

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЧЕРКАСЬКИЙ ДЕРЖАВНИЙ БІЗНЕС-КОЛЕДЖ
КАФЕДРА КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

на тему

Система дистанційного моніторингу кліматичних показників середовища

Виконав: студент груп KI-22

спеціальності

123 «Комп'ютерна інженерія»

Краснощокій Д.М.

Науковий керівник канд. техн. наук,
викладач Розломій І.О.

Рецензент _____

(науковий ступінь, вчене звання, прізвище та ініціали)

Черкаси, 2024

ЗМІСТ

ВСТУП		4
РОЗДІЛ 1. МОНІТОРИНГОВІ СИСТЕМИ: ПОТРЕБА ТА ПЕРСПЕКТИВИ		5
1.1	Основні характеристики та складові систем моніторингу	5
1.2	Важливість моніторингу для сталого розвитку	6
1.3	Аналіз сучасних вимог до моніторингових систем	7
1.4	Висновки до розділу	8
РОЗДІЛ 2. ОГЛЯД ТА ПОРІВНЯННЯ ПРИСТРОЇВ МОНІТОРИНГУ КЛІМАТИЧНИХ ПОКАЗНИКІВ СЕРЕДОВИЩА		9
2.1	Огляд існуючих аналогів пристроїв моніторингу кліматичних показників середовища	9
2.1.1	Опис існуючих систем	9
2.1.2	Nest Learning Thermostat	9
2.1.3	Ecobee Smart Thermostat	11
2.1.4	Netatmo Weather Station	12
2.1.5	Airthings Wave Plus	14
2.1.6	Foobot	15
2.2	Порівняння оглянутих існуючих аналогів	16
2.3	Обґрунтування вибору апаратних та програмних засобів для побудови системи	18
2.4	Вибір апаратних засобів реалізації системи	18
2.5	Вибір інструментів розробки програмної частини системи	20
2.6	Постановка завдання	20
2.7	Висновки до розділу	21
РОЗДІЛ 3. ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ		22
3.1	Опис і моделювання системи дистанційного моніторингу кліматичних показників середовища	22
3.2	Розробка структурної схеми системи дистанційного моніторингу кліматичних показників середовища	23
3.3	Розробка функціональної схеми системи дистанційного моніторингу кліматичних показників середовища	24
3.4	Розробка діаграми процесів схеми системи дистанційного моніторингу кліматичних показників середовища, принципова схема апаратних частин системи	25
3.5	Вибір хмарної платформи для відправлення даних на сервер	29
3.6	Висновки до розділу	31
РОЗДІЛ 4. ПРАКТИЧНА РЕАЛІЗАЦІЯ РОБОТИ		32
4.1	Реалізація апаратної частини системи	32
4.2	Реалізація програмної частини системи	33
4.3	Підключення системи до сервісу ThingSpeak	38
4.4	Висновки до розділу	43
РОЗДІЛ 5. РОЗРОБКА 3D МОДЕЛІ КОРПУСУ В СЕРЕДОВИЩІ SOLIDWORKS		44

5.1	Важливість Розробки Корпусу для Системи	44
5.2	Вимоги до розробки корпусу	45
5.3	Етапи проектування 3D моделі	47
5.3.1	Загальне проектування корпусу	47
5.3.2	Проекції та деталі корпусу та нижньої кришки	49
5.4	Підготовка до 3D друку корпусу	52
5.5	Висновки до розділу	57
РОЗДІЛ 6.	ВПРОВАДЖЕННЯ СИСТЕМИ ДИСТАНЦІЙНОГО МОНІТОРИНГУ МІКРОКЛІМАТУ В ЕКСПЛУАТАЦІЮ ТА ПЛАН ПО НАЛАГОДЖЕННЮ РОБОТИ	58
6.1	Інструкція з використання пристрою вимірювання	58
6.2	Інструкція по налагодженню роботи і виправленню помилок	59
6.3	Можливості покращення системи	62
6.4	Висновки до розділу	63
ВИСНОВКИ		64
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ		65
ДОДАТОК А		67
ДОДАТОК Б		68

ВСТУП

Збирання даних про кліматичні показники - це процес отримання інформації про стан клімату за допомогою спеціальних пристроїв. Отримані дані аналізуються для створення моделі кліматичної поведінки. Одна з проблем з вимірюванням кліматичних показників полягає в правильному виборі датчиків, що залежить від конкретної області застосування та відповідності вимогам вимірювання. Важливою є також обробка та передача отриманої інформації людині або іншим пристроям у системі.

Актуальність цієї теми очевидна у різних сферах людської діяльності, де потрібно вимірювати кліматичні показники. Це охоплює автоматичний контроль температури в системах розумного будинку, регулювання процесів виробництва та прогнозування погоди на майбутнє. Тому пристрої та системи для вимірювання кліматичних показників різної складності завжди будуть в попиті.

Метою даної роботи є проектування та розробка пристрою для моніторингу кліматичних показників середовища, який передає дані на віддалений сервер. З цієї мети впливають основні завдання розробки, що включають аналіз області вимірювання показників клімату, дослідження наявних апаратних та програмних рішень для вимірювання клімату, визначення функціональних вимог до апаратної частини пристрою, вибір апаратних компонентів, реалізація системи з використанням обраних рішень та тестування готової системи.

Об'єктом розробки є процес збору кліматичних показників середовища за допомогою системи датчиків, а предметом - система моніторингу кліматичних показників середовища, яка в реальному часі передає зібрані дані до серверу для подальшої обробки і аналізу.

Практична цінність проекту полягає в можливості користувача в реальному часі контролювати стан кліматичних показників середовища і отримувати статистику зміни клімату.

РОЗДІЛ 1. МОНІТОРИНГОВІ СИСТЕМИ: ПОТРЕБА ТА ПЕРСПЕКТИВИ

1.1 Основні характеристики та складові систем моніторингу

Системи моніторингу є складними технічними структурами, спроектованими для систематичного збору, аналізу та інтерпретації даних з метою спостереження за певними об'єктами або явищами[1]. Вони мають ряд характеристик та складових, які визначають їхню ефективність та функціональність[2]:

1. Апаратне забезпечення:

Датчики та сенсори: Системи моніторингу включають в себе різноманітні датчики та сенсори, які призначені для збору інформації про оточуюче середовище[3]. Ці пристрої можуть вимірювати температуру, вологість, рівень забруднення повітря, рівень води та багато інших параметрів.

Монітори та прилади збору даних: Крім датчиків, системи моніторингу можуть включати в себе монітори та прилади збору даних, які призначені для реєстрації та збереження отриманих вимірювань.

2. Програмне забезпечення.

Системи збору та обробки даних. Для ефективного управління та аналізу великих обсягів даних використовуються спеціальні програмні засоби, які дозволяють збирати, обробляти та аналізувати інформацію з датчиків та сенсорів. Візуалізація результатів. Програмне забезпечення також може включати засоби візуалізації даних, які дозволяють представляти результати моніторингу у зрозумілій та зручній формі для аналізу.

3. Методи аналізу та інтерпретації.

Статистичний аналіз. Для виявлення тенденцій, аномалій та взаємозв'язків у даних застосовуються різноманітні методи статистичного аналізу. Моделювання. Деякі системи моніторингу використовують моделі для прогнозування майбутніх змін та оцінки впливу різних факторів на досліджувані процеси.

4. Різноманітність застосувань. Локальні системи моніторингу: Деякі системи моніторингу призначені для локального використання, наприклад, для

контролю за рівнем забруднення повітря в міських районах або для моніторингу кліматичних умов у певній галузі.

Глобальні мережі моніторингу: Інші системи можуть бути частиною глобальних мереж моніторингу, які охоплюють великі території або навіть весь планету, наприклад, системи моніторингу змін клімату або відслідковування руху забруднень у водах світових океанів.

1.2 Важливість моніторингу для сталого розвитку

Моніторинг є не тільки критичним елементом, але й невід'ємною складовою для досягнення сталого розвитку суспільства та забезпечення екологічної стійкості[4]. Він виконує ряд важливих функцій, які сприяють збереженню природних ресурсів та мінімізації негативного впливу на довкілля:

1. Моніторинг та відстеження змін у кліматі та довкіллі. Системи моніторингу дозволяють постійно відстежувати та аналізувати зміни в кліматі, екосистемах та природних ресурсах. Це дозволяє належним чином реагувати на екологічні та кліматичні виклики, такі як глобальне потепління, зниження біорізноманіття та інші[5].

2. Вплив людської діяльності на екосистеми. Моніторинг дозволяє виявляти та оцінювати вплив різноманітних антропогенних діяльностей на природні екосистеми. Це включає в себе аналіз викидів та забруднень, втручання у водні та земельні ресурси, зміни використання землі та інші аспекти.

3. Оцінка ефективності заходів з охорони довкілля. Через моніторинг можна визначати ефективність заходів та стратегій з охорони навколишнього середовища. Це допомагає виявляти успішні практики та удосконалювати підходи для досягнення екологічних цілей.

4. Попередження екологічних криз. Моніторинг дозволяє попереджати та мінімізувати негативні екологічні наслідки катастроф, забруднень та інших екологічних подій. Швидке виявлення змін у довкіллі дозволяє приймати невідкладні заходи для збереження навколишнього середовища та захисту населення.

1.3 Аналіз сучасних вимог до моніторингових систем

Сучасні вимоги до моніторингових систем стають все більш вимогливими та різноманітними, оскільки швидкі зміни в технологіях, соціально-економічному середовищі та довкіллі вимагають постійного вдосконалення та адаптації існуючих систем[6].

Технічні вимоги:

1. Висока точність та надійність даних: Системи моніторингу повинні забезпечувати високу точність та надійність отриманих даних, особливо у вимірюваннях, які мають важливе значення для прийняття рішень[7].

2. Масштабованість: Сучасні системи моніторингу повинні бути масштабованими, щоб забезпечити ефективне функціонування як у великих, так і у малих масштабах.

3. Забезпечення відкритості та інтеграції: Вимога до відкритості та інтеграції дозволяє системам моніторингу взаємодіяти з іншими системами та програмами, що підвищує їх універсальність та ефективність.

Методологічні вимоги:

1. Стандартизація даних: Вимога до стандартизації даних дозволяє забезпечити їхню порівняльність та обмінність між різними системами та дослідниками.

2. Гнучкість та адаптивність: Сучасні системи моніторингу повинні бути гнучкими та адаптивними до змін у вимогах та умовах експлуатації[8].

Організаційні вимоги:

1. Ефективне управління даними: Системи моніторингу повинні мати ефективні засоби збору, збереження, обробки та аналізу великих обсягів даних.

2. Забезпечення доступу до інформації: Важливо, щоб дані, зібрані системами моніторингу, були доступні для зацікавлених сторін, що сприяє широкому використанню та впровадженню отриманих результатів.

Сучасні вимоги до моніторингових систем враховують не лише технічні аспекти, але й методологічні та організаційні чинники. Їх виконання дозволяє

забезпечити ефективне функціонування та використання систем моніторингу у різних сферах діяльності.

1.4 Висновки до розділу

Розділ 1 надає вступне знайомство з системами моніторингу, їх важливістю та сучасними вимогами до них.

Перше, визначення систем моніторингу розкриває їхню складність та значення у сучасному світі, де вони використовуються для збору та аналізу даних, необхідних для прийняття обґрунтованих рішень у різних сферах діяльності.

Друге, розгляд важливості моніторингу для сталого розвитку підкреслює роль систем моніторингу у забезпеченні екологічної стійкості та раціонального використання природних ресурсів, що є необхідним для збереження довкілля та підтримки життєздатності суспільства.

Третє, аналіз сучасних вимог до моніторингових систем показує, що вони повинні бути технічно ефективними, методологічно валідними та організаційно гнучкими для відповіді на зростаючі потреби суспільства та швидкі зміни у технологічному середовищі.

Загалом, розділ 1 визначає ключові аспекти та вимоги до систем моніторингу, створюючи основу для подальшого розгляду їхнього застосування та впровадження у досліджуваній області. Це дозволяє розуміти значення систем моніторингу як інструменту для досягнення сталого розвитку та збереження природних ресурсів.

РОЗДІЛ 2. ОГЛЯД ТА ПОРІВНЯННЯ ПРИСТРОЇВ МОНІТОРИНГУ КЛІМАТИЧНИХ ПОКАЗНИКІВ СЕРЕДОВИЩА

2.1 Огляд існуючих аналогів пристроїв моніторингу кліматичних показників середовища

2.1.1 Опис існуючих систем

Існує різноманітність пристроїв та систем для моніторингу кліматичних показників середовища[9]. Деякі з них відрізняються високою точністю та розширеними можливостями, в той час як інші є простішими та доступними для використання в домашніх умовах.

Крім того, існують портативні метеостанції, які зручно використовувати в походах, кемпінгах або просто на прогулянках. Ці компактні пристрої легко переносити і надають можливість отримувати основні показники погоди навіть у віддалених місцях. Вони можуть бути оснащені різними сенсорами для вимірювання різноманітних параметрів, від температури повітря до рівня UV-випромінювання, що робить їх універсальними засобами для вивчення погоди та клімату в різних умовах. Нижче наведено огляд деяких існуючих аналогів:

2.1.2 Nest Learning Thermostat

Nest Learning Thermostat [10] є одним із найпопулярніших пристроїв для моніторингу температури в домі. Він використовує технологію навчання, щоб аналізувати звички користувача та автоматично регулювати температуру, коли він перебуває вдома. При відсутності користувача, він автоматично знижує температуру, що сприяє економії енергії. Пристрій може бути підключений до Wi-Fi, що дозволяє контролювати температуру зі смартфона. Крім того, Nest оснащений додатковими датчиками для вимірювання вологості та інших показників середовища. Зовнішній вигляд Nest Learning Thermostat показаний на рисунку 2.1.



Рисунок 2.1 – Зовнішній вигляд Nest Learning Thermostat 1.2.1 Nest Learning Thermostat

Характеристики Nest Learning Thermostat:

- дисплей: 5.3см, 480x480, 24біта;
- датчики: температури, вологості, руху;
- живлення: вбудований акумулятор;
- бездротове підключення: wifi, weave[11];
- точність вимірювання: похибка температури – 0,05°с, вологості – 1%
- енергоспоживання: до 1 квт/міс.



Рисунок 2.2 – Приклад управління Nest Learning Thermostat

Керування Nest Learning Thermostat може здійснюватися за допомогою кнопок та поворотного кільця, розташованих безпосередньо на пристрої. Крім того, його можна керувати за допомогою мобільного додатку на смартфоні. На рисунку 2.2 показано приклад управління Nest Learning Thermostat.

2.1.3 Ecobee Smart Thermostat

Ecobee Smart Thermostat [12] є ще одним популярним термостатом, який може бути підключений до Wi-Fi для дистанційного керування температурою. Він також оснащений додатковими датчиками для вимірювання вологості та може навчитися вашим звичкам, автоматично адаптуючись до них. Зовнішній вигляд Ecobee Smart Thermostat показаний на рисунку 2.3.



Рисунок 2.3 – Зовнішній вигляд Ecobee Smart Thermostat

Характеристики Ecobee Smart Thermostat:

- дисплей: 3.5-дюймовий кольоровий сенсорний lcd-дисплей;
- роздільна здатність: 320 x 480 пікселів;
- температурний діапазон: від 0 °с до 55 °с;
- діапазон вологості: 5% - 95%.

Управління Ecobee Smart Thermostat здійснюється за допомогою сенсорного екрану, розташованого на передній панелі пристрою. На рисунку 1.4 показано приклад управління Ecobee Smart Thermostat.



Рисунок 2.4 – Приклад управління Ecobee Smart Thermostat

2.1.4 Netatmo Weather Station

Netatmo Weather Station [13] - це пристрій, призначений для моніторингу температури, вологості, рівня CO₂ та забруднення повітря в приміщенні. Він може бути підключений до Wi-Fi, що дозволяє відстежувати ці показники зі смартфона. Також він може надсилати повідомлення на телефон, якщо виникає необхідність у регулюванні якості повітря. Однією з переваг цього пристрою є його доступність і можливість додавати до нього до 3 додаткових модулів для моніторингу в інших кімнатах. Однак, недоліком є відсутність вимірювання рівня радону. Зовнішній вигляд Netatmo Weather Station показаний на рисунку 2.5.



Рисунок 2.5 – Зовнішній вигляд Netatmo Weather Station

Характеристики Netatmo Weather Station:

- барометр: 260 - 1160 мбар, похибка 1мбар;
- датчик вологості повітря: 0% - 100%, похибка 3%;
- датчик CO₂: 0 – 5000 ppm[14], похибка 5%;
- датчик температури повітря: 0°с - 50°с, похибка 0,3°с;
- датчик рівня шуму: 35 – 110 дб.



Рисунок 1.6 – Приклад управління Netatmo Weather Station

Управління Netatmo Weather Station здійснюється за допомогою мобільного додатку на смартфоні. Приклад управління Netatmo Weather Station показано на рисунку 2.6.

2.1.5 Airthings Wave Plus

Airthings Wave Plus [15] - це пристрій, який моніторить рівень CO₂, вологості, радону, температури та тиску повітря. Він може бути підключений до Wi-Fi та надсилати повідомлення на телефон, якщо якість повітря досягає небезпечних рівнів. Перевагою цього пристрою є його багатофункціональність та легка установка. Проте, недоліком є його висока ціна порівняно з іншими подібними пристроями. Зовнішній вигляд Airthings Wave Plus показаний на рисунку 2.7.

Управління Airthings Wave Plus здійснюється за допомогою мобільного додатку на смартфоні. Крім того, на передній частині пристрою є датчик руху та світлодіоди. При спрацюванні датчика руху світлодіоди змінять свій колір в залежності від якості повітря в приміщенні. На рисунку 2.8 показано приклад управління Airthings Wave Plus.



Рисунок 2.7 – Зовнішній вигляд Airthings Wave Plus

Характеристики Airthings Wave Plus:

- бездротовий зв'язок: bluetooth, Wi-Fi;
- датчик вологості повітря: 10% - 80%, похибка 3%;

- датчик температури повітря: 4°c - 50°c, похибка 0,3°c;
- датчик вимірювання радону: 0 - 500 bq/m³ похибка 5%;
- датчик вимірювання CO₂: 0 – 5000 ppm похибка 5%.

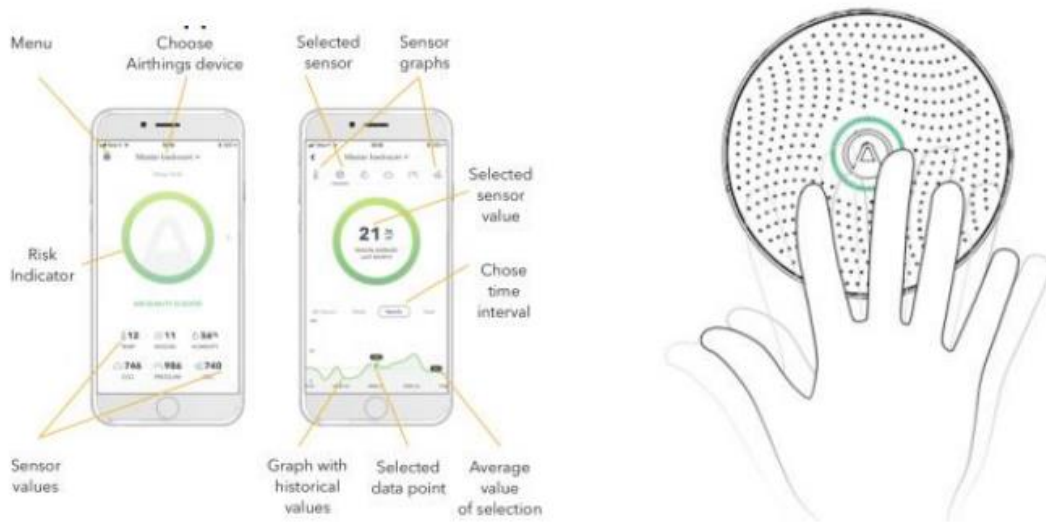


Рисунок 2.8 – Приклад управління Airthings Wave Plus

2.1.6 Foobot

Foobot [16] - це пристрій, який моніторить рівень CO₂, VOC [17], температуру, вологість та пил в приміщенні. Він може бути підключений до Wi-Fi та надсилати повідомлення на телефон, якщо виникає потреба у регулюванні якості повітря. Перевагою цього пристрою є його доступність та можливість підключатися до різних інших пристроїв для керування якістю повітря в будинку. Крім того, він має аналітичну функцію, яка надає рекомендації щодо регулювання якості повітря в приміщенні. Недоліком може бути те, що деякі користувачі відзначають незначну затримку у відображенні даних на дисплеї пристрою. Управління пристроєм здійснюється за допомогою мобільного додатку на смартфоні. Зовнішній вигляд та управління Foobot показано на рисунку 2.9.



Рисунок 2.9 – Зовнішній вигляд та управління Foobot

Характеристики Foobot:

- датчики: датчик для вимірювання рівня CO₂, tvoc (загальна кількість летючих органічних сполук), температури, вологості та pm2.5 (частки розміром менше 2,5 мікрометрів);
- взаємодія з користувачем: здійснюється за допомогою мобільного додатку foobot (доступний для ios та android);
- живлення: вбудований літій-іонний акумулятор з можливістю заряджання через usb-порт або за допомогою мережевого адаптера;
- бездротовий зв'язок: wi-fi 802.11 b/g/n;
- сумісність: foobot сумісний з більшістю систем автоматичного управління кліматом, таких як nest, ecobee, ifttt, amazon alexa та google assistant.

2.2 Порівняння оглянутих існуючих аналогів

Було оглянуто декілька існуючих аналогів систем/пристроїв моніторингу кліматичних показників середовища. Кожен з них має свої переваги та недоліки, а також різноманітні тонкощі роботи. Порівняння оглянутих аналогів показане в таблиці 2.1.

Таблиця 2.1. Порівняння існуючих аналогів

№	Назва	Номенклатура вимірювань	Бездрото вий зв'язок	Ціна \$	Спосіб управління
1	Nest Learning Thermostat	Температура: 0°C - 40°C, ±0,5°C; Вологість: 20% - 90%, ±1%; Датчик руху.	Wi-Fi	200 - 250	Мобільний додаток, керування на пристрої
2	Ecobee Smart Thermostat	Температура: 0°C - 55°C, ±0,5°C; Вологість: 5% - 95%, ±3%; Датчик присутності; Датчик ультрафіолетового випромінювання (UV).	Wi-Fi	200 - 250	Мобільний додаток, керування на пристрої
3	Netatmo Weather Station	Температура: 0°C - 50°C, похибка 0,3°C; Вологість: 0% - 100%, похибка 3%; Барометр: 260 - 1160 мбар, похибка 1 мбар	Wi-Fi	150 - 200	Мобільний додаток
4	Airthings Wave Plus	Температура: 4°C - 50°C, похибка 0,3°C; Вологість: 10% - 80%, похибка 3%; Радон: 0 - 500 bq/m ³ похибка 5%;	Wi-Fi, Bluetooth	200 - 250	Мобільний додаток
5	Foobot	Температура: -10°C - 50°C, похибка 0,3°C; Вологість: 0% - 100%, похибка 3%;	Wi-Fi	200 - 250	Мобільний додаток

Порівняльний аналіз показав, що кожен з оглянутих аналогів має свої переваги та недоліки, які варто врахувати при виборі оптимального рішення для конкретної задачі моніторингу кліматичних показників середовища. Наприклад, один з пристроїв може мати більшу точність вимірювань, але при цьому бути витратним у використанні та складним у встановленні. У той же час інший пристрій може бути більш доступним та простим у використанні, але при цьому мати обмежені можливості з точності вимірювань.

Вибір оптимального аналогу залежить від конкретних потреб та вимог

досліджуваної області, а також від фінансових можливостей та технічних особливостей, які можуть варіюватися в залежності від умов застосування. Важливо уважно зважити на всі переваги та недоліки кожного аналогу, а також врахувати специфіку завдання перед прийняттям рішення щодо вибору оптимального пристрою для моніторингу кліматичних показників середовища.

2.3 Обґрунтування вибору апаратних та програмних засобів для побудови системи

Мета даної роботи полягає у проектуванні та розробці пристрою моніторингу кліматичних показників середовища, який передає дані на віддалений сервер. Для досягнення цієї мети необхідно реалізувати як апаратну, так і програмну частини системи. Апаратна частина включатиме в себе набір мікросхем та датчиків, що забезпечить збір потрібних даних. Програмна частина передбачає інструкції для взаємодії між платами та датчиками, підключення до зовнішньої мережі, передачу даних на віддалений сервер, а також створення веб-сайту на віддаленому сервері для відображення зібраних даних у вигляді статистичних графіків[18].

2.4 Вибір апаратних засобів реалізації системи

В якості центрального пристрою для системи було обрано плату ESP8266, і це рішення має наступні переваги.

- Легкість використання: NodeMCU ESP8266[19] має простий та зрозумілий інтерфейс, що дозволяє швидко та просто програмувати цей мікроконтролер без необхідності написання складних програм.
- Висока продуктивність: Модуль NodeMCU ESP8266 має потужну оброблювальну потужність та вбудований модуль Wi-Fi, що дозволяє створювати надійні та продуктивні мережі Інтернету речей.
- Низька вартість: NodeMCU ESP8266 є дуже доступною платформою, що робить її вигідним варіантом для різноманітних проектів з обмеженим бюджетом.

- Великий вибір готових бібліотек: Модуль NodeMCU ESP8266 має велику спільноту розробників, яка активно розвиває цю платформу, розробляє нові бібліотеки та програмне забезпечення, що дозволяє швидко та ефективно вирішувати будь-які проблеми з цією платформою.

- Вбудований USB-порт: NodeMCU ESP8266 має вбудований USB-порт, що дозволяє підключати його до комп'ютера для програмування та відладки.

- Гнучкість: Модуль NodeMCU ESP8266 можна програмувати на мові Lua або C++, що дозволяє використовувати цю платформу для різноманітних проєктів, включаючи Інтернет речей, домашню автоматизацію, робототехніку та інше.

- Широкий функціонал: NodeMCU ESP8266 має велику кількість різноманітних функцій та можливостей, що дозволяє створювати складні проєкти зі зв'язком через Wi-Fi, підключенням до датчиків та інших пристроїв.

Обраний комбінований датчик температури та вологості повітря DHT11 [20] має кілька переваг, серед яких:

- Низька вартість: DHT11 є дуже доступним датчиком, що робить його привабливим для використання в проєктах з обмеженим бюджетом.

- Простота використання: DHT11 має простий 3-х контактний інтерфейс, що дозволяє легко підключити його до будь-якого мікроконтролера або одноплатного комп'ютера. Його цифровий вихідний сигнал робить інтеграцію простою і зручною.

- Висока точність: DHT11 відзначається досить високою точністю вимірювання температури та вологості, що робить його придатним для застосувань, де потрібна точність.

- Надійність: DHT11 має досить високу надійність, що робить його відмінним вибором для різних промислових, наукових та домашніх застосувань.

- Широкі можливості використання: DHT11 можна використовувати для вимірювання температури та вологості в різних сферах, таких як керування кліматом, контроль вологості в ґрунті, моніторинг стану приміщення та інші.

2.5 Вибір інструментів розробки програмної частини системи

Для розробки програмної частини системи дистанційного моніторингу мікроклімату було обрано такі інструменти:

1. Модуль NodeMCU esp8266.

NodeMCU esp8266 є мікроконтролером з вбудованим модулем Wi-Fi, який забезпечує зручну можливість підключення до мережі Інтернет. Цей модуль обраний через його низьку вартість, компактність та можливість бездротового зв'язку, що робить його ідеальним варіантом для систем моніторингу, де потрібна передача даних на віддалений сервер.

2. Arduino IDE[21].

Arduino IDE - це середовище програмування, яке широко використовується для розробки програмного забезпечення для мікроконтролерів Arduino та сумісних платформ, включаючи NodeMCU esp8266. Обрано Arduino IDE через його простоту використання, наявність багатьох готових бібліотек та документації, а також широкий спектр підтримуваних мікроконтролерів.

3. Мова програмування C++.

Мова програмування C++ була обрана для розробки програмної частини системи через її велику популярність у сфері вбудованих систем та мікроконтролерів. Вона забезпечує високий рівень продуктивності та ефективності, а також можливість використання різноманітних бібліотек та інструментів для розробки.

Обраний набір інструментів дозволяє ефективно розробляти програмне забезпечення для системи дистанційного моніторингу мікроклімату, забезпечуючи надійність, продуктивність та зручність у використанні.

2.6 Постановка завдання

Після проведення огляду існуючих аналогів пристроїв моніторингу кліматичних показників середовища та визначення апаратних і програмних засобів реалізації системи, постановка завдання наступна:

1. Провести опис функціонування системи.

2. Розробити структурну схему системи.
3. Розробка діаграми процесів та моделювання роботи пристрою.
4. Збирання апаратної частини системи.
5. Програмування взаємодії апаратних частин системи.
6. Тестування апаратно-програмної взаємодії комплектуючих.
7. Організація дистанційного зв'язку.
8. Тестування системи.

2.7 Висновки до розділу

У другому розділі виконано аналіз предметної області, що включав огляд терміну "моніторинг" та його історію, а також визначення завдань моніторингу, таких як спостереження за біосферою, оцінка стану природного середовища та виявлення антропогенних впливів. Були розглянуті різні методи моніторингу кліматичних показників та порівняння існуючих аналогів систем моніторингу. Основними критеріями порівняння були номенклатура вимірювань, точність, використання бездротового зв'язку, ціна та способи взаємодії з користувачем.

Деякі апаратні засоби були вибрані для реалізації системи, включаючи плату Arduino UNO, модуль NodeMCU esp8266 та датчик температури та вологості повітря DHT11.

Завдяки проведеному дослідженню та обґрунтуванню вибору апаратних та програмних засобів була визначена постановка завдання на розробку системи.

РОЗДІЛ 3. ОПИС І ОБГРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис і моделювання системи дистанційного моніторингу кліматичних показників середовища

Головна ідея проекту полягає у створенні системи дистанційного моніторингу кліматичних показників середовища. Система складається з двох підсистем: перша - система моніторингу мікроклімату, друга - дистанційна передача та відображення результатів вимірювання.

Перша підсистема реалізується за допомогою мікроконтролера та підключених до нього датчиків, які вимірюють різноманітні кліматичні показники. Датчики реагують на зміни кліматичних умов і передають сигнали до контролера. Контролер отримує та обробляє ці сигнали.

Друга підсистема відповідає за передачу даних та їх відображення. Вона підключається до Інтернету через Wi-Fi та передає зібрані першою підсистемою дані на віддалений сервер. Користувач може дистанційно переглядати результати вимірювання в реальному часі з будь-якої точки з доступом до мережі Інтернет.

Модель системи дистанційного моніторингу кліматичного середовища зображена на рисунку 3.1.



Рисунок 3.1 – Модель системи дистанційного моніторингу кліматичних показників середовища

3.2 Розробка структурної схеми системи дистанційного моніторингу кліматичних показників середовища

Апаратна частина системи дистанційного моніторингу кліматичних показників складається з різних елементів. Перш за все, це датчики, що вимірюють показники мікроклімату. Наступним по важливості елементом є центральна плата для зчитування та обробки даних з датчика. Також в цій системі важливу роль відіграє бездротова передача даних на віддалений сервер, для чого використовується Wi-Fi модуль з мікроконтролером.

Апаратна частина системи дистанційного моніторингу кліматичних показників середовища складається з трьох компонентів:

1. Центральна плата для збирання та обробки даних з датчиків NodeMCU ESP8266 з Wi-Fi модулем.
2. Датчик вимірювання температури та вологості повітря DHT11.
3. Дисплей LCD 1602.

Структурна схема апаратної частини системи дистанційного моніторингу кліматичних показників середовища показана на рисунку 3.2.

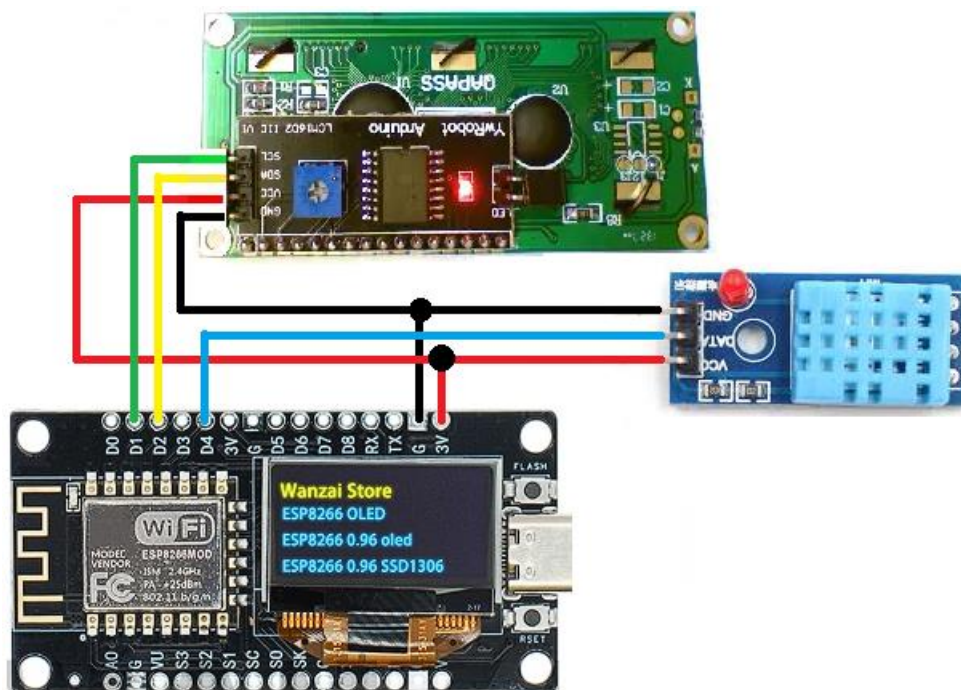


Рисунок 3.2 – Структурна схема апаратної частини системи дистанційного моніторингу кліматичних показників середовища

Принципова схема апаратної частини системи дистанційного моніторингу кліматичних показників середовища показана в додатку Б.

3.3 Розробка функціональної схеми системи дистанційного моніторингу кліматичних показників середовища

Виходячи з аналізу аналогів систем моніторингу мікроклімату, ми можемо сформулювати та описати принципи функціонування майбутньої системи дистанційного моніторингу кліматичних умов навколишнього середовища. Основною метою проекту є створення системи, яка збирає дані про стан мікроклімату в певній області та надсилає результати вимірювань на віддалений сервер. Після цього користувачі можуть підключитися до цього серверу через Інтернет та переглядати отримані дані у реальному часі. Система складається з двох основних компонентів: пристрою, який здійснює збір показників мікроклімату, та віддаленого серверу, який надає можливість дистанційного перегляду результатів вимірювань.

Виходячи з головної ідеї та проведених досліджень, можна описати наступні пункти функціонування системи:

- Система повинна збирати дані стану мікроклімату з підключених до неї датчиків та за необхідності їх обробляти.
- Система повинна мати модуль для бездротового зв'язку.
- Система повинна передавати дані від центральної плати до модуля бездротового зв'язку.
- Система повинна за допомогою модуля бездротового зв'язку підключатися до мережі.
- Система повинна за допомогою модуля бездротового зв'язку передавати дані моніторингу мікроклімату на віддалений сервер і відображати їх на ньому у реальному часі.

Відповідно до поставлених вимог можна виділити елементи пристрою, необхідні для функціонування системи:

- Центральна плата, що буде збирати та обробляти дані з підключених до неї датчиків.
- Датчик температури та вологості.
- Модуль бездротового зв'язку.
- Протокол передачі даних між центральною платою та модулем бездротового зв'язку.

Функціональна схема системи дистанційного моніторингу кліматичних показників середовища показано на рисунку 3.3.

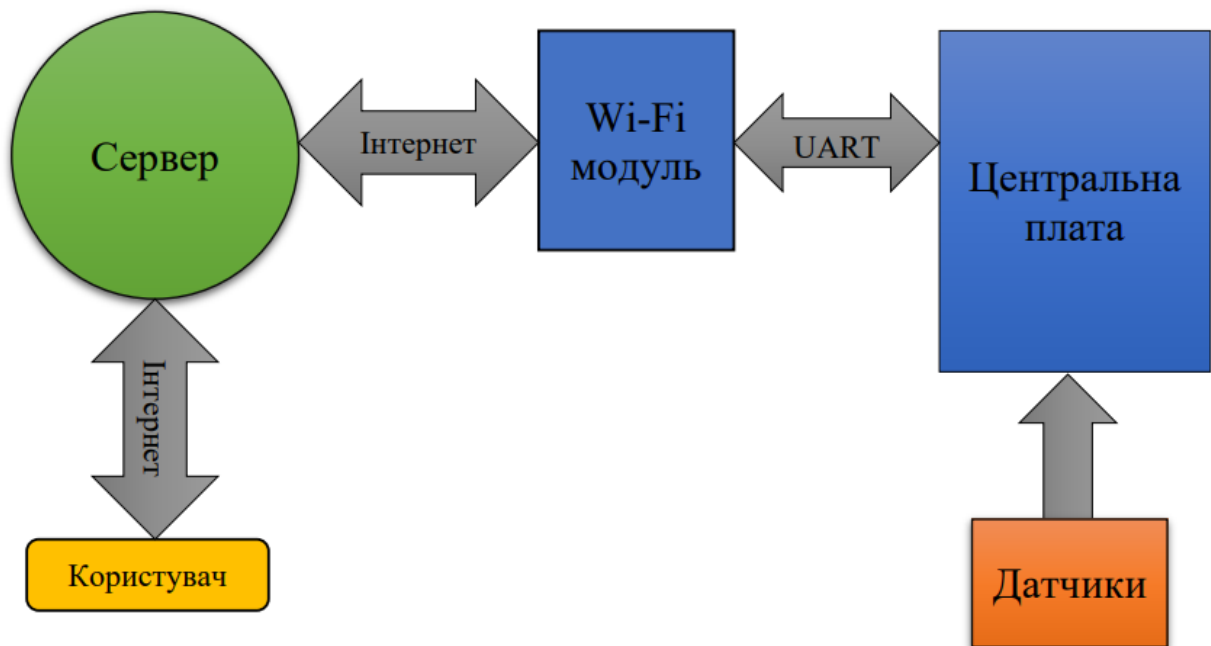


Рисунок 3.3 – Функціональна схема системи дистанційного моніторингу кліматичних показників середовища

3.4 Розробка діаграми процесів схеми системи дистанційного моніторингу кліматичних показників середовища, принципова схема апаратних частин системи

Апаратна частина системи дистанційного моніторингу кліматичних показників середовища складається з трьох основних компонентів.

NodeMCU ESP8266 - це модуль на базі мікроконтролера ESP8266 з вбудованим Wi-Fi, який має потужну оброблювальну потужність, вбудований

USB-порт, легкий у використанні інтерфейс та можливість програмування на мові Lua або C++. Зовнішній вигляд NodeMCU ESP8266 показано на рисунку 3.4.



Рисунок 3.4 – Зовнішній вигляд датчику NodeMCU ESP8266

Електрична схема NodeMCU ESP8266 включає наступні компоненти:

- Мікроконтролер ESP8266 модуля NodeMCU
- Вбудований USB-порт для програмування та відлагодження.
- Регулятор напруги AMS1117, що забезпечує стабільну напругу

живлення

- 3,3 В для ESP8266
- Конденсатори для стабілізації напруги живлення
- Кварцовий резонатор 26 МГц для ESP8266
- Кнопка для reset мікроконтролера
- LED для відображення стану роботи мікроконтролера

Електрична схема NodeMCU ESP8266 показана на рисунку 3.5.

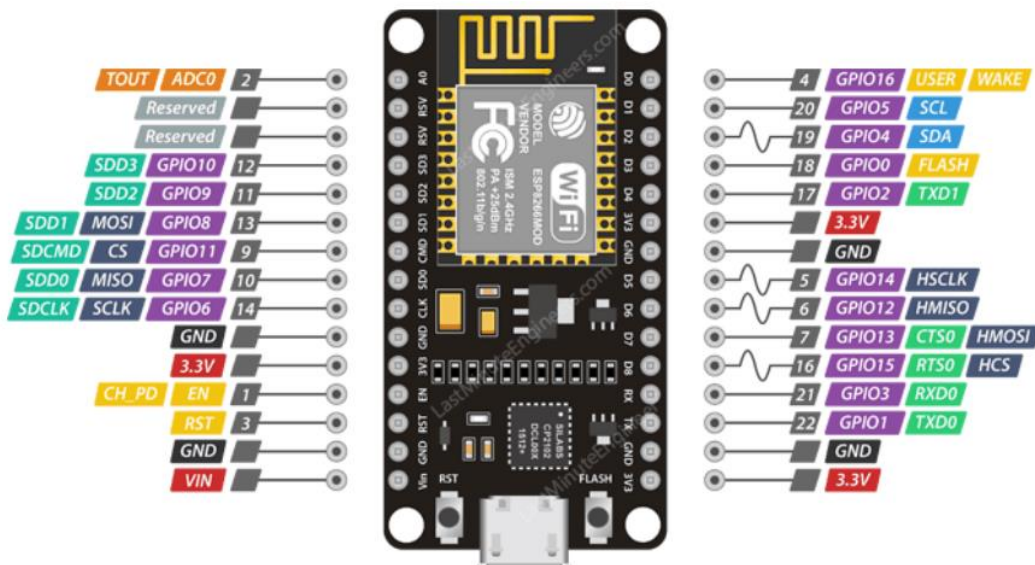


Рисунок 3.5 – Електрична схема NodeMCU ESP8266

Датчик температури та вологості повітря DHT11 - це електронний пристрій, який призначений для вимірювання температури та вологості в навколишньому середовищі. Він може вимірювати температуру у діапазоні від 0 до 50 градусів Цельсія та відносну вологість повітря від 20% до 90%. DHT11 має три виводи: земля, живлення та вихід даних. Вихід даних видає цифровий сигнал, який відповідає виміряним параметрам. Зовнішній вигляд датчика DHT11 зображено на рисунку 3.6.

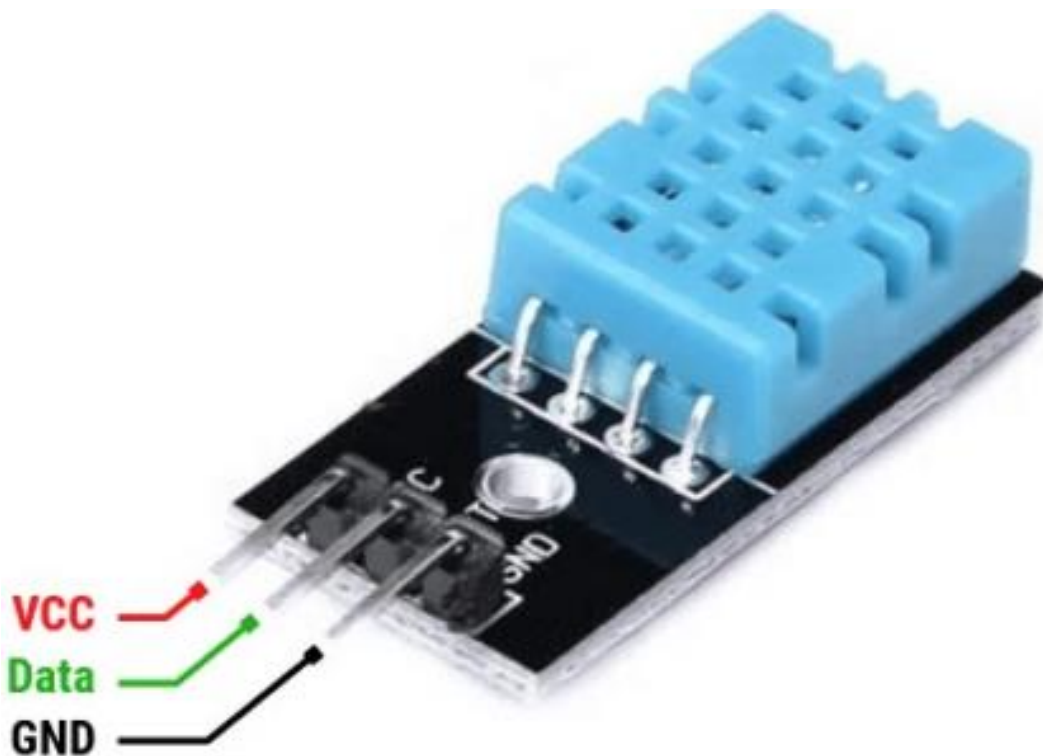


Рисунок 3.6 – Зовнішній вигляд датчика DHT11

Електрична схема DHT11 включає в себе три пінові виводи: VCC, GND та OUT.

- VCC (позначено червоним кольором на датчику) - пін живлення, який зазвичай підключають до джерела живлення напругою 3.3В або 5В.
- GND (позначено чорним кольором на датчику) - пін землі, який підключають до нуля джерела живлення.
- OUT (позначено жовтим кольором на датчику) - цифровий вихід, який передає дані від датчика до мікроконтролера або іншого пристрою. Вихід має високу або низьку рівень залежно від температури та вологості, які вимірює датчик.

Електрична схема DHT11 показана на рисунку 3.7.

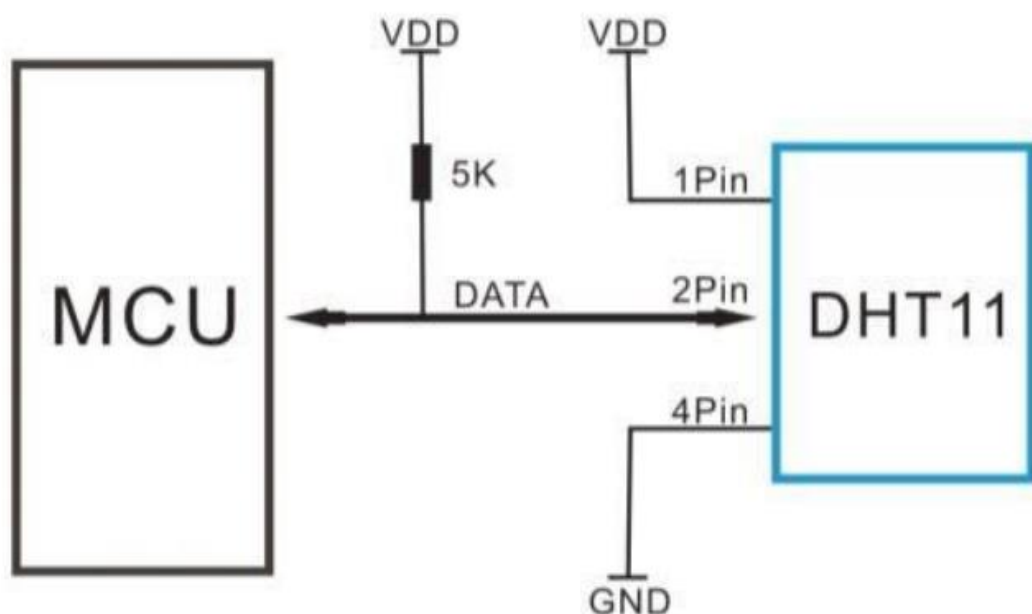


Рисунок 3.7 - Електрична схема DHT11

LCD1602 - це рідкокристалічний дисплей, який призначений для відображення текстової інформації. Він має роздільну здатність 16 символів на 2 рядки, що дає змогу відображати до 32 символів одночасно. Дисплей LCD1602 часто використовується в різних електронних проєктах для відображення повідомлень, даних з сенсорів та іншої інформації.

LCD1602 має кілька виводів, які дозволяють підключати його до мікроконтролера. Основні виводи включають:

1. VSS - земля (GND).

2. VDD - живлення (звичайно 5 В).
 3. V0 - регулювання контрастності дисплея.
 4. RS (Register Select) - вибір регістру (0 - команда, 1 - дані).
 5. RW (Read/Write) - режим читання/запису (звичайно підключається до землі для режиму запису).
 6. E (Enable) - увімкнення сигналу (запускає зчитування даних).
 7. D0-D7 - лінії даних (для передачі 8-бітних даних).
 8. A (Anode) - анод підсвітки (підключається до живлення через резистор).
 9. K (Cathode) - катод підсвітки (зазвичай підключається до землі).
- Для роботи з LCD1602 часто використовують 4-бітний або 8-бітний режими. У 4-бітному режимі використовується лише чотири лінії даних (D4-D7), що дозволяє заощадити на кількості використаних виводів мікроконтролера. Зовнішній вигляд дисплея LCD1602 зображено на рисунку 3.8.



Рисунок 3.8 - Зовнішній вигляд дисплея LCD1602

LCD1602 є популярним вибором для відображення інформації у різних проєктах через свою простоту використання, низьку вартість та зручність інтеграції з мікроконтролерами, такими як Arduino, Raspberry Pi та іншими.

3.5 Вибір хмарної платформи для відправлення даних на сервер

Для реалізації проекту системи дистанційного моніторингу кліматичних показників середовища на сервері був обраний сервіс ThingSpeak[22].

ThingSpeak - це відкрита платформа для інтернету речей (IoT), яка дозволяє збирати, візуалізувати та аналізувати потокові дані в реальному часі. Цей сервіс був розроблений компанією MATLAB для забезпечення зручного способу збору та обробки даних з різноманітних пристроїв та датчиків, з'єднаних з інтернетом. Вигляд сервісу зображений на рисунку 3.9.

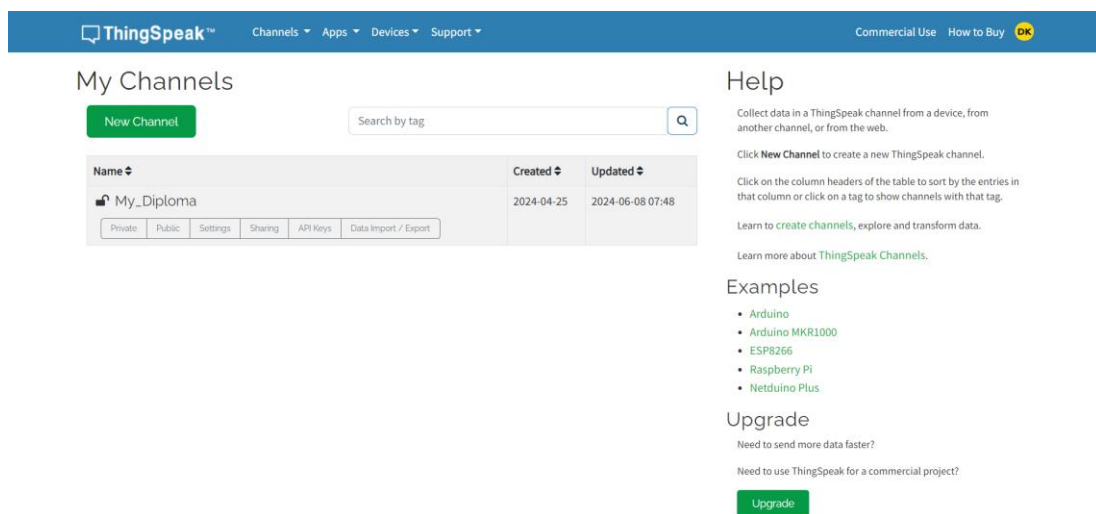


Рисунок 3.9 - Зовнішній вигляд сервісу ThingSpeak

Вибір ThingSpeak був обумовлений рядом переваг, які він пропонує:

1. Відкритий вихідний код: ThingSpeak є безкоштовною платформою з відкритим вихідним кодом, що забезпечує гнучкість та можливість налаштування під конкретні потреби проекту.

2. Інтеграція з популярними платформами IoT: ThingSpeak легко інтегрується з такими популярними платформами, як Arduino, Raspberry Pi, BeagleBone та іншими, що дозволяє відправляти дані з різноманітних пристроїв та датчиків, які використовуються в системі моніторингу.

3. Зручний веб-інтерфейс: Сервіс пропонує зручний веб-інтерфейс для управління каналами даних, візуалізації та аналізу інформації, що спрощує процес моніторингу та обробки даних кліматичних показників.

4. Аналітичні інструменти: ThingSpeak надає вбудовані інструменти для аналізу даних, включаючи функції для обчислення статистичних показників,

виконання математичних операцій та створення користувацьких аналітичних додатків за допомогою MATLAB, що дозволяє глибоко аналізувати зібрані кліматичні дані.

5. Масштабованість: Платформа дозволяє зберігати великі обсяги даних і легко масштабувати систему для обробки зростаючих потоків інформації від датчиків моніторингу.

6. Підтримка різних протоколів: ThingSpeak підтримує різні протоколи передачі даних, такі як HTTP, MQTT та інші, забезпечуючи гнучкість у виборі способу відправки даних від системи моніторингу.

Порівняно з іншими подібними сервісами... (тут можна продовжити порівняння, як у попередньому відповіді).

Таким чином, ThingSpeak став ідеальним вибором для розгортання проекту системи дистанційного моніторингу кліматичних показників завдяки своїй відкритості, простоті інтеграції з апаратними компонентами, зручності аналізу даних та можливостям масштабування.

3.6 Висновки до розділу

У другому розділі детально розглянуто різні аспекти системи дистанційного моніторингу кліматичних показників середовища. Описано та змодельовано систему, розроблено структурну і функціональну схеми, а також діаграму процесів і принципову схему апаратних частин.

Структурна схема системи дозволила визначити взаємозв'язки між компонентами та послідовність обробки даних, що є основою для подальшої розробки. Функціональна схема визначила функції кожного елементу та їх взаємодію, а діаграма процесів розкрила робочі процеси системи. Принципова схема апаратних частин детально описала апаратну архітектуру системи.

РОЗДІЛ 4. ПРАКТИЧНА РЕАЛІЗАЦІЯ РОБОТИ

4.1 Реалізація апаратної частини системи

Система дистанційного моніторингу кліматичних показників середовища складається з двох компонентів:

1. Центральна плата з Wi-Fi модулем NodeMCU ESP8266.
2. Датчик для вимірювання кліматичних показників середовища DHT11.

Датчик DHT11 може вимірювати температуру від 0 до 50 градусів Цельсія та відносну вологість повітря від 20% до 90%. Він підключений до центральної плати NodeMCU ESP8266, яка отримує дані вимірювань і обробляє їх для подальшого використання. Ця плата містить Wi-Fi модуль, який використовується для бездротової передачі даних. Зовнішній вигляд зібраної апаратної частини системи дистанційного моніторингу кліматичних показників середовища показаний на рисунку 4.1.

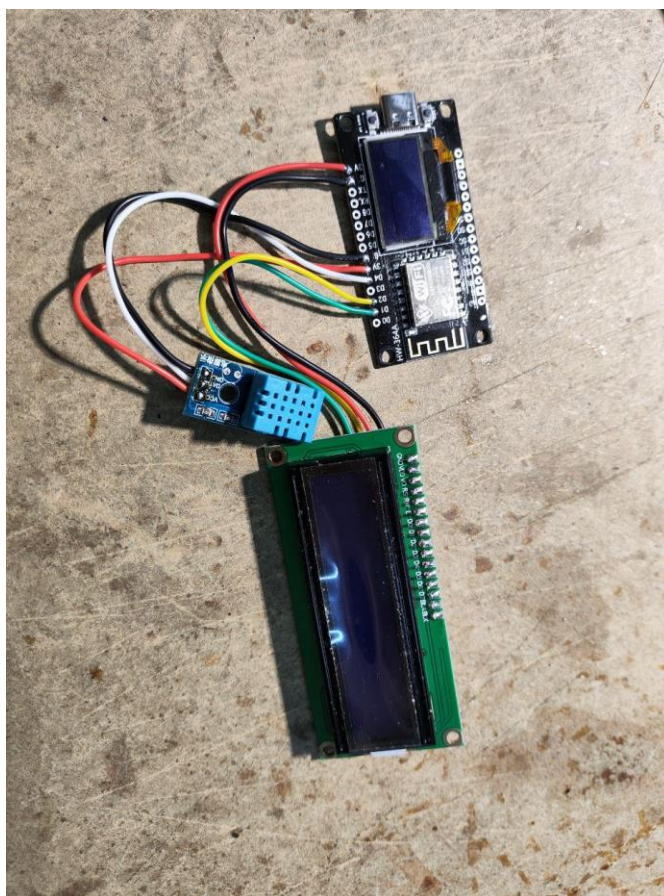


Рисунок 4.1 – Зовнішній вигляд апаратної частини системи дистанційного моніторингу кліматичних показників середовища

4.2 Реалізація програмної частини системи

Система дистанційного моніторингу кліматичних показників середовища складається з однієї програмованої частини NodeMCU ESP8266.

Для її програмування використовувались інструмент Arduino Ide та мова програмування C++.

Програмування плати NodeMCU ESP8266 здійснювалось в середовищі Arduino IDE на мові C++. Задача плати NodeMCU ESP8266 полягає в отриманні даних від DHT11, та їх обробки. Також NodeMCU ESP8266 повинна підключатися до мережі Інтернет за допомогою бездротового зв'язку Wi-Fi та передавати дані до віддаленої бази даних.

Початкова частина програми включає ініціалізацію змінних та підключення необхідних бібліотек. Серед змінних ініціалізації вказуються ті, які використовуються для зберігання та обробки отриманих даних від центральної плати, змінні для взаємодії з мережею, змінні для роботи з базою даних та інші допоміжні змінні. Скріншот початку коду ініціалізації зображено на рисунку 4.2.

```
#include <ESP8266WiFi.h>
#include <ESPAsyncTCP.h>
#include <ESPAsyncWebServer.h>
#include <DHT.h>
#include <LiquidCrystal_I2C.h>
#include <Wire.h>
```

Рисунок 4.2 – Підключення бібліотек

Блок коду зображений на рисунку 3.2 підключає необхідні бібліотеки для роботи з ESP8266, веб-сервером, дисплеєм LCD та датчиком DHT. А саме:

1. ESP8266WiFi.h для роботи з Wi-Fi на ESP8266.
2. ESPAsyncTCP.h і ESPAsyncWebServer.h для асинхронного веб-сервера.
3. DHT.h для роботи з датчиком DHT11.
4. LiquidCrystal_I2C.h і Wire.h для роботи з LCD дисплеєм через I2C.

```

/* Введіть ваші SSID і пароль */
const char* ssid = "POCO_F5"; // SSID
const char* password = "12344321"; // пароль

// Write API ключ ThingSpeak
const char* apiKey = "WSAVW7L5ATMWOA20";

AsyncWebServer server(80);

// LCD на адресі 0x27
LiquidCrystal_I2C lcd(0x27, 16, 2);

float Temperature;
float Humidity;

// датчик DHT
#define DHTPIN 2
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);

```

Рисунок 4.3 – Оголошення констант

Блок коду зображений на рисунку 4.3 оголошує константи:

1. ssid та password – змінні для зберігання назви та паролю Wi-Fi мережі.
2. apiKey – API ключ для відправки даних на платформу ThingSpeak.
3. server – об'єкт асинхронного веб-сервера.
4. lcd – об'єкт для керування LCD дисплеєм через I2C інтерфейс.
5. Temperature та Humidity – змінні для зберігання значень температури та вологості.
6. DHTPIN та DHTTYPE – визначення піну та типу датчика DHT.
7. dht – об'єкт для керування датчиком DHT.

```

void setup()
{
  Serial.begin(115200);
  delay(100);

  lcd.init();           // Ініціалізуємо LCD екран
  lcd.backlight();     // Вмикаємо підсвітку

  lcd.setCursor(0, 0); // Переміщаємо курсор на початок першого рядка
  lcd.print("Hello");  // Виводимо "Hello" на LCD

  lcd.setCursor(0, 1); // Переміщаємо курсор на початок другого рядка
  lcd.print("connecting..."); // Виводимо "connecting..." на LCD

  pinMode(DHTPIN, INPUT);

  dht.begin();

  Serial.println("Connecting to ");
  Serial.println(ssid);

  // Connect to the local Wi-Fi network
  WiFi.begin(ssid, password);

  // Check if the Wi-Fi module is connected to the Wi-Fi network
  while (WiFi.status() != WL_CONNECTED)
  {
    delay(1000);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected..!");
  Serial.print("Got IP: "); Serial.println(WiFi.localIP());

  lcd.clear();         // Clear the screen
  lcd.setCursor(0, 0); // Move the cursor to the beginning of the first row
  lcd.print("Connected"); // Print "Connected" on the LCD
  lcd.setCursor(0, 1); // Move the cursor to the beginning of the second row
  lcd.print("successfully"); // Print "successfully" on the LCD

  server.on("/", HTTP_GET, [Temperature, Humidity](AsyncWebServerRequest *request) {
    String htmlResponse = SendHTML(Temperature, Humidity);
    request->send(200, "text/html", htmlResponse);
  });

  server.begin();
  Serial.println("HTTP server started");
}

```

Рисунок 4.4 – Функція «setup()»

Блок коду зображений на рисунку 4.4 виконує початкову ініціалізацію системи, а саме:

1. Serial.begin(115200): Ініціалізація серійного зв'язку з швидкістю 115200 бод.
2. lcd.init(): Ініціалізація LCD дисплею.
3. lcd.backlight(): Увімкнення підсвітки LCD дисплею.
4. lcd.setCursor(0, 0) та lcd.print("Hello"): Виведення тексту "Hello" на першому рядку LCD дисплею.
5. lcd.setCursor(0, 1) та lcd.print("connecting..."): Виведення тексту "connecting..." на другому рядку LCD дисплею.
6. pinMode(DHTPIN, INPUT): Встановлення піна DHT як вхідного.
7. dht.begin(): Ініціалізація датчика DHT.
8. WiFi.begin(ssid, password): Підключення до Wi-Fi мережі.

9. `while (WiFi.status() != WL_CONNECTED)`: Очікування підключення до Wi-Fi.
10. Виведення стану підключення до Wi-Fi на серійний монітор та LCD дисплей.
11. Налаштування обробника HTTP GET запитів для кореневої сторінки.
12. `server.begin()`: Запуск веб-сервера.

```

void loop()
{
    // Обробка клієнтів
    // server.handleClient(); // Виправлена стрічка

    // Отримання температури та вологості
    Temperature = dht.readTemperature(); // отримати значення температури
    Humidity = dht.readHumidity();      // отримати значення вологості

    // Відправка на ThingSpeak
    sendToThingSpeak(Temperature, Humidity);

    delay(5000); // Затримка для оновлення даних

    // Відображення на дисплеї
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Temp: ");
    lcd.print(Temperature);
    lcd.print(" C");

    lcd.setCursor(0, 1);
    lcd.print("Humidity: ");
    lcd.print(Humidity);
    lcd.print(" %");

    // Відправка на ThingSpeak
    sendToThingSpeak(Temperature, Humidity);

    delay(5000); // Затримка для оновлення даних
}

```

Рисунок 4.5 – Функція «loop()»

Блок коду зображений на рисунку 4.5 виконує основний цикл програми:

1. `Temperature = dht.readTemperature()`: Зчитування температури з датчика DHT.
2. `Humidity = dht.readHumidity()`: Зчитування вологості з датчика DHT.
3. `sendToThingSpeak(Temperature, Humidity)`: Відправка даних на ThingSpeak.
4. `delay(5000)`: Затримка в 5 секунд для оновлення даних.
5. Оновлення дисплея новими значеннями температури та вологості.

```

void sendToThingSpeak(float temp, float hum) {
    if (WiFi.status() != WL_CONNECTED) {
        Serial.println("WiFi not connected");
        return;
    }

    WiFiClient client;
    const int httpPort = 80;
    if (!client.connect("api.thingspeak.com", httpPort)) {
        Serial.println("Connection failed");
        return;
    }

    String url = "/update?api_key=";
    url += apiKey;
    url += "&field1=";
    url += String(temp);
    url += "&field2=";
    url += String(hum);

    Serial.print("Requesting URL: ");
    Serial.println(url);

    client.print(String("GET ") + url + " HTTP/1.1\r\n" +
                "Host: api.thingspeak.com\r\n" +
                "Connection: close\r\n\r\n");

    delay(500); // Затримка для дочеку, поки дані відправляються

    while (client.available()) {
        String line = client.readStringUntil('\r');
        Serial.print(line);
    }
}

```

Рисунок 4.6 – Функція «sendToThingSpeak()»

Блок коду зображений на рисунку 4.6 відповідає за відправку даних на сервіс ThingSpeak і включає такі функції:

1. Перевірка підключення до Wi-Fi.
2. Створення клієнта Wi-Fi.
3. Підключення до сервера ThingSpeak.
4. Формування HTTP GET запиту для відправки даних.
5. Відправка запиту на сервер ThingSpeak.

Функція sendHTML (додаток А) сторінки для веб-сервера і виконує такі функції:

1. Формує HTML структуру для відображення значень температури та вологості.

2. Використовує JavaScript для автоматичного оновлення даних на сторінці.

4.3 Підключення системи до сервісу ThingSpeak

У цьому підрозділі представлено практичні кроки та приклади коду, використані для інтеграції системи моніторингу кліматичних показників з платформою ThingSpeak. Система базується на мікроконтролері ESP8266 та датчику DHT11 для вимірювання температури та вологості.

1. Реєстрація облікового запису MathWorks

Першим кроком є реєстрація облікового запису MathWorks на сайті <https://www.mathworks.com>.

2. Доступ до ThingSpeak

Після створення облікового запису MathWorks потрібно перейти на сайт <https://thingspeak.com> та авторизуватися через створений обліковий запис як показано на рисунку 4.8.

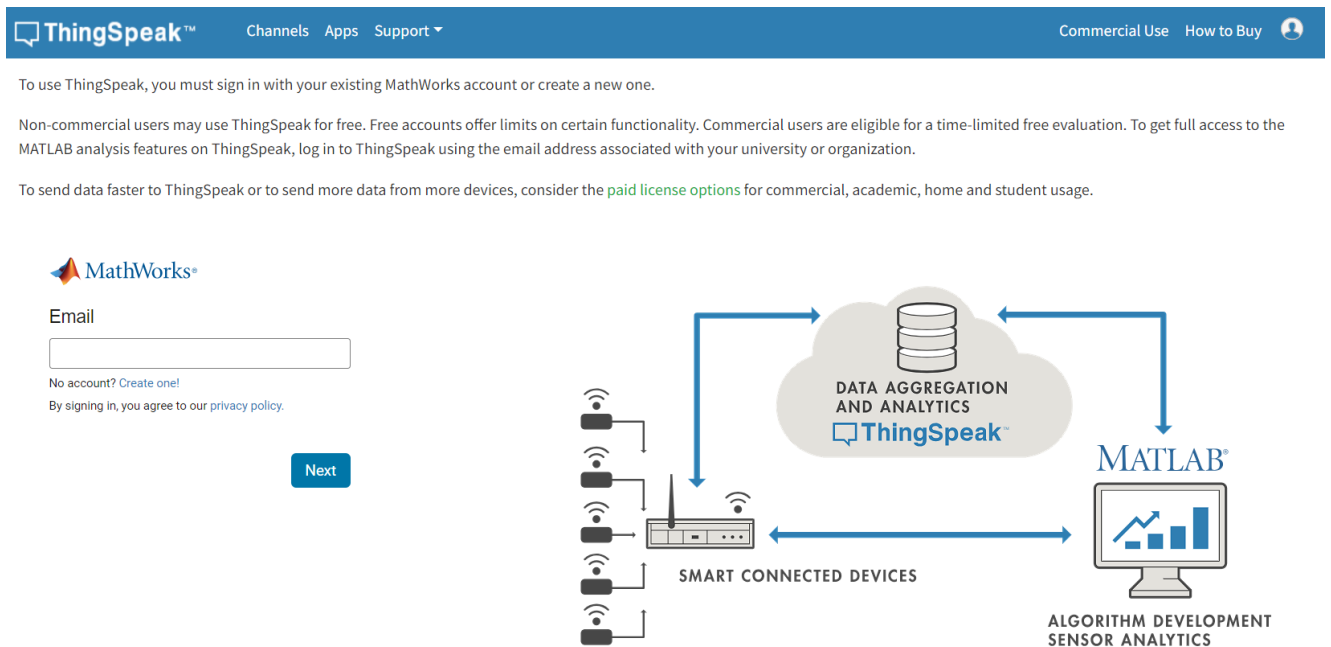


Рисунок 4.8 – Авторизація в системі ThingSpeak

3. Створення нового каналу

Далі потрібно створити новий канал для збору даних з системи моніторингу. Для цього необхідно перейти в розділ "Channels" і натиснути "New Channel". На

цьому етапі з'явиться форма для заповнення деталей нового каналу.

Спочатку потрібно задати ім'я каналу, яке буде його унікальним ідентифікатором. Рекомендується обирати коротке та змістовне ім'я, що відображає призначення каналу, наприклад, "Моніторинг_кліматичних_даних".

Далі необхідно додати опис каналу, який може містити більш детальну інформацію про систему моніторингу, місце розташування датчиків, тип збираних даних тощо. Опис дозволить краще ідентифікувати канал у майбутньому.

Наступним кроком є визначення полів даних, які будуть зберігатися в каналі. Платформа ThingSpeak підтримує до 8 полів даних для кожного каналу. У випадку системи моніторингу кліматичних показників, варто вибрати два поля, наприклад, "Температура" та "Вологість". Під час вибору потрібно також вказати одиниці вимірювання для кожного поля, наприклад, °C для температури та % для вологості.

Після заповнення всіх необхідних деталей, потрібно натиснути кнопку "Create Channel" для завершення процесу створення. Процес створення нового каналу показаний на рисунку 4.9

The screenshot shows the 'New Channel' form in ThingSpeak. The form is divided into several sections:

- Name:** A text input field containing 'My_Diploma'.
- Description:** A large text area for entering a description.
- Field 1:** A dropdown menu with 'Temperature' selected and a checked checkbox.
- Field 2:** A dropdown menu with 'Humidity' selected and a checked checkbox.
- Field 3:** An empty dropdown menu with an unchecked checkbox.
- Field 4:** An empty dropdown menu with an unchecked checkbox.
- Field 5:** An empty dropdown menu with an unchecked checkbox.
- Field 6:** An empty dropdown menu with an unchecked checkbox.
- Field 7:** An empty dropdown menu with an unchecked checkbox.
- Field 8:** An empty dropdown menu with an unchecked checkbox.
- Metadata:** A text input field at the bottom.

On the right side, there is a 'Help' section with the following text:

Channels store all the data that a ThingSpeak application collects. Each channel includes eight fields that can hold any type of data, plus three fields for location data and one for status data. Once you collect data in a channel, you can use ThingSpeak apps to analyze and visualize it.

Channel Settings

- Percentage complete:** Calculated based on data entered into the various fields of a channel. Enter the name, description, location, URL, video, and tags to complete your channel.
- Channel Name:** Enter a unique name for the ThingSpeak channel.
- Description:** Enter a description of the ThingSpeak channel.
- Field#:** Check the box to enable the field, and enter a field name. Each ThingSpeak channel can have up to 8 fields.
- Metadata:** Enter information about channel data, including JSON, XML, or CSV data.
- Tags:** Enter keywords that identify the channel. Separate tags with commas.
- Link to External Site:** If you have a website that contains information about your ThingSpeak channel, specify the URL.
- Show Channel Location:**
 - Latitude:** Specify the latitude position in decimal degrees. For example, the latitude of the city of London is 51.5072.
 - Longitude:** Specify the longitude position in decimal degrees. For example, the longitude of the city of London is -0.1275.

Рисунок 4.9 – Створення нового каналу

4. Генерація API ключа

Після створення каналу необхідно згенерувати API ключ, який буде використовуватися для відправки даних на ThingSpeak. Потрібно перейти на сторінку свого каналу та знайти розділ "API Keys". У цьому розділі будуть

відображатися два типи API ключів: "Write API Key" та "Read API Key".

"Write API Key" використовується для надсилання та запису даних на канал ThingSpeak. Цей ключ потрібно скопіювати та зберегти в безпечному місці, оскільки він буде використовуватися в коді програми для відправки даних з системи моніторингу на ThingSpeak.

З міркувань безпеки платформа ThingSpeak не показує "Write API Key" повністю на сторінці. Замість цього відображається лише частина ключа, як показано на рисунку 4.10. Щоб отримати повний ключ, потрібно натиснути на кнопку "Re-generate API Keys", після чого ThingSpeak надасть новий повний "Write API Key".

"Read API Key" використовується для читання та отримання даних з каналу ThingSpeak. Цей ключ не потрібен для відправки даних, але може знадобитися, якщо буде потреба читати дані з каналу за допомогою зовнішніх додатків або скриптів.

Після копіювання "Write API Key" його необхідно вставити у відповідну змінну в коді програми для системи моніторингу. Це дозволить програмі успішно відправляти дані на створений канал ThingSpeak.

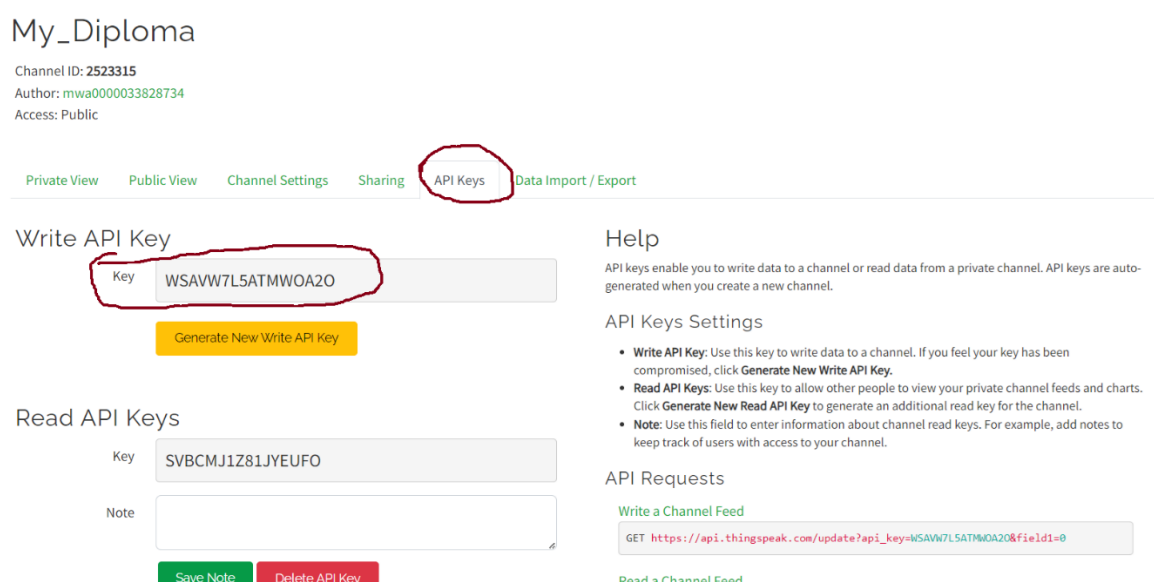


Рисунок 4.10 – Створення API ключа

5. Конфігурування системи моніторингу

Система моніторингу базується на мікроконтролері ESP8266 та датчику

DHT11. Потрібно підключити систему до локальної Wi-Fi мережі, використовуючи SSID та пароль, вказані у відповідних змінних коду (рисунок 4.11).

```
const char* ssid = "POCO_F5"; // SSID
const char* password = "12344321"; // пароль
```

Рисунок 4.11 – Зміні коду SSID і пароля Wi-Fi

Після успішного підключення до мережі WiFi, система ініціалізує датчик DHT11 та LCD екран для відображення даних температури та вологості.

6. Відправка даних на ThingSpeak

Функція `sendToThingSpeak` відповідає за відправку даних температури та вологості на ThingSpeak. Вона формує HTTP запит з використанням API ключа та значень температури та вологості, отриманих з датчика DHT11. Ця функція викликається в циклі `loop()` кожні 5 секунд для постійного оновлення даних на ThingSpeak.

7. Візуалізація даних на ThingSpeak

Після успішної відправки даних на ThingSpeak, їх можна переглянути у вигляді графіків та діаграм на сторінці свого каналу, як показано на рисунку 4.12. ThingSpeak автоматично візуалізує дані в реальному часі, що дозволяє легко відстежувати зміни кліматичних показників.

На сторінці каналу буде відображено декілька вкладок, таких як "Private View", "Public View" та "Data Import/Export". Вкладка "Private View" доступна лише для власника каналу та показує всі поля даних, які зберігаються на каналі. Тут можна побачити графіки зміни значень для кожного поля даних окремо або в комбінації з іншими полями.

Вкладка "Public View" дозволяє переглядати та аналізувати дані публічно, без потреби входу в систему. Ця вкладка може бути корисною для спільного доступу до даних з іншими користувачами або для публічної демонстрації результатів моніторингу.

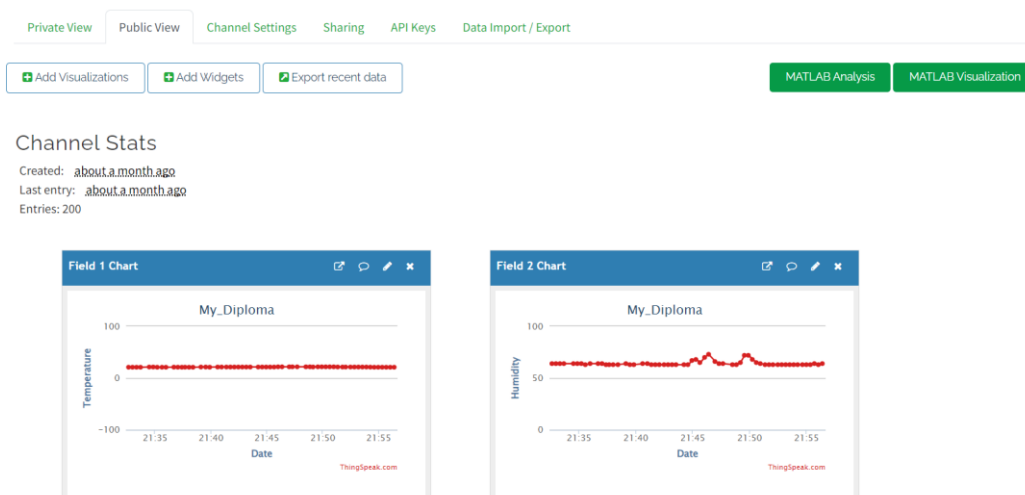


Рисунок 4.12 – Графіки даних в ThingSpeak

8. Веб-інтерфейс для перегляду даних

У коді реалізовано простий веб-інтерфейс за допомогою AsyncWebServer для відображення поточних значень температури та вологості. Функція SendHTML генерує HTML-сторінку з відображенням даних. Цей веб-інтерфейс можна відкрити, вказавши IP-адресу мікроконтролера ESP8266 у веб-браузері.

Веб-інтерфейс дозволяє не тільки відправляти дані на ThingSpeak, але й локально відображати їх на веб-сторінці, яка автоматично оновлюється кожні 200 мілісекунд за допомогою JavaScript. Робота сторінки показана на рисунку 4.13.

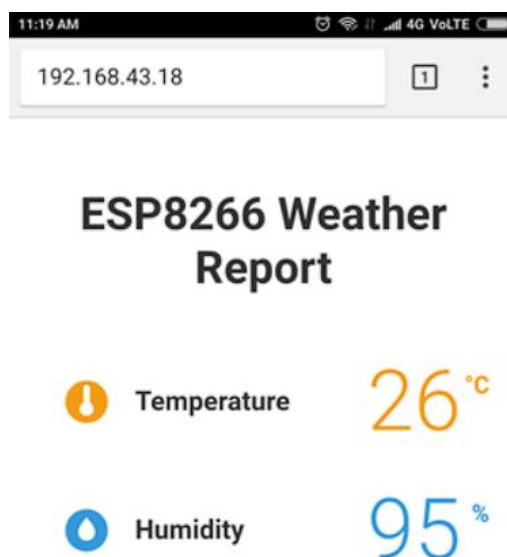


Рисунок 4.13 – Веб інтерфейс відображення даних з датчика

4.4 Висновки до розділу

У третьому розділі було вичерпно описано реалізацію системи дистанційного моніторингу кліматичних показників середовища. Ця реалізація включала успішну настройку апаратної частини, зокрема, NodeMCU ESP8266, яка використовувалась для зчитування даних від датчиків та їх передачі до віддаленого сервера.

Програмна частина була також успішно реалізована за допомогою середовищ Arduino IDE для ESP та лістингів коду, які демонстрували функціонал пристроїв. Крім того, було детально описано інтеграцію системи з хмарною платформою ThingSpeak для зберігання, візуалізації та аналізу зібраних кліматичних даних. Використання ThingSpeak забезпечило зручний та ефективний спосіб моніторингу показників навколишнього середовища в режимі реального часу.

Була проведена демонстрація роботи системи, що підтвердила її ефективність та практичну цінність. У результаті успішної реалізації апаратної та програмної частин, а також інтеграції з платформою ThingSpeak, забезпечено стабільне та ефективне функціонування системи моніторингу кліматичних показників. Це відкриває широкі можливості для використання розробленої системи у реальних умовах для збору, аналізу та візуалізації екологічних даних.

Таким чином, успішна реалізація проекту дистанційного моніторингу кліматичних показників середовища з використанням NodeMCU ESP8266, середовища Arduino IDE та інтеграції з хмарною платформою ThingSpeak забезпечує надійну та гнучку систему для відстеження стану навколишнього середовища.

РОЗДІЛ 5. РОЗРОБКА 3D МОДЕЛІ КОРПУСУ В СЕРЕДОВИЩІ SOLIDWORKS

5.1 Важливість Розробки Корпусу для Системи

Створення корпусу для системи моніторингу кліматичних показників середовища є важливою складовою проекту з огляду на кілька ключових аспектів. Насамперед, корпус забезпечує захист внутрішніх електронних компонентів, таких як плата ESP8266, датчик температури та вологості DHT11, дисплей LCD1602 та інші, від шкідливого впливу зовнішніх факторів, таких як пил, волога та механічні пошкодження. Захист цих компонентів, які є досить чутливими до навколишнього середовища, сприяє продовженню терміну служби системи. У середовищах з підвищеною вологістю або великою кількістю пилу корпус допомагає уникнути коротких замикань та інших несправностей, викликаних зовнішніми факторами.

По-друге, корпус покращує зручність використання пристрою та його естетичний вигляд. Він забезпечує зручне розташування елементів керування, таких як дисплей та кнопки, і надає системі привабливий зовнішній вигляд, що покращує її сприйняття користувачами. Зручне та логічне розташування всіх елементів полегшує взаємодію з пристроєм. Естетично привабливий дизайн також сприяє викликанню довіри та прихильності у користувачів, що є важливим для будь-якої технології, що взаємодіє з людьми.

Нарешті, компактний та функціональний корпус підвищує портативність та мобільність системи. Це розширює можливості її використання, полегшує транспортування та встановлення в різних місцях. Мобільність є важливою характеристикою сучасних пристроїв, особливо якщо вони призначені для використання в різних умовах. Легкість транспортування дозволяє використовувати систему на виставках, у польових умовах, на тимчасових робочих місцях та в інших місцях, де потрібно швидко розгортати та згортати обладнання.

Крім основних функцій, слід також врахувати додаткові аспекти розробки корпусу. Матеріали, з яких виготовлений корпус, повинні бути зносостійкими та витримувати тривалу експлуатацію. Важливо передбачити можливість легкого

доступу до внутрішніх компонентів для обслуговування та ремонту без потреби у повному розбиранні корпусу. Також необхідно врахувати ергономіку – зручність тримання, розміщення кнопок та екрану, а також простоту використання. Крім цього, важливо забезпечити наявність вентиляційних отворів для охолодження системи, що сприятиме ефективному відведенню тепла від внутрішніх компонентів і запобігатиме їх перегріву, забезпечуючи стабільну роботу пристрою.

Таким чином, розробка корпусу є невід'ємною частиною проекту, оскільки вона забезпечує захист електронних компонентів від зовнішніх факторів, покращує зручність використання та естетичний вигляд пристрою, а також сприяє його компактності та портативності. Врахування всіх зазначених аспектів дозволить створити надійну, зручну та привабливу систему моніторингу кліматичних показників, яка відповідатиме високим стандартам якості та задовольнятиме потреби користувачів.

5.2 Вимоги до розробки корпусу

Розробка корпусу для системи дистанційного моніторингу кліматичних показників середовища є важливим аспектом проекту з кількох ключових причин.

Габаритні розміри і форма корпусу:

- Корпус пристрою матиме точні розміри відповідно до вимог проекту: довжина 150 мм, ширина 100 мм, висота 75 мм. Ці параметри забезпечують достатньо місця для розміщення всіх необхідних компонентів, а також зручність монтажу та обслуговування пристрою.

- Форма корпусу буде прямокутною з прямими кутами. Така геометрія дозволить максимально ефективно використовувати внутрішній простір для оптимального розміщення всіх елементів системи.

- На лицьовій панелі корпусу передбачена виїмка та отвір для LCD дисплея 1602, а також кнопок поруч з ним. Це забезпечить зручний доступ до інформації на дисплеї та легке керування пристроєм. Розташування цих елементів сприятиме зручності користування пристроєм та полегшить навігацію користувача.

- Проект корпусу враховує всі необхідні аспекти в плані розмірів, форми та розташування елементів, щоб забезпечити оптимальну функціональність, зручність експлуатації та естетичний вигляд продукту.

Матеріал виготовлення:

- Матеріал корпусу є важливим елементом конструкції. Для досягнення оптимальної комбінації міцності, гнучкості та захисту внутрішніх компонентів від зовнішніх факторів було обрано пластик PETG (рисунок 5.1).



Рисунок 5.1 – Пластик PETG

- PETG – це високоякісний матеріал, відомий своєю міцністю та гнучкістю. Він ідеально підходить для виробництва корпусів електронних пристроїв завдяки можливості 3D-друку.

- Основною перевагою PETG є його здатність надійно захищати внутрішні компоненти від зовнішніх факторів, таких як волога, пил та механічні впливи. Це дозволяє збільшити надійність та тривалість служби пристрою, зменшуючи ризик пошкоджень та витрат на обслуговування.

- Використання PETG також забезпечує високу естетичну привабливість продукту. Матеріал доступний у різних кольорах і здатний зберігати свій зовнішній

вигляд протягом тривалого часу, що робить його ідеальним для створення сучасних та привабливих дизайнів корпусів.

- Загалом, використання пластику PETG для виготовлення корпусу є оптимальним рішенням, яке забезпечує баланс між міцністю, гнучкістю, захистом внутрішніх компонентів та естетичним виглядом продукту.

Додаткові вимоги:

- При розробці конструкції корпусу пріоритетом є забезпечення легкого доступу для обслуговування або заміни внутрішніх компонентів. Це включає створення системи, яка дозволить швидко та безпечно проводити ремонтні роботи або модифікації без використання спеціалізованого обладнання чи витрати великої кількості часу.

- Необхідно враховувати потребу в ефективному відведенні тепла від електронних елементів, щоб уникнути перегріву та зберегти їх оптимальну робочу температуру. Рекомендується розглянути можливість встановлення вентиляційних отворів або використання спеціальних систем охолодження для забезпечення надійного відведення тепла.

- Крім основних функцій, важливим аспектом є естетичний дизайн корпусу. Він повинен гармонійно вписуватися у загальний вигляд системи, бути привабливим та відповідати сучасним тенденціям дизайну. Це підвищить загальне сприйняття пристрою користувачами та сприятиме довірі до продукту.

- Міцність та стійкість корпусу до впливу зовнішніх факторів, таких як вологість, пил та механічні впливи, також є важливими вимогами. Це може вимагати використання відповідних матеріалів та технологій виробництва для забезпечення високої якості та довговічності продукту.

5.3 Етапи проектування 3D моделі

5.3.1 Загальне проектування корпусу

Процес створення 3D моделі корпусу для системи, що включає датчик DHT11, ESP8266 і дисплей LCD1602, в середовищі SolidWorks передбачає два

основні етапи: проектування корпусу та проектування нижньої кришки.

Проектування корпусу.

Проектування корпусу передбачає створення основної геометрії та всіх необхідних отворів для розміщення компонентів системи.

1. Створення базової форми корпусу

На першому етапі розробляється основна форма корпусу, що включає визначення габаритів і загальної конструкції. Цей корпус забезпечує розміщення всіх внутрішніх компонентів та їх захист.

2. Додавання отвору для роз'єму USB

На задній панелі корпусу створюється отвір для роз'єму USB, який забезпечує підключення живлення та даних до ESP8266. Отвір розташовується таким чином, щоб забезпечити легкий доступ до USB порту.

3. Додавання отвору для LCD1602

На передній панелі корпусу створюється отвір для LCD1602 дисплея. Отвір розташовується з урахуванням зручності перегляду інформації користувачем. Всі необхідні кріплення для дисплея проектуються в корпусі для забезпечення надійного фіксування.

4. Додавання отвору для датчика DHT11

На верхній частині корпусу створюється отвір для датчика DHT11. Отвір проектується таким чином, щоб забезпечити оптимальне функціонування датчика і точне зчитування даних про температуру та вологість.

5. Проектування кріплень для ESP8266

Всередині корпусу створюються місця для кріплення ESP8266. Розташування кріплень визначається для забезпечення зручного монтажу та доступу до компонентів під час технічного обслуговування.

Проектування нижньої кришки

Нижня кришка проектується з урахуванням необхідності забезпечення належної вентиляції та зручного доступу до компонентів всередині корпусу.

1. Створення основної форми нижньої кришки

Нижня кришка розробляється для надійного закриття корпусу та

забезпечення захисту внутрішніх компонентів. Форма кришки відповідає габаритам корпусу та забезпечує легке кріплення.

2. Визначення місць для кріплення ESP8266

На нижній кришці передбачені місця для кріплення ESP8266. Це дозволяє надійно фіксувати плату всередині корпусу та забезпечити її стабільне положення.

3. Створення отворів для вентиляції

На нижній кришці передбачені вентиляційні отвори для забезпечення ефективного охолодження компонентів. Вентиляційні отвори проектуються у вигляді прорізів або перфорацій, розташованих в оптимальних місцях для відведення тепла.

5.3.2 Проекції та деталі корпусу та нижньої кришки

Корпус:

1. Проекція зверху (рис. 5.2):

На цій проекції можна побачити верхню площину корпусу з отвором для датчика DHT11. Розташування отвору забезпечує оптимальне функціонування датчика.

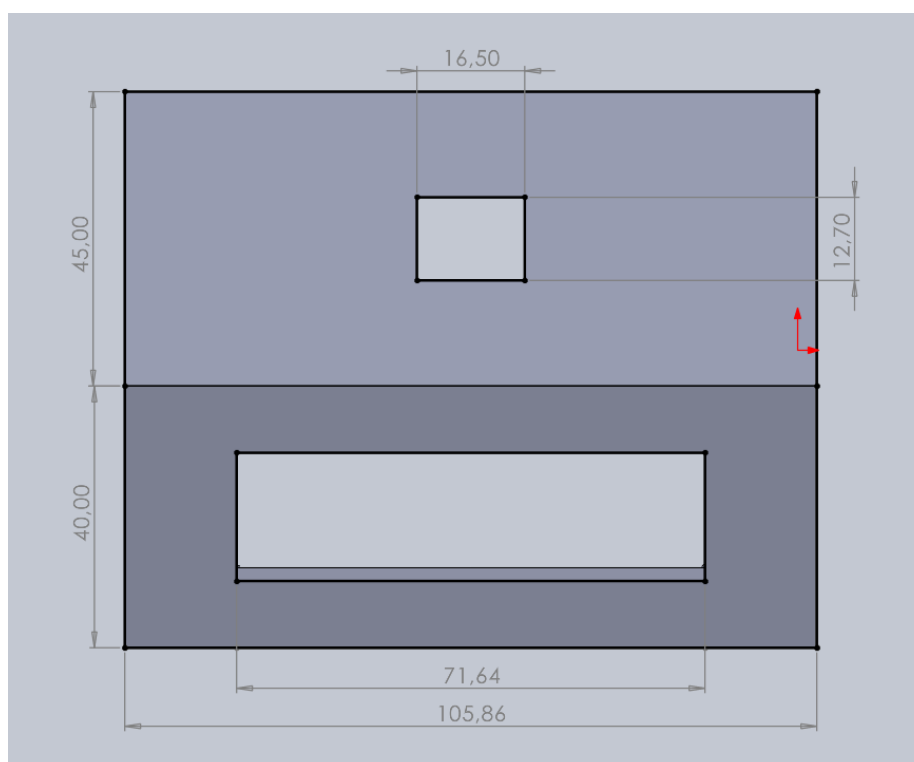


Рисунок 5.2 – Проекція корпусу зверху

2. Проекція збоку (рис. 5.3):

Профіль корпусу показує товщину та форму, включаючи отвори для кріплення ESP8266 та дисплея LCD1602. Всі елементи розташовані таким чином, щоб забезпечити зручний доступ для користувача.

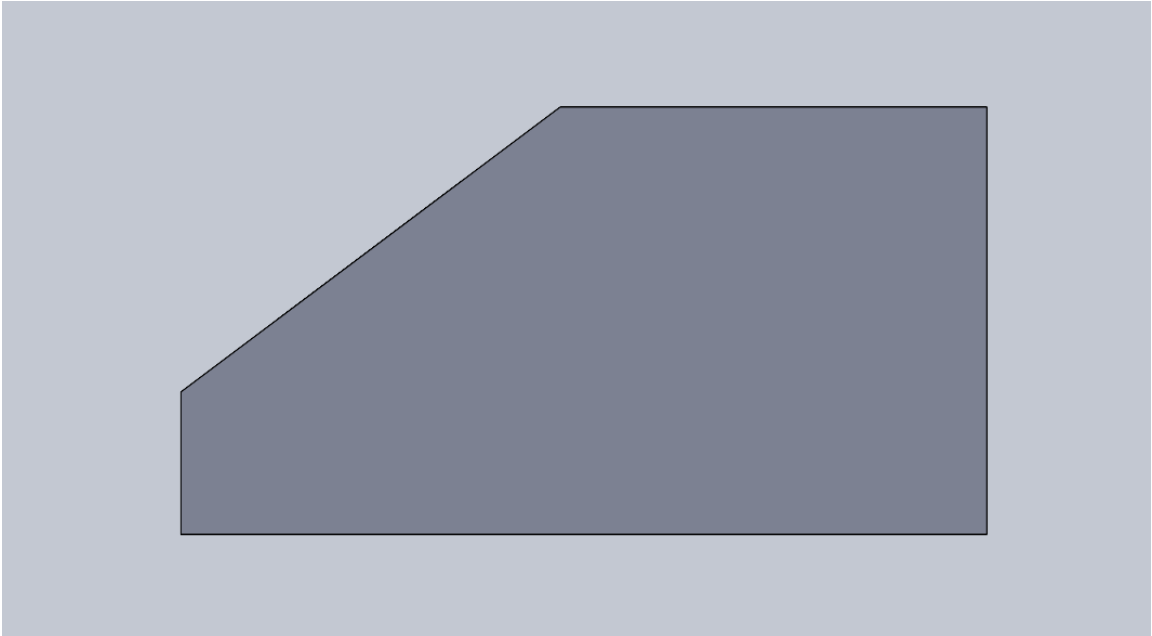


Рисунок 5.3 – Проекція корпусу збоку

3. Проекція спереду (рис. 5.4):

На цій проекції видно фронтальну панель корпусу з отвором для дисплея LCD1602. Отвір розташовано для забезпечення зручного перегляду даних користувачем.

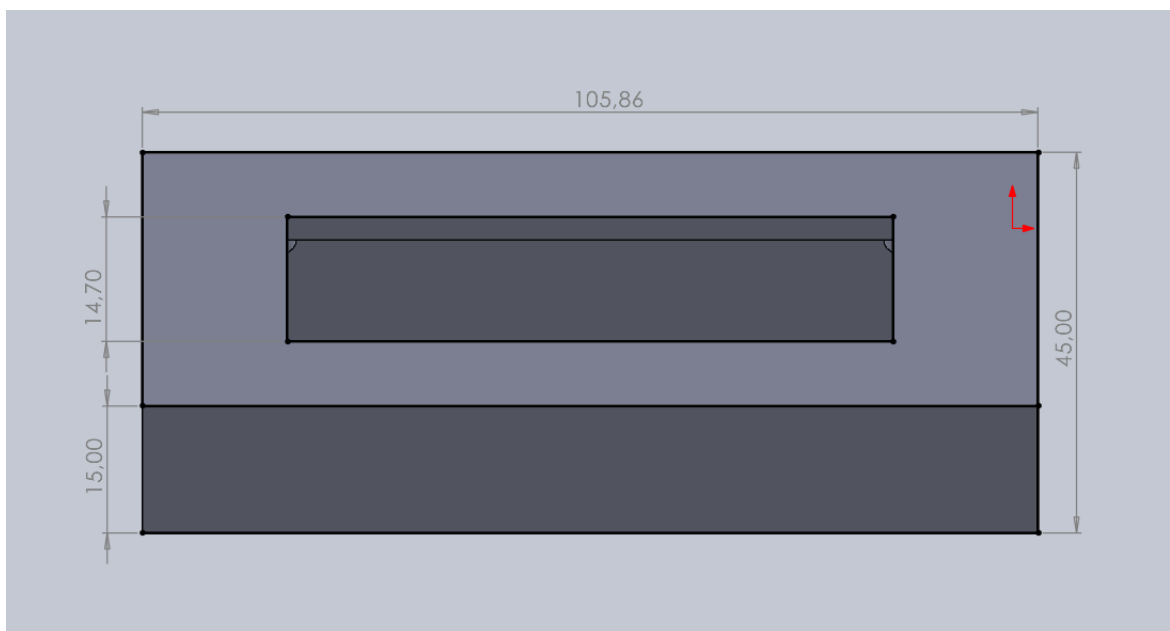


Рисунок 5.4 – Проекція корпусу спереду

Нижня кришка:

Проекція зверху (рис. 5.5):

На цій проекції відображено розташування кріплень для ESP8266 на внутрішній площині кришки. Кріплення забезпечують надійне фіксування плати.

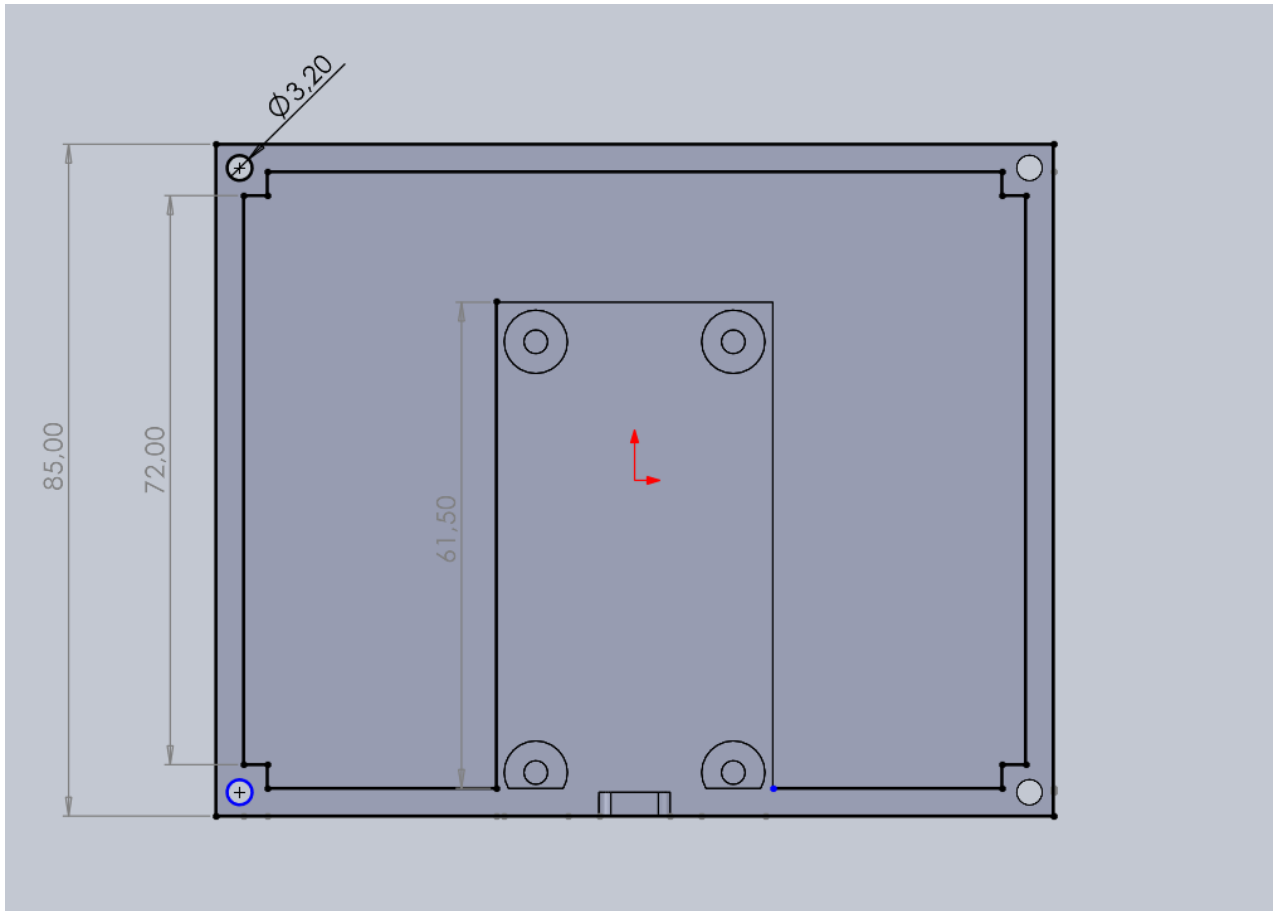


Рисунок 5.5 – Проекція кришки зверху

Проекція збоку (рис. 5.6):

Профіль нижньої кришки показує товщину та форму.

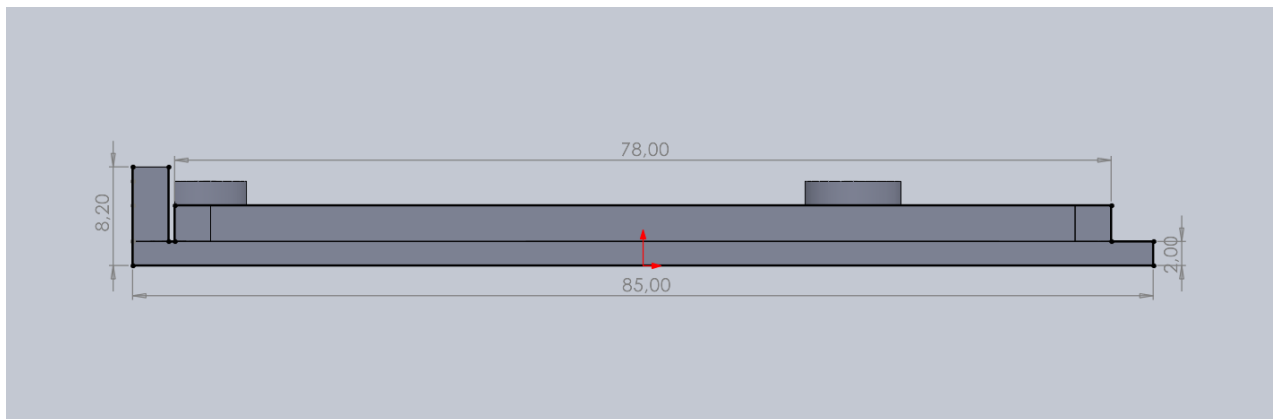


Рисунок 5.6 – Проекція кришки збоку

Проекція спереду (рис. 5.7):

На цій проекції видно фронтальну панель, її товщину і форму.

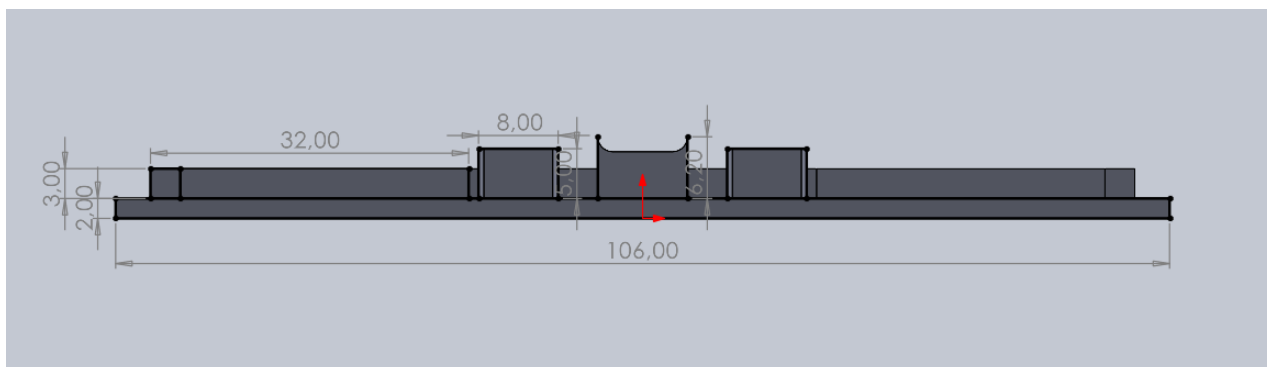


Рисунок 5.7 – Проекція кришки спереду

Ці етапи проектування забезпечують створення оптимального корпусу для системи, що включає датчик DHT11, ESP8266 і дисплей LCD1602. Точність і детальність моделювання в SolidWorks дозволяють створити конструкцію, яка забезпечить надійність, зручність у використанні та ефективну роботу всіх компонентів.

5.4 Підготовка до 3D друку корпусу

Експорт моделі у необхідний формат: Перш за все, у SolidWorks необхідно зберегти 3D модель корпусу у форматі STL або OBJ, забезпечуючи високу деталізацію та точність геометрії. Перед експортом слід перевірити модель на відсутність помилок та некоректних елементів, щоб уникнути проблем під час друку.

Налаштування параметрів для 3D друку: Для роботи в програмі Orca Slicer з принтером Creality K1 MAX необхідно вибрати пластик PETG в якості матеріалу для друку. Встановіть рекомендовані налаштування температури екструдера та платформи для друку PETG (рис. 5.).

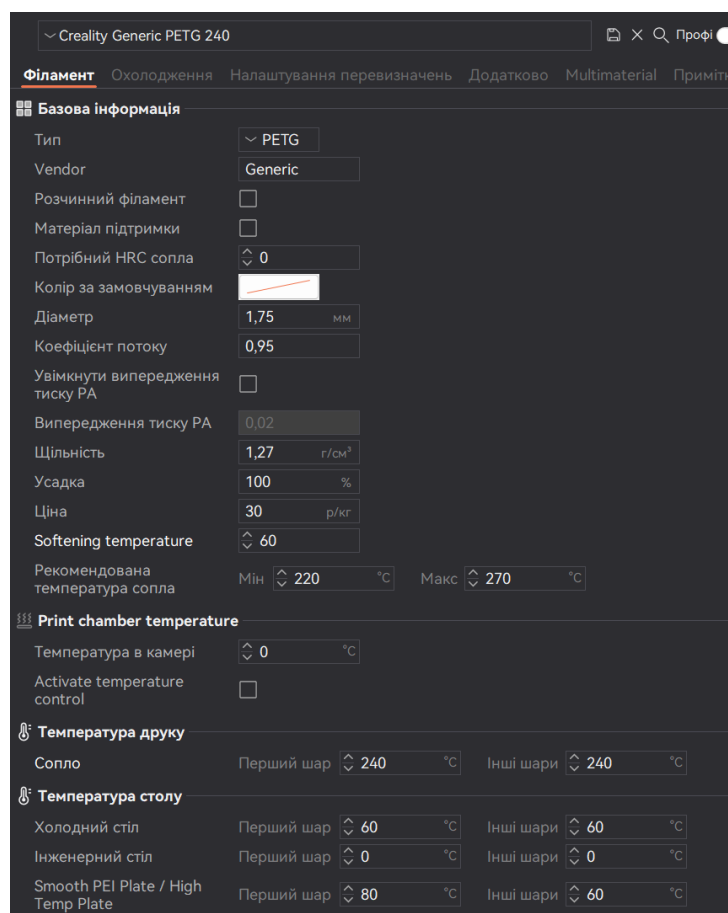


Рисунок 5. – рекомендовані налаштування температури екструдера та платформи для друку

Правильна орієнтація моделі на платформі та використання необхідних опор є важливим етапом у процесі 3D друку корпусу електронного пристрою. Орієнтація моделі має бути вибрана таким чином, щоб мінімізувати кількість необхідних опор і забезпечити оптимальну якість друку на важливих поверхнях. Наприклад, якщо модель містить великі горизонтальні виступи, їх слід орієнтувати так, щоб вони потребували мінімальної кількості підтримок. Це полегшить процес видалення підтримок після завершення друку і зменшить ризик пошкодження поверхні моделі. Крім того, правильна орієнтація може зменшити внутрішні напруження у матеріалі, що виникають під час охолодження, та запобігти деформаціям корпусу (рис. 5.14).

Щільність заповнення моделі також відіграє критичну роль у забезпеченні необхідної міцності корпусу. Щільність заповнення визначає, наскільки внутрішній простір моделі буде заповнений матеріалом під час друку. Висока

щільність заповнення (наприклад, 50% і більше) забезпечує велику міцність і стійкість до механічних навантажень, але збільшує витрату матеріалу і час друку. Натомість, низька щільність заповнення (20% або менше) зменшує витрати матеріалу і скорочує час друку, але може призвести до меншої міцності корпусу. Для корпусу електронного пристрою оптимальним може бути вибір середньої щільності заповнення, наприклад, 30-40%, що забезпечить достатню міцність для захисту внутрішніх компонентів, зберігаючи при цьому ефективність процесу друку. Важливо також враховувати тип структурного заповнення, наприклад, сітчасте або стільникове, що може додатково підвищити міцність та стійкість конструкції без значного збільшення ваги та витрат матеріалу (рис. 5.15).

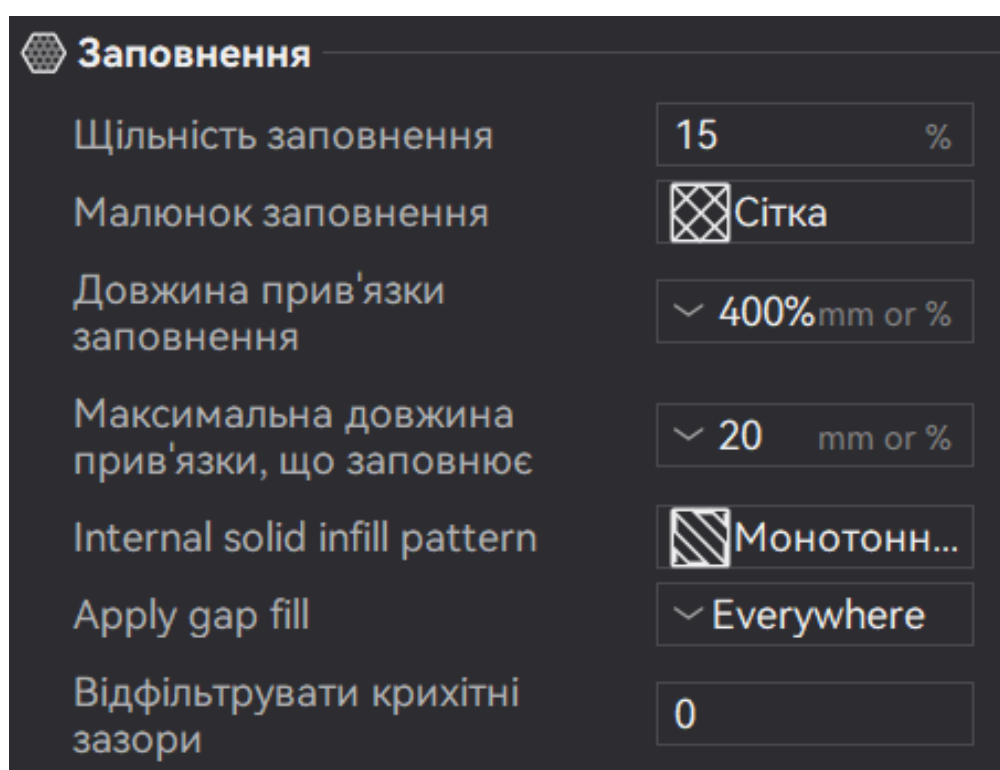


Рисунок 5. – Вікно вибору щільності заповнення моделі

Вибір товщини шару друку є важливим аспектом при виготовленні корпусу електронного пристрою за допомогою 3D друку. Висока товщина шару (наприклад, 0.3 мм або більше) дозволяє зберегти час друку, проте може призвести до меншої деталізації та неоднаковості поверхні. У цьому випадку кінцевий продукт може мати відчуття шерохватості на дотик та виглядати менш привабливо. Навпаки, тонкий шар друку (наприклад, 0.1 мм або менше) дозволяє досягти високої

деталізації та гладкої поверхні, але збільшує час друку. Варто враховувати, що тонкі шари можуть бути менш міцними та більш схильними до деформації. Оптимальний вибір товщини шару друку зазвичай залежить від конкретних потреб проекту та вимог до якості кінцевого виробу. Для корпусу електронного пристрою може бути рекомендовано використання середньої товщини шару, наприклад, 0.2 мм, яка забезпечить баланс між деталізацією та міцністю, зберігаючи при цьому прийнятний час друку (рис. 5.16).

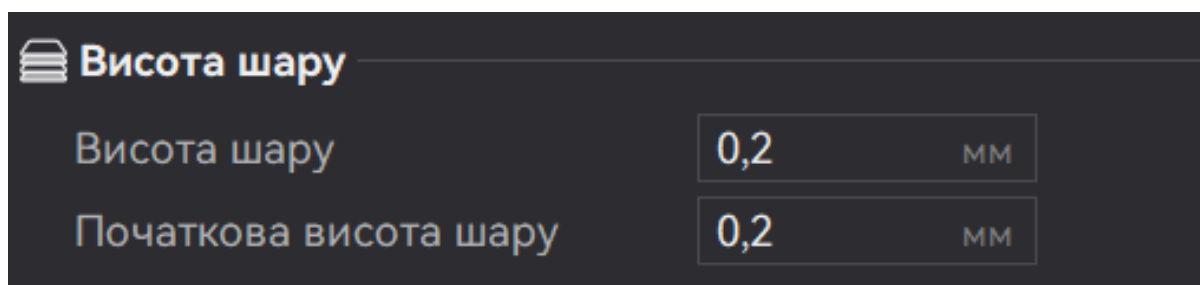


Рисунок 5. – Вікно вибору товщини шару

Вибір швидкості друку відповідно до вимог якості: Висока швидкість друку дозволяє значно скоротити час виготовлення моделі, що є важливим фактором при виробництві великих партій або при потребі в швидкому прототипуванні. Однак, збільшення швидкості друку часто призводить до зниження якості поверхні, з'являються видимі шари, нерівності та можливі дефекти, такі як пропуски матеріалу чи нерівномірне заповнення. Крім того, при високій швидкості друку може знизитися адгезія між шарами, що впливає на загальну міцність виробу. Низька швидкість друку, навпаки, забезпечує високу якість поверхні, точність деталей і кращу адгезію між шарами. Це особливо важливо для корпусів електронних пристроїв, де естетичний вигляд та точність розмірів мають вирішальне значення. Деталі, виготовлені на низькій швидкості, мають гладку поверхню, відсутність видимих шарів і високу міцність завдяки кращій зчепленості шарів. Однак, цей підхід значно збільшує час друку, що може бути неприйнятним у випадках, коли важлива швидкість виготовлення. Оптимальний вибір швидкості друку повинен враховувати баланс між якістю та часом виготовлення. Для корпусів електронних пристроїв рекомендується використовувати середню швидкість друку, що дозволяє досягти достатньої якості поверхні та точності деталей, при

цьому зберігаючи прийнятний час виготовлення. Наприклад, швидкість друку в межах 40-60 мм/с може бути оптимальною для більшості проектів, забезпечуючи гарний баланс між якістю та ефективністю. Крім того, важливо враховувати особливості матеріалу, що використовується для друку, оскільки деякі матеріали можуть вимагати певної швидкості для досягнення найкращих результатів. Наприклад, PETG, який часто використовується для виготовлення корпусів, зазвичай друкується на середній швидкості для забезпечення гарної якості та міцності (рис. 5.17).

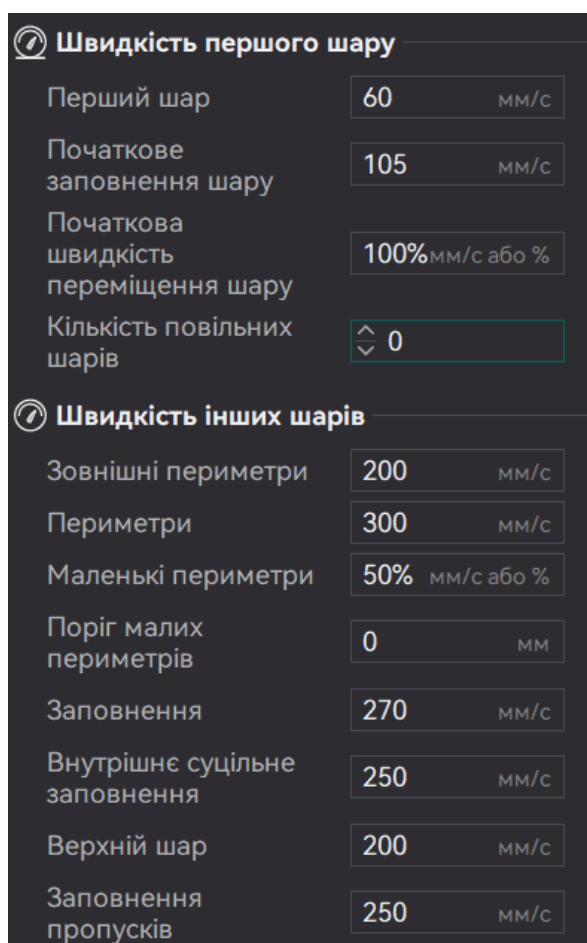


Рисунок 5. – Вікно вибору швидкості друку

Підготовка файлів для 3D принтера Creality K1 MAX: Після завершення налаштувань, необхідно імпортувати експортований STL або OBJ файл в Orca Slicer. Далі генерується G-код для управління рухами 3D принтера. Важливо перевірити відповідність налаштувань друку вимогам до корпусу, після чого зберегти підготовлений G-код на карту пам'яті для подальшого друку.



Рисунок 5. – Принтер CREALITY K1 MAX

Запуск друку корпусу на 3D принтері Creality K1 MAX (рис. 5.) проводиться, використовуючи згенерований G-код та налаштування в Orca Slicer. Час друку залежатиме від розміру моделі та вибраних параметрів друку. По завершенню друку можливо буде потрібна додаткова обробка, як видалення опор чи шліфування поверхні.

5.5 Висновки до розділу

У цьому розділі було детально розглянуто процес проектування та підготовки 3D моделі корпусу для системи моніторингу кліматичних показників у середовищі SolidWorks. Корпус забезпечує захист внутрішніх компонентів, поліпшує зручність використання, естетичний вигляд та портативність пристрою. Було визначено вимоги до розмірів, форми, матеріалу (PETG) та додаткових елементів корпусу. Описано етапи проектування корпусу та нижньої кришки, включаючи створення необхідних отворів та кріплень для компонентів. Наведено проєкції та деталі конструкції. Також розглянуто підготовку моделі до 3D друку, налаштування параметрів друку (температури, орієнтації, заповнення, товщини шару, швидкості) та експорт файлів для принтера Creality K1 MAX.

РОЗДІЛ 6. ВПРОВАДЖЕННЯ СИСТЕМИ ДИСТАНЦІЙНОГО МОНІТОРИНГУ МІКРОКЛІМАТУ В ЕКСПЛУАТАЦІЮ ТА ПЛАН ПО НАЛАГОДЖЕННЮ РОБОТИ

Розроблена система дистанційного моніторингу кліматичних показників середовища представляє собою комплексне рішення для збору, зберігання та відображення даних мікроклімату. Вона включає в себе пристрій для вимірювання та дистанційної передачі показників, базу даних на віддаленому сервері та веб-сайт для візуалізації даних.

Основні функції системи включають:

1. Збір даних: Програмований пристрій здійснює вимірювання кліматичних показників за допомогою датчиків та передає ці дані на віддалений сервер.
2. Зберігання даних: Отримані дані зберігаються в базі даних на віддаленому сервері для подальшого аналізу та використання.
3. Відображення даних: Спеціальний веб-сайт надає можливість перегляду поточних показників мікроклімату та історії вимірювань у формі таблиць і графіків. Користувачі можуть фільтрувати дані за часовим проміжком та виконувати аналіз.

Ця система є потужним інструментом для моніторингу кліматичних умов у реальному часі та надає користувачам зручний доступ до важливих даних через Інтернет. Вона може бути використана в різних сферах, включаючи сільське господарство, науку, промисловість та домашнє використання.

6.1 Інструкція з використання пристрою вимірювання

Для функціонування пристрою потрібно під'єднати кабель живлення USB-C – USB Type A. При поданні живлення плати будуть сигналізувати свою роботу за допомогою розміщених на них діодів. Логіка сигналів діодів наступна:

Синій діод плати esp8266:

1. Швидкий одиночний блим – плата запустилась/перезапустилась.
2. Постійне світіння – локальний сервер запущено.

3. Два швидких блима – успішне підключення до бази даних.
4. Постійне блимання з довгими паузами – не вдалось підключитись до бази даних.

Фото увімкненого пристрою показано на рисунку 6.1.

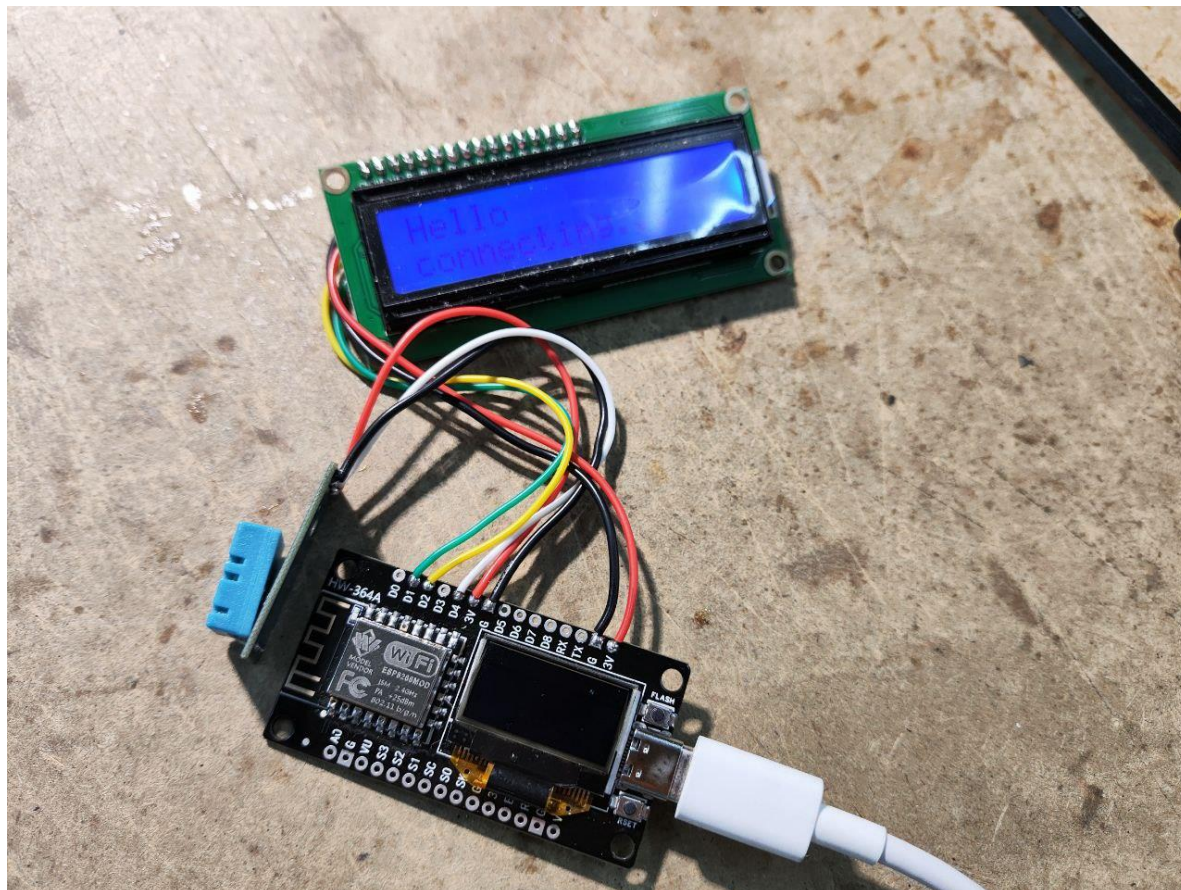


Рисунок 6.1 - Фото увімкненого пристрою

Програма пристрою передбачає стандартний режим роботи, що можна активувати за допомогою кнопки на платі. У цьому режимі пристрій автоматично підключається до заданої раніше точки Wi-Fi та розпочинає процес вимірювання і передачі даних до віддаленої бази даних. У разі успішного підключення, пристрій сигналізує двома швидкими блиманнями синього світлодіода, а в разі помилки діод почне постійно блимати з довгими паузами.

6.2 Інструкція по налагодженню роботи і виправленню помилок

Хоча розроблена система дистанційного моніторингу кліматичних показників середовища загалом є надійною, можливі технічні несправності можуть негативно вплинути на її функціонування. У цьому підрозділі розглядаються типові

проблеми, що можуть виникнути під час експлуатації системи, та надаються рекомендації щодо їх усунення. Крім того, представлено загальні методи налагодження для забезпечення стабільної роботи системи.

1. Проблеми з модулем ESP8266

Можливі поломки:

- Відсутність з'єднання з Wi-Fi. Модуль не може підключитися до мережі Wi-Fi.

- Нестабільне підключення. Постійні розриви з'єднання з мережею.

Вирішення:

- Перевірка налаштувань Wi-Fi: Переконайтеся, що SSID і пароль введені правильно.

- Перевірка рівня сигналу: Розташувати модуль ближче до маршрутизатора або використовувати підсилювач сигналу.

- Оновлення прошивки: Перевірити наявність оновлень для модуля ESP8266 і прошити нову версію.

2. Проблеми з датчиком DHT11

Можливі поломки:

- Неточні показники: Датчик видає некоректні дані.

- Відсутність показників: Датчик не передає дані на мікроконтролер.

Вирішення:

- Перевірка з'єднань: Переконайтеся, що всі з'єднання між датчиком та мікроконтролером міцні та коректні.

- Тестування датчика: Використати інший датчик для перевірки, чи проблема не в самому DHT11.

- Очищення датчика: Очистити датчик від можливого забруднення.

3. Проблеми з дисплеєм LCD1602

Можливі поломки:

- Немає зображення: Дисплей не показує жодної інформації.

- Неправильне відображення: Дані відображаються некоректно або частково.

Вирішення:

- Регулювання контрасту: Налаштувати потенціометр для регулювання контрасту дисплея.
- Перевірка з'єднань: Переконайтеся, що всі з'єднання між дисплеєм та мікроконтролером правильні.
- Оновлення коду: Перевірити правильність коду, що відповідає за роботу з дисплеєм.

4. Проблеми з програмним забезпеченням в Arduino IDE

Можливі поломки:

- Помилки компіляції: Код не компілюється через помилки.
- Неправильне виконання: Код виконується некоректно або система зависає.

Вирішення:

- Перевірка коду: Переглянути код на наявність синтаксичних помилок або помилок логіки.
- Оновлення бібліотек: Переконайтеся, що всі використовувані бібліотеки актуальні та сумісні.
- Налаштування: Використати серійний монітор для відстеження виконання коду та виявлення помилок.

5. Проблеми з сервісом ThingSpeak

Можливі поломки:

- Відсутність даних: Дані не відправляються на платформу ThingSpeak.
- Неправильне відображення: Дані відображаються некоректно або з затримками.

Вирішення:

- Перевірка API ключів: Переконайтеся, що API ключі правильно введені в код.

- Моніторинг статусу сервера: Перевірити статус сервера ThingSpeak на наявність технічних робіт або збоїв.
- Тестування відправки даних: Використати тестові запити для перевірки роботи API.

Загальні кроки щодо налагодження роботи системи

- Тестування кожного компонента окремо: Перевірити роботу кожного компоненту системи окремо, щоб виявити можливі проблеми на ранньому етапі.
- Систематичний підхід до налагодження: Визначити порядок перевірки компонентів, починаючи з апаратної частини і закінчуючи програмною.
- Використання інструментів для налагодження: Застосувати серійний монітор та інші інструменти для діагностики та виявлення проблем.
- Постійне оновлення знань: Слідкувати за новими версіями прошивок, бібліотек та документації для своєчасного оновлення системи.
- Документація процесу: Вести детальний журнал проведених налагоджень та змін для полегшення майбутніх ремонтів та покращень.

6.3 Можливості покращення системи

Хоча розроблена система дистанційного моніторингу виконує свої завдання, існують можливості для подальшого покращення, що може значно підвищити її ефективність та корисність. Ось деякі з них:

1. Розширення функціоналу датчиків:

Заміна поточних датчиків на більш точні та розширення набору вимірюваних параметрів дозволить системі ефективно моніторити кліматичні умови в більш широкому діапазоні середовищ. Наприклад, додавання датчиків рівня CO₂ або альтернативних джерел освітлення.

2. Зміна типу живлення:

Додавання можливості живлення системи від батарей або акумуляторів підвищить її мобільність та незалежність від стаціонарних джерел електроживлення. Це може бути особливо корисно для використання системи в

місцях без постійного доступу до електромережі, таких як віддалені ферми або природні резервати.

3. Покращення інтерфейсу користувача:

Розширення можливостей візуалізації та аналізу даних на веб-сайті дозволить забезпечити користувачам більш гнучкий та зручний доступ до інформації, що вимірюється системою. Додавання функцій експорту даних та додаткових фільтрів полегшить аналіз та використання інформації. Наприклад, можливість вибору періоду відображення даних або налаштування сповіщень про надходження певних показників.

Загалом, виконання цих покращень сприятиме розширенню можливостей та ефективності системи дистанційного моніторингу кліматичних показників середовища, забезпечуючи її більш широке застосування та корисність.

6.4 Висновки до розділу

У четвертому розділі детально розглядалися ключові аспекти впровадження системи дистанційного моніторингу мікроклімату в експлуатацію. Вже готова система здатна вимірювати показники клімату та передавати їх на віддалений сервер через підключення до точки Wi-Fi, що має доступ до Інтернету. Зібрані дані доступні для перегляду на сайті монітора через глобальну мережу Інтернет. Їх відображення представлене у вигляді таблиць та графіків, а також можливо фільтрувати за часовим діапазоном вимірів та експортувати у форматі CSV.

У цьому розділі також міститься докладна інструкція з використання пристрою вимірювання, що дозволяє користувачеві ефективно використовувати систему для моніторингу кліматичних показників у реальному часі та отримувати детальну інформацію про минулі заміри.

Додатково, розділ включає розгляд можливостей для подальшого удосконалення системи. Враховуючи отримані результати, можна розглядати такі покращення, як розширення функціоналу датчиків, додавання альтернативного джерела живлення та розширення функціоналу веб-сайту монітора.

ВИСНОВКИ

У даній роботі проведено аналіз та порівняння пристроїв для моніторингу кліматичних показників середовища, а також розроблено систему дистанційного моніторингу на основі пристроїв Arduino UNO. Порівняння різних пристроїв дозволило обґрунтувати вибір апаратних засобів для системи.

У розділі створено структурну та функціональну схеми системи моніторингу клімату, що візуалізують роботу та взаємозв'язок її компонентів. Реалізація системи на базі Arduino UNO була обґрунтована з точки зору доступності, вартості та функціональності.

Детально описано процес розробки апаратної та програмної частин системи, що дозволило забезпечити необхідний зв'язок з датчиками та передачу даних на сервер для подальшого аналізу та відображення.

Під час демонстрації роботи системи було показано її ефективність та можливість, зокрема збір та відображення кліматичних даних у реальному часі через Інтернет. Також надано інструкцію з використання пристрою вимірювання, що дозволить користувачам ефективно використовувати систему.

У розділі також висвітлено можливості подальшого удосконалення системи, такі як розширення функціоналу та підвищення точності вимірювань. Впровадження системи дистанційного моніторингу кліматичних показників на базі Arduino UNO дозволяє використовувати її у різних галузях для забезпечення контролю та аналізу кліматичних умов середовища.

Загалом, ця робота сприяє розвитку наукових досліджень у галузі моніторингу кліматичних умов та пропонує практичні рішення для створення ефективних систем дистанційного моніторингу. Результати досліджень та розробок можуть бути використані для подальшого розвитку систем моніторингу в різних галузях, де необхідний контроль та аналіз кліматичних умов.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Сміт, Дж., & Джонсон, Р. (2020). "ІоТ-базована система моніторингу навколишнього середовища для смарт-сільського господарства." Міжнародний журнал дослідження навколишнього середовища та громадського здоров'я, 17(3), 932.
2. Клименко, М. О., Прищепа, А. М., & Вознюк, Н. М. (2014). Моніторинг довкілля: підручник. Видавничий центр НАУ.
3. Джиган, В. І., & Жарінова, В. Г. (2016). Моніторинг довкілля: підручник. Національний університет біоресурсів і природокористування України.
4. Арутюнян, В. О., Григор'єв, В. А., & Лихоман, А. В. (2018). Моніторинг довкілля: навчальний посібник. Харківський національний університет міського господарства імені О. М. Бекетова.
5. Мельник, Л. Г., & Єгоров, В. В. (2016). Моніторинг довкілля: навчальний посібник. Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського".
6. Буренніков, Ю. Б., & Журавель, Є. В. (2016). Моніторинг довкілля: навчальний посібник. Видавництво ХНУ імені В.Н. Каразіна.
7. Касіянчук, В. Д., & Гунчак, В. М. (2017). Системи екологічного моніторингу: навчальний посібник. Видавництво Львівської політехніки.
8. Філатов, В. І., & Бойко, Т. В. (2015). Моніторинг довкілля: навчальний посібник. Чорноморський державний університет імені Петра Могили.
9. Вержбицький, П. П., & Лаврик, Г. І. (2019). Моніторинг довкілля: методичні вказівки до виконання практичних робіт. Національний університет водного господарства та природокористування.
10. Nest Learning Thermostat: [Електронний ресурс]. – URL: https://en.wikipedia.org/wiki/Nest_Thermostat
11. Weave protocol: [Електронний ресурс]. – URL: [https://en.wikipedia.org/wiki/Weave_\(protocol\)](https://en.wikipedia.org/wiki/Weave_(protocol))
12. Ecobee Smart Thermostat: [Електронний ресурс]. – URL: <https://www.ecobee.com/en-ca/smart-thermostats/>

13. Netatmo Weather Station: [Електронний ресурс]. – URL: <https://www.netatmo.com/smart-weather-station>
14. Ppm (particles per million): [Електронний ресурс]. – URL: https://groups.molbiosci.northwestern.edu/holmgren/Glossary/Definitions/Def-P/parts_per_million.html
15. Airthings Wave Plus: [Електронний ресурс]. – URL: <https://www.airthings.com/wave-plus>
16. Foobot: [Електронний ресурс]. – URL: <https://foobot.io/technology/>
17. VOC level: [Електронний ресурс]. – URL: <https://www.health.state.mn.us/communities/environment/air/toxins/voc>
18. Гупта, С., Жайн, А., & Мішра, П. (2019). "Проектування та впровадження IoT-базованої системи моніторингу та управління навколишнім середовищем." Журнал електротехніки та технології, 14(4), 1503-1511.
19. NodeMCU esp8266: [Електронний ресурс]. – URL: <https://nodemcu.readthedocs.io/en/release/>
20. DHT11: [Електронний ресурс]. – URL: <https://www.mouser.com/datasheet/2/758/DHT11-Technical-DataSheet-Translated-Version-1143054.pdf>
21. Arduino IDE: [Електронний ресурс]. – URL: <https://www.arduino.cc/en/software>
22. ThingSpeak: [Електронний ресурс]. – URL: https://thingspeak.com/pages/learn_more

Лістинг коду дизайну web сторінки

```
String SendHTML(float TempCstat,float Humiditystat)
{
String ptr = "<!DOCTYPE html> <html>\n";
ptr += "<head><meta name=\"viewport\" content=\"width=device-width, initial-scale=1.0, user-scalable=no\">\n";
ptr += "<link href=\"https://fonts.googleapis.com/css?family=Open+Sans:300,400,600\" rel=\"stylesheet\">\n";
ptr += "<title>ESP8266 Weather Report</title>\n";
ptr += "<style>html { font-family: 'Open Sans', sans-serif; display: block; margin: 0px auto; text-align: center;color: #333333;}\n";
ptr += "body{margin-top: 50px;}\n";
ptr += "h1 {margin: 50px auto 30px;}\n";
ptr += ".side-by-side{display: inline-block;vertical-align: middle;position: relative;}\n";
ptr += ".humidity-icon{background-color: #3498db;width: 30px;height: 30px;border-radius: 50%;line-height: 36px;}\n";
ptr += ".humidity-text{font-weight: 600;padding-left: 15px;font-size: 19px;width: 160px;text-align: left;}\n";
ptr += ".humidity{font-weight: 300;font-size: 60px;color: #3498db;}\n";
ptr += ".temperature-icon{background-color: #f39c12;width: 30px;height: 30px;border-radius: 50%;line-height: 40px;}\n";
ptr += ".temperature-text{font-weight: 600;padding-left: 15px;font-size: 19px;width: 160px;text-align: left;}\n";
ptr += ".temperature{font-weight: 300;font-size: 60px;color: #f39c12;}\n";
ptr += ".superscript{font-size: 17px;font-weight: 600;position: absolute;right: -20px;top: 15px;}\n";
ptr += ".data{padding: 10px;}\n";
ptr += "</style>\n";
ptr += "<script>\n";
ptr += "setInterval(loadDoc,200);\n";
ptr += "function loadDoc() {\n";
ptr += "var xhttp = new XMLHttpRequest();\n";
ptr += "xhttp.onreadystatechange = function() {\n";
ptr += "if (this.readyState == 4 && this.status == 200) {\n";
ptr += "document.getElementById(\"webpage\").innerHTML =this.responseText;\n";
ptr += "};\n";
ptr += "xhttp.open(\"GET\", \"/\", true);\n";
ptr += "xhttp.send();\n";
ptr += "};\n";
ptr += "</script>\n";
ptr += "</head>\n";
ptr += "<body>\n";

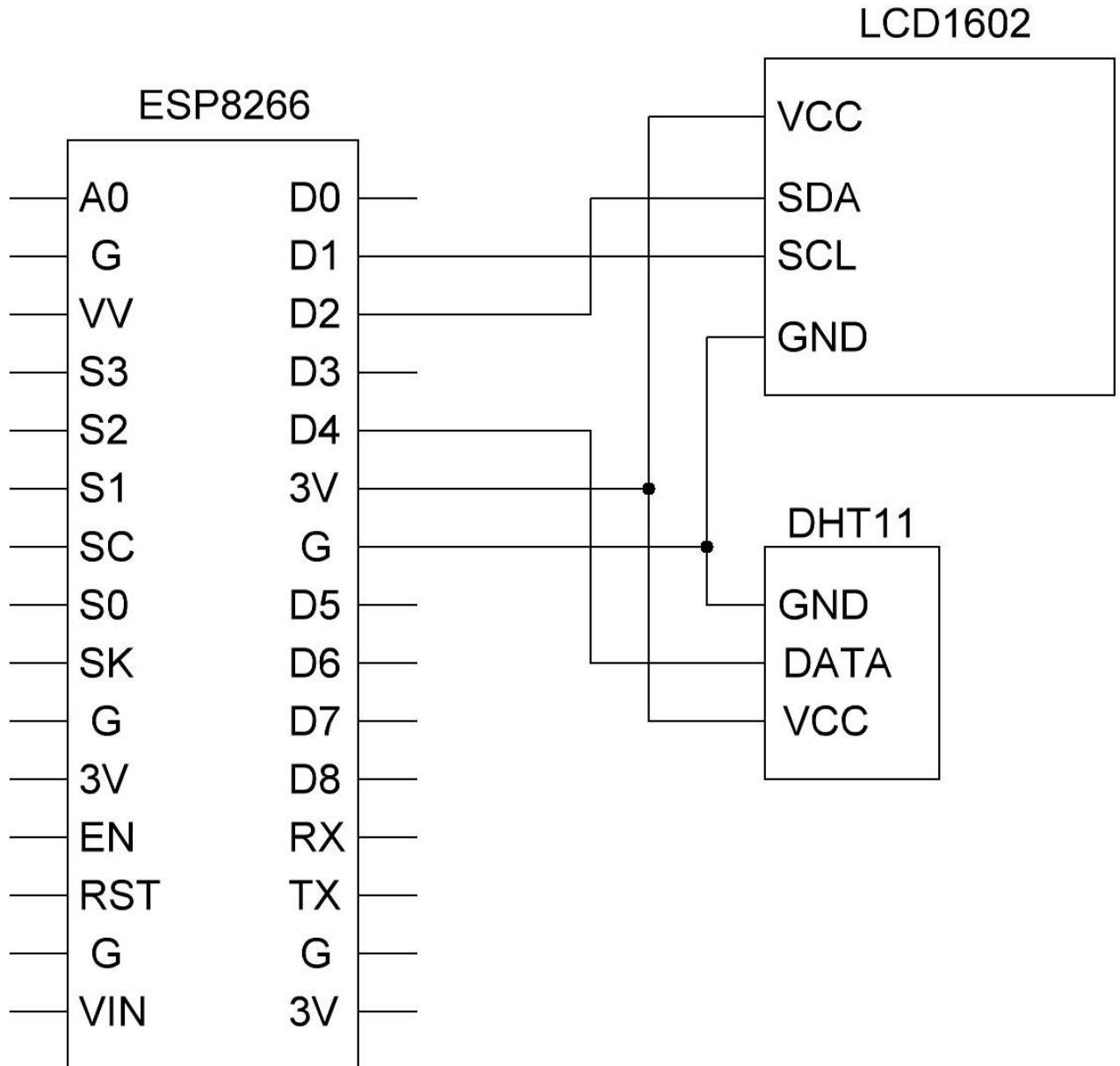
ptr += "<div id=\"webpage\">\n";

ptr += "<h1>ESP8266 Weather Report</h1>\n";
ptr += "<div class=\"data\">\n";
ptr += "<div class=\"side-by-side temperature-icon\">\n";
ptr += "<svg version=\"1.1\" id=\"Layer_1\" xmlns=\"http://www.w3.org/2000/svg\" xmlns:xlink=\"http://www.w3.org/1999/xlink\" x=\"0px\" y=\"0px\">\n";
ptr += "width=\"9.915px\" height=\"22px\" viewBox=\"0 0 9.915 22\" enable-background=\"new 0 0 9.915 22\" xml:space=\"preserve\">\n";
ptr += "<path fill=\"#FFFFFF\" d=\"M3.498,0.53c0.377-0.331,0.877-0.501,1.374-0.527C5.697-0.04,6.522,0.421,6.924,1.142\n";
ptr += "c0.237,0.399,0.315,0.871,0.311,1.33C7.229,5.856,7.245,9.24,7.227,12.625c1.019,0.539,1.855,1.424,2.301,2.491\n";
ptr += "c0.491,1.163,0.518,2.514,0.062,3.693c-0.414,1.102-1.24,2.038-2.276,2.594c-1.056,0.583-2.331,0.743-3.501,0.463\n";
ptr += "c-1.417-0.323-2.659-1.314-3.3-2.617C0.014,18.26-0.115,17.104,0.1,16.022c0.296-1.443,1.274-2.717,2.58-3.394\n";
ptr += "c0.013-3.44,0-6.881,0.007-10.322C2.674,1.634,2.974,0.955,3.498,0.53z\"/>\n";
ptr += "</svg>\n";
ptr += "</div>\n";
ptr += "<div class=\"side-by-side temperature-text\">Temperature</div>\n";
ptr += "<div class=\"side-by-side temperature\">\n";
ptr += (int)TempCstat;\n";
ptr += "<span class=\"superscript\">C</span></div>\n";
ptr += "</div>\n";
ptr += "<div class=\"data\">\n";
ptr += "<div class=\"side-by-side humidity-icon\">\n";
ptr += "<svg version=\"1.1\" id=\"Layer_2\" xmlns=\"http://www.w3.org/2000/svg\" xmlns:xlink=\"http://www.w3.org/1999/xlink\" x=\"0px\" y=\"0px\">\n";
ptr += "width=\"12px\" height=\"17.955px\" viewBox=\"0 0 13 17.955\" enable-background=\"new 0 0 13 17.955\" xml:space=\"preserve\">\n";
ptr += "<path fill=\"#FFFFFF\" d=\"M1.819,6.217C3.139,4.064,6.5,0.6,6.5,0.6s3.363,4.064,4.681,6.217c1.793,2.926,2.133,5.05,1.571,7.057\n";
ptr += "c-0.438,1.574-2.264,4.681-6.252,4.681c-3.988,0-5.813-3.107-6.252-4.681c-0.313,11.267,0.026,9.143,1.819,6.217\"/></path>\n";
ptr += "</svg>\n";
ptr += "</div>\n";
ptr += "<div class=\"side-by-side humidity-text\">Humidity</div>\n";
ptr += "<div class=\"side-by-side humidity\">\n";
ptr += (int)Humiditystat;\n";
ptr += "<span class=\"superscript\">%</span></div>\n";
ptr += "</div>\n";

ptr += "</div>\n";
ptr += "</body>\n";
ptr += "</html>\n";
return ptr;
}
```

ДОДАТОК Б

Принципова схема системи дистанційного моніторингу кліматичних показників середовища



ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

AMS	Advanced Microcontroller Systems	Сучасні системи мікроконтролерів
API	Application Programming Interface	Інтерфейс програмування застосунків
CSV	Comma-Separated Values	Значення, розділені комами
DHT	Digital Humidity and Temperature	Цифровий датчик вологості та температури
ESP	Espressif Systems	Виробник мікроконтролерів серії ESP
GND	Ground	Заземлення
HTTP	HyperText Transfer Protocol	Протокол передачі гіпертексту
I2C	Inter-Integrated Circuit	Міжінтегральний інтерфейс
IDE	Integrated Development Environment	Інтегроване середовище розробки
LCD	Liquid Crystal Display	Рідкокристалічний дисплей
LED	Light Emitting Diode	Світлодіод
MQTT	Message Queuing Telemetry Transport	Протокол телеметрії черги повідомлень
RS	Register Select	Вибір регістру
RW	Read/Write	Читання/Запис
USB	Universal Serial Bus	Універсальна послідовна шина
VCC	Voltage Common Collector	Живлення
VOC	Volatile Organic Compounds	Леткі органічні сполуки
Wi-Fi	Wireless Fidelity	Бездротова мережа високої точності

АНОТАЦІЯ

У дипломній роботі розглядається питання створення системи дистанційного моніторингу кліматичних показників середовища. Спочатку проводиться огляд та порівняння існуючих пристроїв моніторингу, вибір апаратних і програмних засобів для побудови системи.

Система складається з апаратної частини у вигляді плати NodeMCU ESP8266 та датчика температури і вологості DHT11, а також програмної частини для обробки даних і передачі їх на віддалений сервер з використанням бездротової Wi-Fi мережі. Для зберігання та візуалізації даних застосовується хмарна платформа ThingSpeak.

Детально описано процес реалізації апаратної та програмної частин системи, інтеграцію з платформою ThingSpeak. Наведено інструкцію з використання та можливості подальшого вдосконалення системи шляхом розширення функціоналу датчиків, зміни типу живлення та покращення інтерфейсу користувача.

Розроблена система забезпечує збір, зберігання та візуалізацію даних мікроклімату з можливістю дистанційного доступу через Інтернет. Вона може бути використана в різних сферах для моніторингу кліматичних умов в реальному часі.

Ключові слова: система моніторингу, мікроклімат, датчики, NodeMCU ESP8266, ThingSpeak, збір даних, апаратна частина, програмна частина, бездротова мережа.

ABSTRACT

The thesis considers the issue of creating a system for remote monitoring of environmental climate indicators. First, an overview and comparison of existing monitoring devices are conducted, followed by the selection of hardware and software tools for building the system.

The system consists of a hardware part in the form of a NodeMCU ESP8266 board and a DHT11 temperature and humidity sensor, as well as a software part for data processing and transmission to a remote server using a wireless Wi-Fi network. The ThingSpeak cloud platform is used for data storage and visualization.

The process of implementing the hardware and software parts of the system, as well as the integration with the ThingSpeak platform, is described in detail. Instructions for use and possibilities for further improvement of the system by expanding the functionality of sensors, changing the type of power supply, and improving the user interface are provided.

The developed system ensures the collection, storage, and visualization of microclimate data with the possibility of remote access via the Internet. It can be used in various fields for real-time monitoring of climatic conditions.

Keywords: monitoring system, microclimate, sensors, NodeMCU ESP8266, ThingSpeak, data collection, hardware part, software part, wireless network.

ЧЕРКАСЬКИЙ ДЕРЖАВНИЙ БІЗНЕС-КОЛЕДЖ

Кафедра комп'ютерної інженерії та інформаційних технологій

(повна назва випускової кафедри)

Спеціальність 123 "Комп'ютерна інженерія "

(шифр і назва спеціальності)

Освітня програма Комп'ютерна інженерія

(назва освітньої програми)

ЗАТВЕРДЖУЮ

Завідувач кафедри
комп'ютерної інженерії та інформаційних технологій

(назва кафедри)

Хотунів В.І.

(підпис)

(ПІБ)

« _____ » _____ 2023 р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

Краснощокотому Дмитру Миколайовичу

(прізвище, ім'я, по батькові студента)

1. Тема кваліфікаційної роботи Система дистанційного моніторингу кліматичних показників середовища

Науковий керівник роботи к.т.н., Розломій Інна Олександрівна

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від " 13 " жовтня 2023 року № 65У.

2. Строк подання студентом випускної роботи 03.06.2024

3. Вихідні дані до випускної роботи Перелік кліматичних показників для моніторингу, необхідна точність вимірювання показників, необхідна дальність передачі даних від точок вимірювання до центру збору і обробки даних, умови експлуатації системи, мова Arduino.

4. Зміст випускної роботи (перелік питань, які потрібно розробити) визначення актуальності теми, мети, завдання, об'єкту, предмету, огляд та порівняння аналогів системи, огляд і аналіз існуючих систем дистанційного зондування та моніторингу навколишнього середовища. вибір кліматичних показників для моніторингу (температура, вологість, тиск, швидкість вітру тощо), моделювання роботи системи та перевірка працездатності, розробка рекомендацій щодо впровадження та використання системи.

5. Дата видачі завдання 15.09.2023р.

Актуальність теми	Вимірювання кліматичних показників в сучасному світі є невід'ємною частиною різноманітних сфер людської діяльності, від побутових систем у "розумних" будинках до прогнозування погоди та кліматичних змін на майбутнє. Такі пристрої та системи завжди відзначаються великим попитом.
Мета	Метою даної роботи є проектування та розробка пристрою моніторингу кліматичних показників середовища, що передає дані на віддалений сервер.
Об'єкт дослідження	Процес збору кліматичних показників середовища системою датчиків.
Предмет дослідження	Система моніторингу кліматичних показників середовища, що в реальному часі передає зібрані дані до серверу, з їх подальшою обробкою і аналізом.
Задачі які вирішуватимуться	Моніторинг температури повітря та вологості в режимі реального часу в різних точках. Відстеження рівня забруднення повітря.

Студент

(підпис)

_____ Краснощокій Д.М.

(прізвище та ініціали)

Науковий керівник роботи

(підпис)

_____ Розломій І.О.

(прізвище та ініціали)