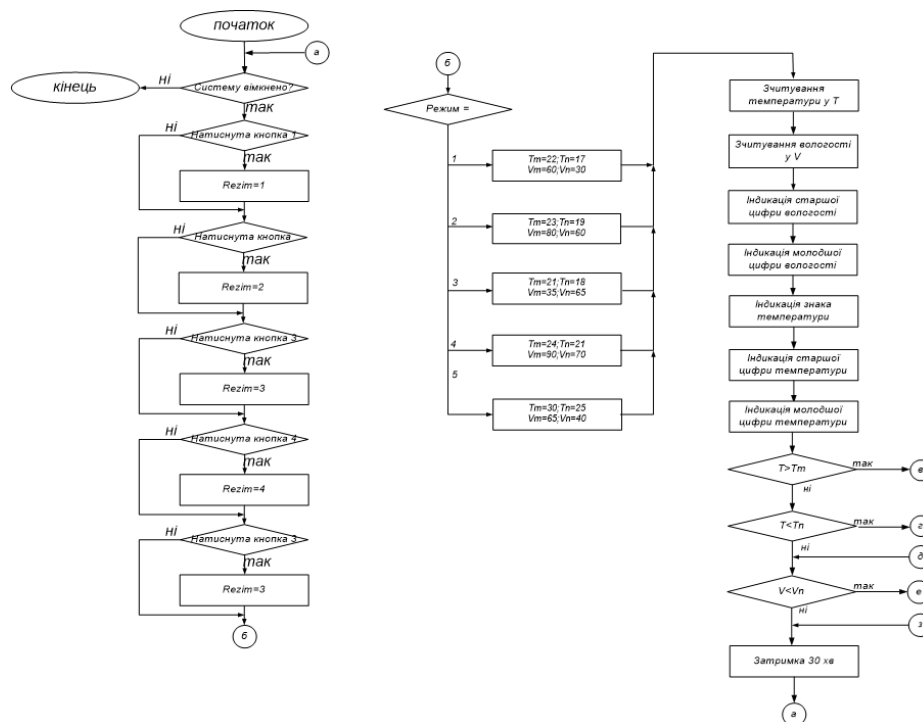
Відповідно до вимог технічного завдання, система поділяється на основні модулі:

- **Мікроконтролер** – головний керуючий елемент
- **Сенсори** – збір даних з навколишнього середовища
- **Виконавчі механізми** – реалізація дій за командою системи



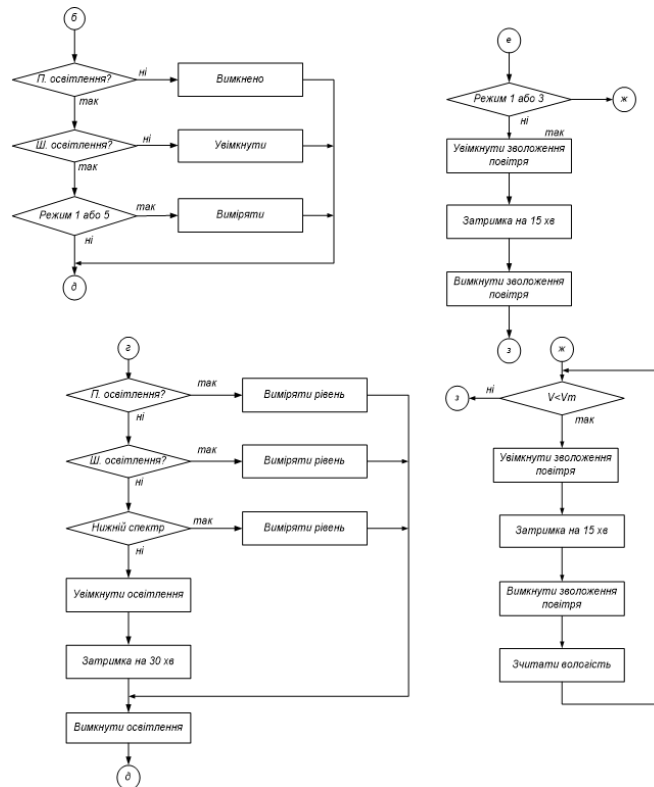
5

АЛГОРИТМ РЕАЛІЗАЦІЇ ПРО ЕСУ АВТОМАТИЗАЦІЇ САДІВНИ ТВА



6

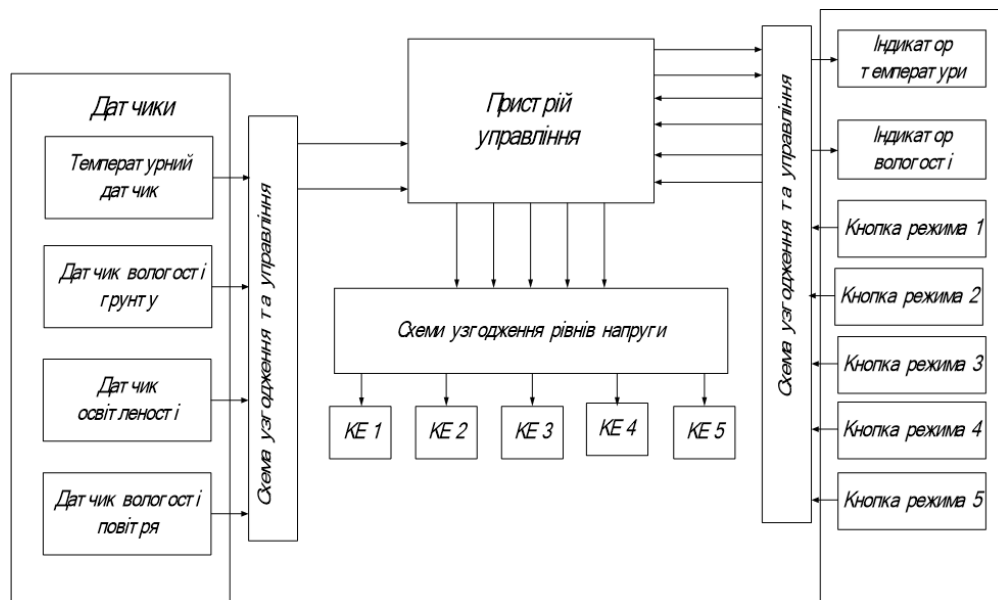
- Алгоритм роботи Smart Garden



7

Алгоритм роботи Smart Garden

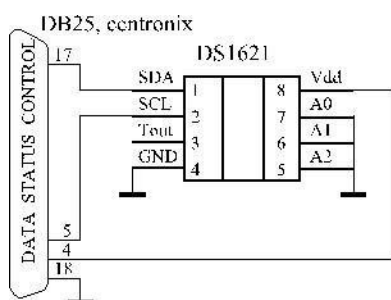
РОЗРОБКА ІОТ-СИСТЕМ ДЛЯ АВТОМАТИЗОВАНОГО ДОГЛЯДУ ЗА САДОМ



8

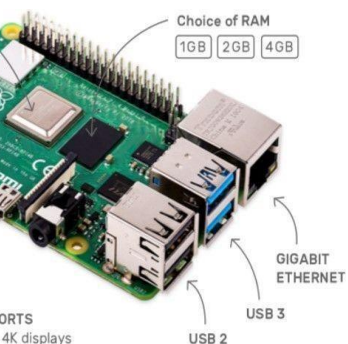
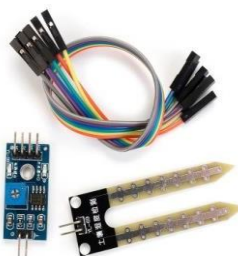
Структурна схема ІоТ-системи для автоматизованого догляду за садом

ОБЛАДНАННЯ

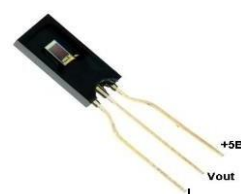


Датчике температури DS1621 фірми Dallas

Датчике воюгості фрунту



Плата Raspberry Pi



Датчике воюгості
HIN 4000-003

9

ВНСНОВЕН

У даній еваніфікаційній роботі буио:

1. РоЗроблено IoT-систему дие автоматизованого догпеду За садом.
2. Сучасні техноюгії Smart Gardening автоматизують жоиив, удобренне та боротьбу Зи жеїдниеами.
3. Продуети стають достужніжими та Зручніжими дие еористувачів.
4. Риное роЗумного садівництва аетивно Зростас, очіеується жоева нових інноваційних ріжень.
5. Основні виеиiei: ееономічна ефеетивність, стандартиЗаціє, жростота, точність і надійність систем.
6. IoT-система доЗвоиес регуиювати температуру та воюгість Заиехно від рехиму роботи.
7. Тестуванне системи мас жроводитисе на реаньному об'сеті З жідеиюченням віджовідних датчииєів.
8. Наиаяжтуванне системи виеонується За дожомогою еоду.

10

АПРОБА ЦІЯ

Сучасний стан та перспективи розвитку ІОТ 15.04 2025

Тезис 1

"Інтелектуальна система моніторингу та управління мікрокліматом у теплицях"

Тезис 2

"Розробка адаптивної системи автоматизованого догляду за кімнатними рослинами"