

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

магістр

(назва освітнього ступеня)

на тему: Алгоритмічне та програмно-апаратне забезпечення керуючого модуля мобільної метеостанції для обприскувача

Виконав(ла): студент(ка) 6 курсу, групи СІМ-62
спеціальності _____

123 «Комп'ютерна інженерія»

(шифр і назва спеціальності)

_____ Муштин Д.І.
(підпис) (прізвище та ініціали)

Керівник _____ Осухівська Г.М.
(підпис) (прізвище та ініціали)

Нормоконтроль _____ Тили С.В.
(підпис) (прізвище та ініціали)

Завідувач кафедри _____ Осухівська Г.М.
(підпис) (прізвище та ініціали)

Рецензент _____
(підпис) (прізвище та ініціали)

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)
Кафедра комп'ютерних систем та мереж
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
Осухівська Г.М.
(підпис) (прізвище та ініціали)
«01» грудня 2023 р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

на здобуття освітнього ступеня магістр
(назва освітнього ступеня)
за спеціальністю 123 «Комп'ютерна інженерія»
(шифр і назва спеціальності)
студенту Муштина Денису Івановичу
(прізвище, ім'я, по батькові)

1. Тема роботи Алгоритмічне та програмно-апаратне забезпечення керуючого модуля мобільної метеостанції для обприскувача

Керівник роботи Осухівська Галина Михайлівна, к.т.н., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «01» грудня 2023 року № 4/7-1132

2. Термін подання студентом завершеної роботи 28.12.2023 р.

3. Вихідні дані до роботи принцип функціонування систем на базі технології Інтернету речей, технічні характеристики сенсорів для вимірювання метеорологічних параметрів, документація Node-red, документація Telegram-bot.

4. Зміст роботи (перелік питань, які потрібно розробити)
ЗМІСТ; ВСТУП; РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ОГЛЯД АНАЛОГІЧНИХ СИСТЕМ; РОЗДІЛ 2 ТЕХНОЛОГІЇ РОЗРОБКИ КЕРУЮЧОГО МОДУЛЯ МОБІЛЬНОЇ МЕТЕОСТАНЦІЇ; РОЗДІЛ 3 РЕАЛІЗАЦІЯ ПРИЙНЯТИХ РІШЕНЬ ТА ТЕСТУВАННЯ; РОЗДІЛ 4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ; ВИСНОВКИ.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)

1. Тема та актуальність кваліфікаційної роботи магістра.
2. Мета і завдання дослідження, об'єкт дослідження, предмет дослідження, наукова новизна.
3. Загальна архітектура керуючого модуля мобільної метеостанції для обприскувача.
4. Блок-схема процедури setup() та блок-схема алгоритму роботи функції loop() та інших пов'язаних з нею функцій
5. Потік отримання, відправки та візуалізації даних від мікроконтролера NodeMCU в Node-red, Потік прогнозу погоди по API, Візуалізація даних на моніторі оператора.
6. Потік телеграм бота в Node-red.
7. Результати роботи.
8. Висновки

АНОТАЦІЯ

Алгоритмічне та програмно-апаратне забезпечення керуючого модуля мобільної метеостанції для обприскувача // Кваліфікаційна робота магістра // Муштин Денис Іванович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра комп'ютерних систем та мереж, група СІм-62 // Тернопіль, 2023 // с. – 70, рис. – 57, табл. – 1, аркушів А1 – 8, додат. – 2, бібліогр. – 39.

Ключові слова: метеостанція, мобільна метеостанція, система моніторингу, обприскувач, мікроконтролер, Node-RED, Telegram bot.

У кваліфікаційній роботі проведено дослідження, що мають на меті підвищити ефективність внесення засобів захисту рослин шляхом розробки мобільної метеостанції для обприскувача яка надає рекомендації для оператора обприскувача, а також враховує такі параметри як дельта Т та інверсію повітря.

Синтезовано структурно-функціональну та електричну схему мобільної метеостанції для обприскувача з обґрунтуванням вибору апаратного забезпечення для неї.

Розроблено ефективний алгоритм програми для мікроконтролера NodeMCU NodeMCU по взаємодії з датчиками і передачі даних до платформи Node-RED яка забезпечує візуалізацію даних для оператора обприскувача.

Розроблено телеграм бота, який надає зручний інтерфейс для керування та отримання інформації, щодо основних фактичних параметрів на полі, а також забезпечує отримання прогнозу погоди. Основний функціонал системи дозволяє швидко коригувати процес обприскування відповідно до метеоумов і цим самим покращуючи ефективність застосування ЗЗР.

ANNOTATION

Algorithmic and Software-Hardware Provision of the Control Module for the Mobile Weather Station for Sprayers // Master's graduation thesis // Mushtyn Denys Ivanovych // Ternopil Ivan Puluj National Technical University, Faculty of Computer Information System and Software Engineering, Department of Computer Systems and Networks, group CIm-62 // Ternopil, 2023 // p. – 70, fig. – 57, tabl. – 1, sheets A1 – 8, addit. – 2, bibliography – 39.

Keywords: weather station, mobile weather station, monitoring system, sprayer, microcontroller, Node-RED, Telegram bot.

The qualification work conducted research aimed at improving the efficiency of pesticide application by developing a mobile weather station for a sprayer that provides recommendations for the sprayer operator, taking into account parameters such as delta T and atmospheric inversion.

The structural-functional and electrical scheme of the mobile weather station for the sprayer was synthesized, with justification for the selection of hardware for it.

An efficient algorithm for the program was developed for the NodeMCU ESP8266 microcontroller to interact with sensors and transmit data to the Node-RED platform, which ensures data visualization for the sprayer operator.

A Telegram bot was developed, providing a convenient interface for control and obtaining information about the main actual parameters in the field, as well as providing weather forecasts. The main functionality of the system allows for quick adjustment of the spraying process according to meteorological conditions, thereby improving the effectiveness of pesticide application.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ І ТЕРМІНІВ.....	7
ВСТУП.....	8
РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ОГЛЯД АНАЛОГІЧНИХ СИСТЕМ.....	11
1.1. Фактори, що впливають на прийняття рішення по внесенню пестицидів..	11
1.2. Огляд систем-аналогів на ринку України.....	15
1.2.1. Метеостанція для обприскувача від виробника John Deere.....	15
1.2.2. Метеостанція для обприскувача від виробника WatchDog.....	17
1.2.3. Метеостанція для обприскувача від виробника MISOL.....	18
1.2.4. Метеостанція для обприскувача від виробника МЕТЕОТРЕК.....	19
1.3. Висновки до першого розділу.....	20
РОЗДІЛ 2 ТЕХНОЛОГІЇ РОЗРОБКИ КЕРУЮЧОГО МОДУЛЯ МОБІЛЬНОЇ МЕТЕОСТАНЦІЇ.....	22
2.1. Основні засоби та технології розробки.....	22
2.2. Загальна архітектури системи.....	23
2.3. Алгоритмічне забезпечення.....	25
2.4. Обґрунтування вибору та аналіз компонентів апаратного забезпечення.....	29
2.5. Структура мобільної метеостанції.....	30
2.6. Висновки до другого розділу.....	32
РОЗДІЛ 3 РЕАЛІЗАЦІЯ ПРИЙНЯТИХ РІШЕНЬ ТА ТЕСТУВАННЯ.....	33
3.1. Опис програмного забезпечення для мікроконтролера NodeMCU.....	33
3.2. Опис програмного коду в Node-RED.....	38
3.2.1. Отримання та опрацювання даних з мікроконтролера ESP8266.....	38
3.2.2. Отримання прогнозу погоди в Node-RED.....	40
3.2.3. Створення Telegram bot в додатку Node-RED.....	42
3.3. Збір та аналіз отриманих даних.....	50

3.4. Тестування керуючого модуля мобільної метеостанції.....	51
3.5. Висновки до третього розділу.....	56
РОЗДІЛ 4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ.....	57
4.1. Охорона праці.....	57
4.2. Фактори виробничого середовища і їх вплив на життєдіяльність промислово-виробничого персоналу.....	60
4.3. Забезпечення електробезпеки користувачів ПК.....	62
4.4. Висновки до четвертого розділу.....	65
ВИСНОВКИ.....	66
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	67
Додаток А Опубліковані тези конференцій за темою кваліфікаційної роботи магістра.....	71
Додаток Б Лістинг коду.....	77

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ І ТЕРМІНІВ

АПК – агропромисловий комплекс;

ЗЗР – засоби захисту рослин;

NodeMCU - мікроконтролер NodeMCU v2 ESP8266 (CP2102);

ГРЧ – годинник реального часу;

МК – мікроконтролер;

МП – метеорологічні параметри;

НС – навколишнє середовище;

ПЗ – програмне забезпечення;

ПК – персональний комп'ютер;

ОС – операційна система;

ІоТ – Internet of Things;

GPIO – General-purpose input/output;

GPS – Global Position System;

ВДТ – відеотермінали.

ВСТУП

Актуальність теми. Динамічні зміни в погодних умовах, такі як опади, вітер, температурні коливання та вологість, суттєво впливають на ефективність спрацювання препаратів під час внесення пестицидів і тому вимагають ощадливіше ставитись до затрат пестицидів не тільки з боку екологічних норм, але і з точки зору їхньої економічної доцільності.

Одним із шляхів підвищення якості спрацювання пестицидів є контроль за погодними умовами і налаштування обприскувача до фактичних умов на кожному полі обробки. Оскільки при вдалому підборі розміру краплі, швидкості руху та норми внесення під фактичні умови поля, можна на виході отримання краще спрацювання пестицидів і як наслідок зниження виробничих затрат.

Розуміння фактичних умов на кожному полі і підбір вірних параметрів для обприскувача допомагають данні з метеостанцій, але оскільки вони динамічно змінюються з поля до поля. Один із найточнішим елементом для їх вірного підбору є монтаж метеостанції на обприскувач.

Аналізуючи існуючі рішення мобільних метеостанцій які монтуються на обприскувач було встановлено, що більшість із наявних рішень на ринку мають високу вартість і через це не набули широкої популярності, а доступні рішення використовують як правила найпростіші алгоритми і їх функціонал досить обмежений.

Зважаючи на вищесказане ринок потребує доступної по ціні метеостанції для обприскувача яка буде мати необхідний функціонал і надаватиме рекомендації для оператора обприскувача які параметри змінити для підвищення якості внесення пестицидів. При цьому буде враховувати такі параметри як дельта T, а також інверсію повітря.

Таким чином, підвищення якості внесення ЗЗР шляхом розробки доступної по ціні мобільної метеостанції є досить складною і водночас актуальною задачею.

Мета і завдання дослідження. Метою роботи є розробка мобільної метеостанції для обприскувачів, яка в режимі реального часу надає рекомендації для оператора обприскувача.

Для досягнення поставленої мети необхідно вирішити наступні основні завдання:

- виконати огляд наукової літератури, що стосується питання покращення якості внесення пестицидів та провести аналіз наявних аналогічних систем;
- описати основні засоби та технології розробки, а також архітектуру системи;
- розробити алгоритмічне та програмно-апаратне забезпечення керуючого модуля мобільної метеостанції для обприскувача;
- провести тестування керуючого модуля мобільної метеостанції для обприскувача.

Об'єкт дослідження – процес керування параметрами мобільної метеостанції для обприскувачів.

Предмет дослідження – програмно-апаратне забезпечення керуючого модуля мобільної метеостанції для обприскувачів.

Наукова новизна отриманих результатів:

1. Вперше запропоновано архітектуру комп'ютеризованої системи, яка дозволяє покращити якість внесення пестицидів завдяки отриманню прогнозованих та фактичних метеорологічних даних на полі обробки.
2. Отримали подальший розвиток підходи, щодо створення мобільних метеостанцій для обприскувачів, які мають недорогу вартість та можуть бути адаптовані під вимоги будь-якого господарства.

Практичне значення одержаних результатів кваліфікаційної роботи полягає у тому, що запропоновані та реалізовані програмно-апаратні засоби керуючого модуля мобільної метеостанції для обприскувача, який завдяки отриманню точних метеорологічних параметрів, а також швидкій комунікації між оператором обприскувача та агрономом дозволяє забезпечити точне налаштування обприскувача під метеоумови і цим самим забезпечити

підвищення якості внесення пестицидів.

Публікації. Результати кваліфікаційної роботи апробовані у вигляді двох тезах доповідей на конференціях:

1. Муштин Д.І. Мобільна метеостанція для обприскувача. Матеріали XII Міжнародної науково-практичної конференції молодих учених та студентів Тернопільського національного технічного університету імені Івана Пулюя «Актуальні задачі сучасних технологій» (6-7 грудня 2023 року). Тернопіль: ТНТУ. 2023. С. 431.

2. Осухівська Г.М., Муштин Д.І. Комп'ютеризована система контролю за метеоданими для обприскувача. Матеріали XI науково-технічної конференції Тернопільського національного технічного університету імені Івана Пулюя «Інформаційні моделі, системи та технології» (13-14 грудня 2023 року). Тернопіль: ТНТУ. 2023. С. 166.

Структура роботи. Кваліфікаційна робота містить розрахунково пояснювальну записку та графічний матеріал. До складу записки входить вступу, 4 розділи, загальні висновки, список використаних джерел і додатки. Обсяг роботи: розрахунково-пояснювальна записка – 70 арк. формату А4, графічна частина – 8 аркушів формату А1.

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ОГЛЯД АНАЛОГІЧНИХ СИСТЕМ

1.1. Фактори, що впливають на прийняття рішення по внесенню пестицидів

Один з найважливіших елементів у вирощування сільськогосподарської культури є її захист від різного роду патогенів, які, якщо їх не контролювати засобами захисту рослин, можуть повністю або частково знищити увесь майбутній урожай. Сучасне сільське господарство використовує для захисту вирощуваної культури ЗЗР, які вносяться за допомогою причіпних або самохідних обприскувачів. Для того, щоб препарати ефективно працювали їх потрібно застосовувати в правильну фазу розвитку рослини, а також враховувати метеоумови.

Оптимальним умовами для внесення ЗЗР вважаються такі, коли вологість повітря 70-80%, температура повітря знаходиться в межах від +12 до +25°C, швидкість вітру 5-6м/с, відсутня інверсія повітря, а також відсутність опадів. Будь-яке відхилення від оптимальних умови обробки знижує ефективність пестицидів, а також в деяких випадках при температурних інверсіях чи сильному вітру та інших факторах може завдати шкоди навколишньому середовищу [1].

Основні метрологічні фактори, що впливають на ефективність ЗЗР [2]:

- 1) температура під час внесення ЗЗР;
- 2) швидкість та пориви вітру на момент обприскування;
- 3) відносна вологість повітря;
- 4) ризик випадання опадів;
- 5) роса на рослинах;
- 6) сонячна інсоляція;
- 7) норма витрати робочого розчину.

Основні фактори, що впливають на прийняття рішення по внесенню ЗЗР:

1. Вітер. Під час внесення препаратів крізь отвір форсунки відбувається створення різного розміру крапель, які потім попадають на цільову поверхню. При цьому легкі дрібні краплі легко переносяться вітром за межі оброблюваного поля, великі краплі не зносяться вітром, але не позбавлені недоліків. Важкі краплі погано утримуються на оброблюваній поверхні й не забезпечують достатній ступінь покриття. Тому оптимальний діаметр крапель є компромісом між максимально щільним покриттям поверхні за мінімальної норми робочого розчину і мінімальними втратами від випаровування й знесення препарату.

Міжнародний стандарт вимірювання швидкості вітру становить 10м над рівнем землі [3].

Оскільки, як правило, метеостанції, які надають прогноз погоди знаходяться далеко від місця роботи обприскувача, тому фактичні дані можуть бути різними. Тому один із найточніших методів визначення місцевих умов є наявність метеостанції або портативного пристрою для визначення швидкості вітру (рис.1.1).



Рис.1.1. Ручний та стаціонарний варіант анемометра для вимірювання швидкості вітру

2. Вологість. Оскільки приблизно 99% об'єму бризок складається з води, випаровування цієї води може сильно вплинути на поведінку крапель. Краплі починають випаровуватися, як тільки вони залишають сопло, стаючи меншими та більш схильними до дрейфу ще в польоті.

Найпоширенішою мірою вмісту води в повітрі є відносна вологість (RH). Проте відносна вологість не розповідає всю історію, оскільки однакова відносна вологість за різних температур призводить до двох різних швидкостей випаровування води.

Найкращий показник, що може показати характеристику випаровування являється різниця показника сухого і вологого термометра. Різницю їх значень отримала назву « Дельта Т » і показані на рис.1.2 [3].

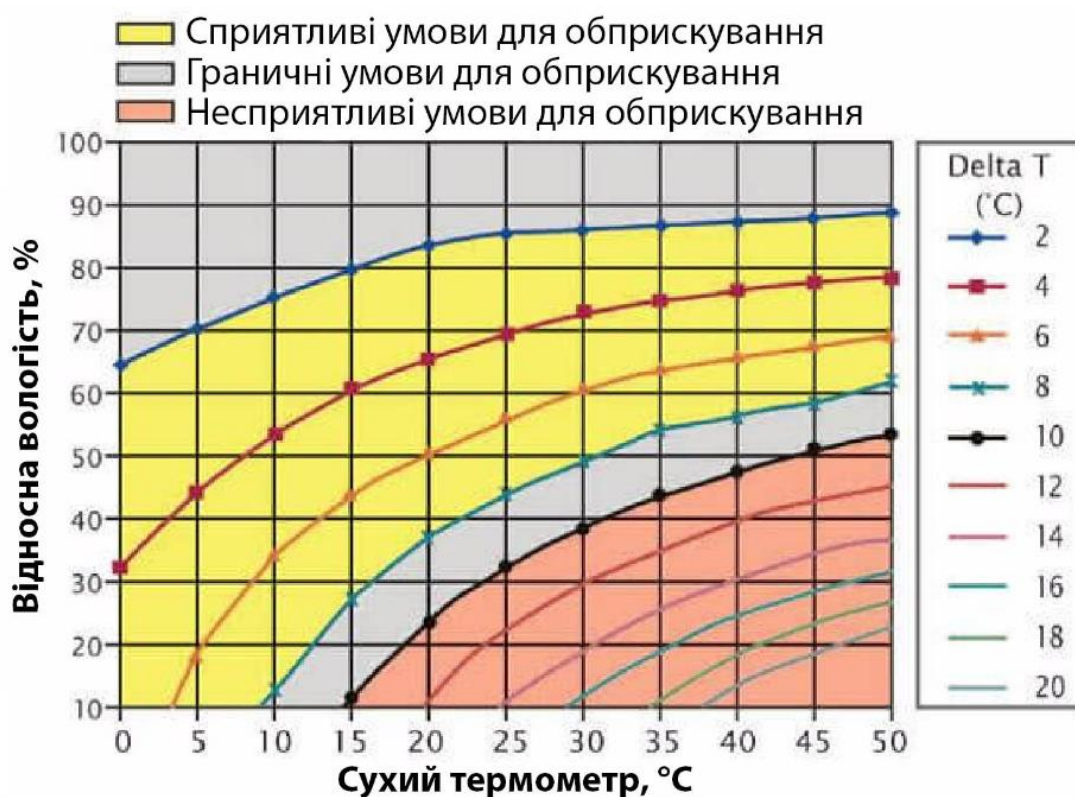


Рис.1.2. Діаграма швидкості випаровування води (Дельта Т)

3. Температура. Вплив температури на ефективність спрацювання ЗЗР не можна нехтувати, оскільки внесення ЗЗР у високу температуру суттєво впливає не тільки на ефективність спрацювання ЗЗР, а може і викликати стрес

рослини, який в подальшому вплине на кінцевий урожай. Тому для ефективного внесення пестицидів необхідно дотримуватись наступних умов:

- 1) оптимальна температура для гербіцидів зазвичай коливається від $+10^{\circ}\text{C}$ до $+25^{\circ}\text{C}$;
- 2) температура впливає на процеси обміну речовин в рослині та інші фізіологічні процеси, які мають вплив на проникну дію пестицидів;
- 3) гербіциди більш ефективні при температурі від $+18^{\circ}\text{C}$ до $+24^{\circ}\text{C}$;
- 4) інсектициди найкраще працюють за температури від $+15^{\circ}\text{C}$ до $+24^{\circ}\text{C}$.

Забезпечення належного теплового режиму допомагає підвищити ефективність спрацювання пестицидів [4].

Температурні інверсії. Температурна інверсія - це метеорологічний феномен, при якому температура повітря зростає зі зростанням висоти, ніж зазвичай зменшується. Тобто, в оберненому порядку від звичайного градієнту температури, де повітря стає прохолоднішим при підйомі вгору.

Температурні інверсії мають негативний вплив на обприскування, а саме:

- 1) Обмеження розподілу пестицидів. У температурних інверсіях тепле повітря знаходиться вище, а холодне - ближче до землі. Це може призводити до обмеження вертикального руху повітря, що заважає покриттю препаратів.
- 2) Затримка дії хімічних речовин. У температурних інверсіях розподіл хімічних речовин може бути непрогнозованим, оскільки їхні пари можуть залишатися близько до поверхні. Це може призвести до недостатньої ефективності обробки і потребує збільшення норми внесення пестицидів.
- 3) Зависання препарату в повітрі у зв'язку відсутності вітру.

Для уникнення негативного впливу температурних інверсій на обприскування, фермери і оператори мають бути уважні до метеорологічних умов і враховувати їх при плануванні обприскувань.

Більшість метеорологічних служб активно не вимірюють інверсії [4]. І тому доцільно використовувати метеостанцію для визначення її на полі.

1.2. Огляд систем-аналогів на ринку України

На ринку існує багато виробників метеостанцій, але, в основному, вони стаціонарні і тільки деякі із виробників виготовляють мобільні версії, які можуть монтуватись на обприскувачі.

Нижче додана інформацію про найпопулярніші варіанти метеостанцій, що монтуються на обприскувачі:

1.2.1. Метеостанція для обприскувача від виробника John Deere. На рис.1.3. показана мобільна метеостанція американського виробника John Deere яка широко представлена на ринку України і виключно монтується на техніку John Deere [5].



Рис.1.3. Мобільна метеостанція виробника John Deere

John Deere Mobile Weather вимірює та відображає наступні данні:

- 1) температуру повітря;
- 2) вологість повітря;
- 3) швидкість вітру;
- 4) напрямок вітру;
- 5) дельта Т.

Моніторинг температури, швидкості вітру, напрямку вітру, дельта T і відносної вологості на ходу дозволяє оператору швидко і легко зрозуміти, чи сприятливі погодні умови для застосування продукту. Вбудовані оповіщення дозволяють оператору легко встановлювати робочі параметри. Наявність інформації про погоду дозволяє підвищити ефективність продукту та зменшити ризик хімічного дрейфу.

Візуалізація даних в моніторі обприскувача показана на рис.1.4. Звіт, на який надає платформа як факт виконаної роботи показано на рис.1.5.

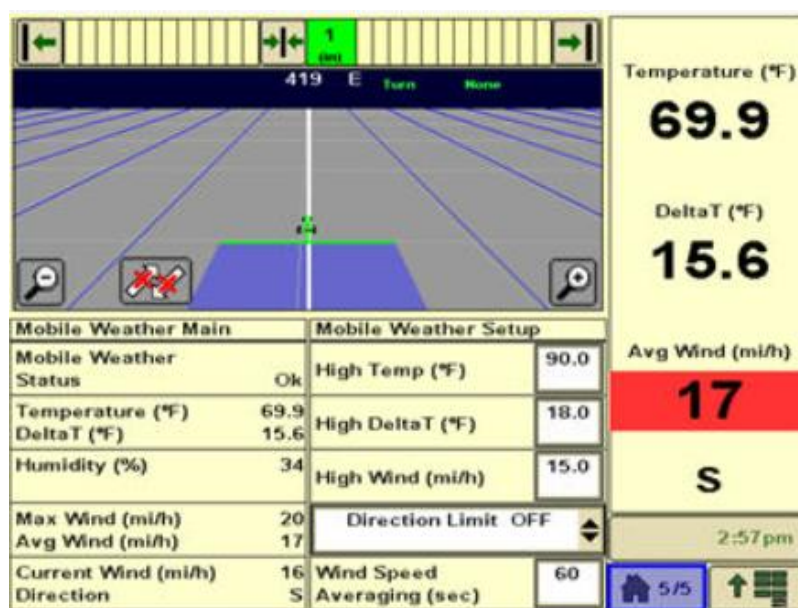


Рис.1.4. Візуалізація даних на моніторі обприскувача

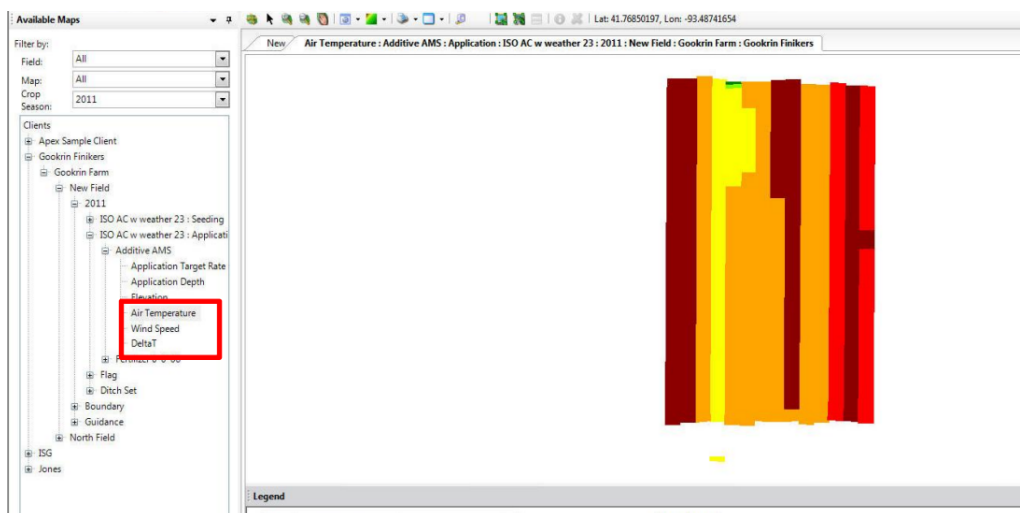


Рис.1.5. Звіт по отриманим даним

Якщо проаналізувати наявний функціонал метеостанції, який не надає розуміння по прогнозу погоди, а також враховуючи вартість системи, яка коштує 4500 дол., то можна зробити висновки, чому так і не набула дана метеостанція широкої популярності на ринку України.

1.2.2. Метеостанція для обприскувача від виробника WatchDog. Мобільна метеостанція для оприскувача від американського виробника WatchDog (рис.1.6.) також досить популярне рішення на Українському ринку [7]. Її функціонал дозволяє вимірювати такі параметри як температуру повітря, вологість, точку роси, охолодження вітру, барометричний тиск, швидкість та напрям вітру.

Метеостанція використовує внутрішній компас і GPS для налаштування значень датчика вітру шляхом віднімання напрямку транспортного засобу від швидкості вітру для обчислення напрямку вітру, а також справжньої швидкості вітру.



Рис.1.6. Мобільна метеостанція виробника WatchDog

Функціонал метеостанції дозволяє передачу даних на монітор, який стоїть в обприскувачі таких виробників як: Raven, Trimble та або консоль [8]. При цьому система тільки передає фактичні дані вимірів на монітор і не надає

можливості їхнього бездротової передачі на сервер господарства, але при цьому вона більш доступна та її вартість на ринку України становить 120 тис. грн.

1.2.3. Метеостанція для обприскувача від виробника MISOL. Метеостанція китайського виробництва MISOL модель HP2550 показана на рис. 1.7 [8].



Рис.1.7. Мобільна метеостанція виробника MISOL модель HP2550

MISOL HP2550 – це метеостанція для відслідковування та прогнозування погодних умов. Функціонал метеостанції дозволяє використовувати її для прогнозування процесу обприскування.

Метеостанція забезпечує вимірювання наступних параметрів як: температуру, відносну вологість, кількість опадів, напрямок вітру, швидкість вітру, пориви вітру, сонячне випромінювання, ультрафіолетове випромінювання, барометричний тиск, індекс холоду, точку роси, схід захід сонця, прогноз погоди.

Аналізуючи дану метеостанцію, а також інші метеостанції китайського виробники, було встановлено, що більшість із них мають подібний функціонал. Їх алгоритмічне та програмно-апаратне забезпечення є подібним. Дані з датчиків, як правило, не передаються на сервер в режимі реального часу для аналізу, а записуються тільки на карту пам'яті у форматі (*.csv), які потім можна

використати для аналізу.

Деякі метеостанції китайського виробництва мають можливість передачі даних онлайн завдяки наявному wi-fi модулю на сторонні сайти, але функціонал даних сайтів досить обмежений.

Вартість метеостанції виробника MISOL моделі HP2550 становить 26 тис. грн. При цьому в модельному ряді присутні метеостанції від 12 тис. грн для домашнього використання і до 40 тис. грн.

1.2.4. Метеостанція для обприскувача від виробника МЕТЕОТРЕК. На українському ринку також присутній вітчизняний виробник метеостанцій, які реалізуються під брендом МЕТЕОТРЕК. На рис.1.8. показано мобільна метеостанція виробника МЕТЕОТРЕК моделі МОВІ [9].



Рис.1.8. Мобільна метеостанція виробника МЕТЕОТРЕК модель МОВІ

Мобільна метеостанція створена для встановлення на рухомій техніці для визначення метеорологічних умов при здійсненні процедури обприскування.

Перегляд параметрів метеостанції відбувається через мобільний застосунок Meteotrek МОВІ, який працює на ОС iOS або Android (рис.1.9).



Рис.1.9. Мобільний застосунок Meteotrek MOBI

Функціонал метеостанції створений таким чином, що фактичні дані з метеостанції показуються в мобільному застосунку, а параметри прогнозу погоди, а також аналіз даних можливий тільки у десктопній версії. При цьому дані з метеостанції у веб-застосунок передаються через наявний Wi-fi.

Аналізуючи основні недоліки Meteotrek MOBI можна зауважити те, що вона не надає повноти картини по прогнозу погоди, а також динаміку зміни основних параметрів для оператора, а тільки надає розуміння для контролера, який особисто приймає рішення по змінні параметрів внесення ЗЗР.

1.3. Висновки до першого розділу

Проаналізовано фактори, що впливають на прийняття рішення по внесенню пестицидів і з яких встановлено, що найбільший вплив на якість спрацювання пестицидів мають фактичні метеоумови на полі, а саме: температура повітря, відносна вологість повітря, швидкість вітру, ризик випадання опадів, Дельта Т та температурні інверсії.

Проведено огляд найпопулярніших мобільних метеостанцій для обприскувачів з якого встановлено, що є потреба у доступних за ціною та функціональних метеостанцій для обприскувачів, які матимуть можливість швидко надавати рекомендації для оператора обприскувача, при цьому враховуватимуть такі параметри як Delta T та інверсію повітря.

Зважаючи на вищесказане ринок потребує доступної по ціні метеостанції для обприскувача, яка буде мати простий функціонал і надаватиме рекомендації для оператора обприскувача. При цьому буде враховувати такі параметри як Delta T, а також інверсію повітря.

РОЗДІЛ 2

ТЕХНОЛОГІЇ РОЗРОБКИ КЕРУЮЧОГО МОДУЛЯ МОБІЛЬНОЇ МЕТЕОСТАНЦІЇ

2.1. Основні засоби та технології розробки

Розробка комп'ютеризованої системи контролю за метеоданими для обприскувача, яка би враховувала фактичні умови поля, прогноз погоди від найближчих метеостанцій, а також надавала рекомендації оператору обприскувача про підбір розміру краплі, швидкості руху та норми внесення пестицидів при цьому була б доступна по вартості є важливою та актуальною задачею.

У проекті Arduino IDE використовувалось для програмування мікроконтролера ESP8266 NodeMCU, який виконує роль апаратного забезпечення для збору даних з таких датчиків як: 2-х датчиків температури та вологості DHT-22 або DHT-11, датчик вітру ZTS-3000-FSJT, датчик атмосферного тиску BMP180, а також GPS приймача E108-GN02D.

Основною причиною використання Arduino IDE є те, що для вибраного типу мікроконтролера дане середовище є найпопулярнішим, оскільки має велику кількість бібліотек, які забезпечують реалізацію різного типу задач.

Наступним інструментом, який виконує роль сервера по збору даних, опрацювання їх, а також забезпечує візуалізацію отриманого результату є Node-RED.

Node-RED — середовище для візуального програмування потоком даних, який розроблений компанією IBM з цілю поєднання різних пристроїв, API та онлайн-сервісів як одних із важливих елементів Інтернету речей [21].

Node-RED - це середовище, що дозволяє створювати програми для обробки потоку даних у веб-браузері. Засноване на Node.js, воно надає можливість використання готових або самостійно створених вузлів для обробки різних функцій. Вузли можна отримувати як з вбудованих, так і з бібліотеки NPM, або створювати свої власні. Програми, розроблені в Node-RED

зберігаються у форматі JSON та можуть використовуватись для різноманітних завдань обробки даних.

У якості апаратної частини, де інстальовано Node-RED використовувався ноутбук з операційною системою Windows, при цьому гнучкість системи дозволяє здійснити інсталяцію на широкий вибір ОС в тому числі - Android.

У проекті Node-RED забезпечує виконання наступних завдань:

1. отримує дані з мікроконтролера ESP8266 NodeMCU по Wi-Fi по протоколу MQTT;
2. опрацьовує отриманні дані, а також надає їх візуалізацію у вигляді графіків;
3. отриманні фактичні дані з мікроконтролера ESP8266 завантажуються в онлайн-платформу ThingSpeak, яка використовується у якості бази даних і їх подальшого поглибленого аналізу;
4. завантажує дані прогнозу погоди з сайту OpenWeather, які потім транслюються на платформі Node-RED;
5. виконує роль сервера для Telegram bot, що в свою чергу забезпечує комунікацію монітору Node-RED, який бачить оператор обприскувача перед собою з контролером, та за допомогою Telegram bot може віддалено надавати команди по коректуванні процесом внесення пестицидів на основі отримання фактичних даних. Також функціонал розробленого телеграм бота дозволяє для контролера отримати прогноз погоди з 4-х джерел відповідно до місцезнаходження обприскувача на полі, отримати дані точного місцезнаходження обприскувача на полі по його GPS-координатам та, за необхідності, отримати інструкцію користувача через бот.

2.2. Загальна архітектури системи.

Для реалізації мети кваліфікаційної роботи була розроблена загальна архітектура керуючого модуля мобільної метеостанції для обприскувача, яка зображена на рис. 2.1.

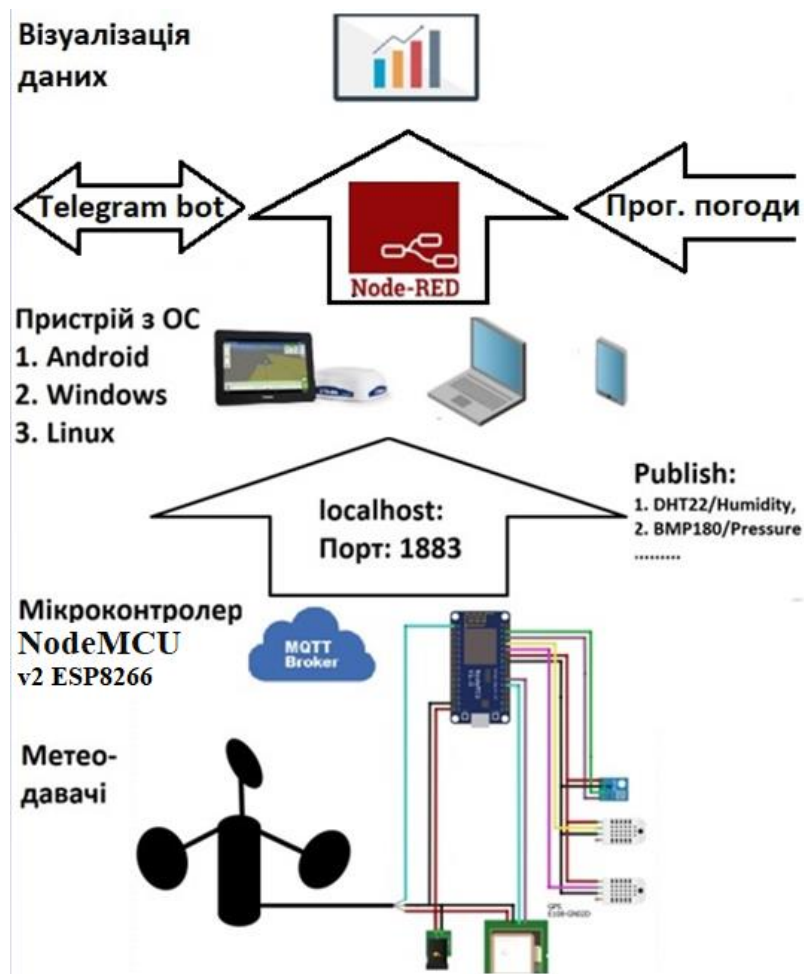


Рис.2.1. Загальна архітектура керуючого модуля мобільної метеостанції для обприскувача

Керуючий модуль мобільної метеостанції розроблено на базі NodeMCU v2 ESP8266(рис.2.3), який збирає дані від 2-х давачів температури та вологості DHT-22 або DHT-11, давача вітру ZTS-3000-FSJT, давача атмосферного тиску BMP180, а також GPS приймача E108-GN02D. Усі зібрані дані від давачів передаються на NodeMCU, який обробляє їх і в подальшому передає по протоколу MQTT. У даному рішенні NodeMCU виконує роль брокера.

Дані від брокера MQTT передаються через Wi-Fi модуль на інший пристрій, який працює на ОС Windows або Linux чи будь-якій іншій ОС, які мають можливість інсталяції інструменту для візуалізації програмних потоків даних Node-RED. Node-RED забезпечує обробку даних, візуалізацію, а також отримання прогнозу погоди на кожному конкретному полі по GPS координатах.

2.3. Алгоритмічне забезпечення

Алгоритмічне забезпечення включає побудову блок-схеми для двох основних функцій: `setup()` і `loop()`. При запуску пристрою спочатку виконується функція `setup()`, яка відповідає за конфігурацію налаштувань мікроконтролера та його периферії показана на рис.2.2.

Алгоритм роботи функції `setup()` з врахуванням затримок можна розбити на наступні кроки:

1. Встановлення зв'язку між мікроконтролером та комп'ютером за допомогою серійного порту. Значення швидкості передачі даних для серійного порту мікроконтролера становить 115200 біт на секунду.

2. Підключення до мережі Wi-Fi встановлюється викликом функції `startWiFiClient()`.

Основні її складові:

- Вивід повідомлення про спробу підключення до мережі Wi-Fi.
- Запуск процесу підключення за допомогою `WiFi.begin(ssid, pass)`.
- Очікування на успішне підключення з інтервалом у 2 секунди (`delay(2000)`) та вивід крапки протягом цього часу.
- Вивід інформації про успішне підключення до мережі Wi-Fi разом із виведенням локальної IP-адреси.

3. Ініціалізація MQTT брокера. Викликом функції `myBroker.init()`, яка ініціалізує MQTT брокер на мікроконтролері NodeMCU.

4. Ініціалізація датчиків температури, вологості, атмосферного тиску та GPS-модуля функціями:

- `dht.begin()`: ініціалізація датчика DHT22 для вимірювання температури та вологості.

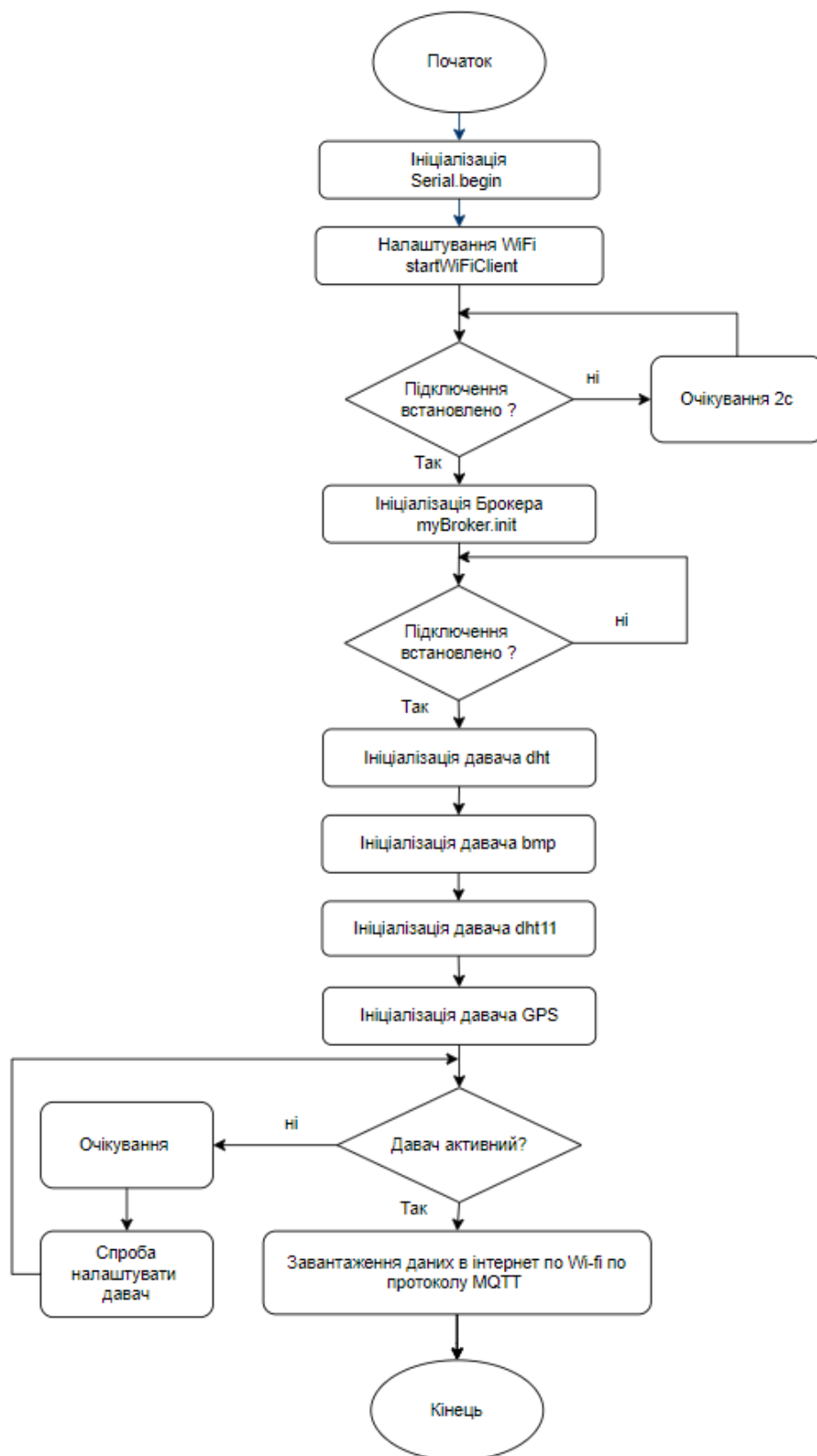


Рис.2.2. Блок-схема процедури setup()

- bmp.begin(): ініціалізація датчика BMP180 для вимірювання тиску та температури.

- dht11.begin(): ініціалізація датчика DHT11 для вимірювання температури

та вологості.

- `ss.begin(9600)`: ініціалізація з'єднання з GPS-модулем за допомогою `SoftwareSerial`.

5. Виведення інформації про початок роботи MQTT брокера та ініціалізованих давачів.

Після успішної ініціалізації усіх давачів відбувається зчитування даних, якщо ініціалізація не успішна, то відбувається повторна ініціалізація.

Алгоритм роботи функції `loop()` використовує концепцію скінченного автомата для організації роботи програми в режимі `endle`-стійкому. Основний цикл викликається безперервно і виконує обробку в залежності від поточного стану програми.

Алгоритм роботи функції `loop()` можна розбити на наступні кроки:

1. GPS Дані. Початково, у внутрішньому циклі перевіряється наявність нових даних від GPS. Якщо дані отримані, вони обробляються за допомогою бібліотеки `TinyGPS++`, і викликається функція `publishGPSData()`, яка виводить GPS-дані через MQTT брокер

2. Станові машини. Потім, у зовнішньому циклі, використовується конструкція `switch-case`, щоб визначити поведінку програми в залежності від поточного стану (змінна `currentState`).

3. Обробка станів. У кожному з випадків `switch-case` виконується відповідна дія в залежності від поточного стану

- 3.1. `STATE_READ_SENSORS`:

- Зчитування даних з датчиків за допомогою функції `readSensors()`.
- Зміна стану на `STATE_PUBLISH_DATA` після зчитування.

- 3.2. `STATE_PUBLISH_DATA`:

- Публікація зчитаних даних через MQTT брокер за допомогою функції `publishData()`.

- Зміна стану на `STATE_READ_SENSORS` після публікації.

4. Таймери для інтервалів. Основна мета використання таймеру інтервалів полягає в зменшенні надлишковості у зчитуванні та публікації даних.

Для керування часовими інтервалами використовуються таймери

(previousMillisReadSensors, previousMillisPublishData, intervalReadSensors, intervalPublishData). Таймери дозволяють визначити, коли слід виконати зчитування даних чи публікацію.

Блок-схема алгоритму роботи функції loop() та інших функцій показана на рис.2.3.

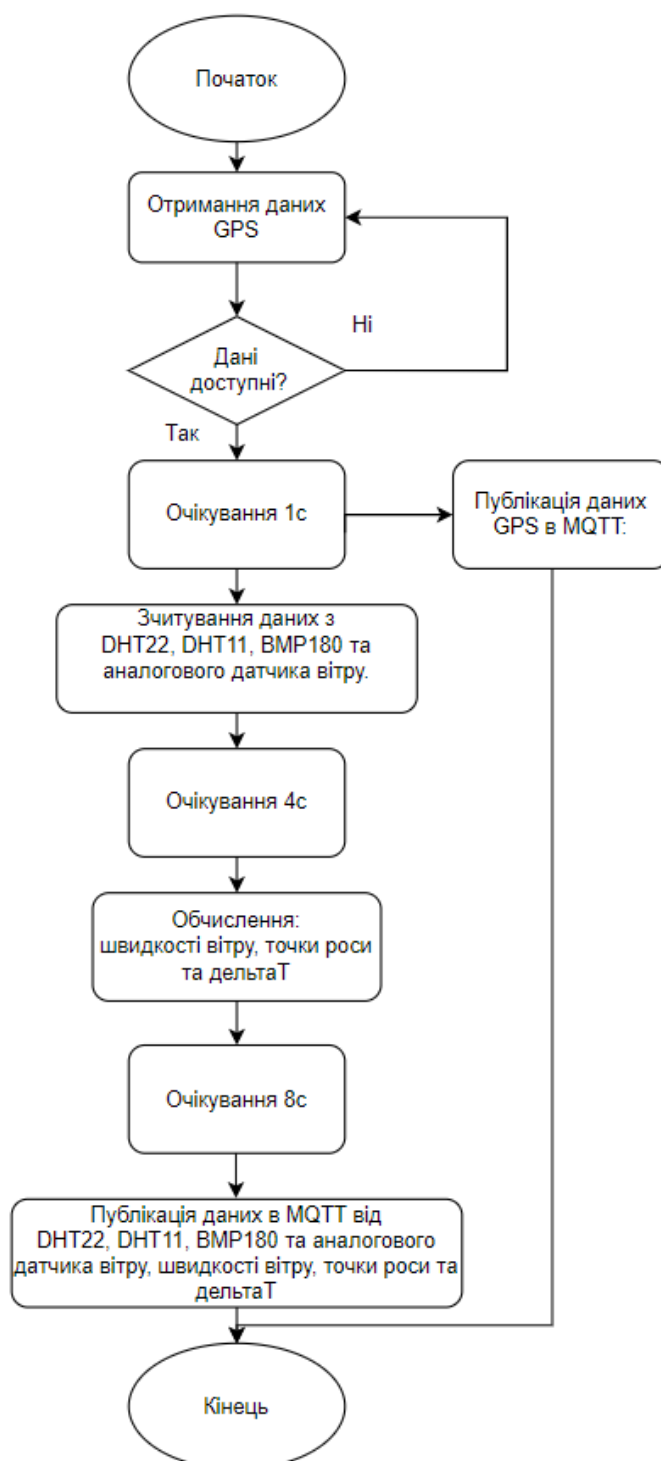


Рис.2.3. Блок-схема алгоритму роботи функції loop() та інших пов'язаних з нею функцій

2.4. Обґрунтування вибору та аналіз компонентів апаратного забезпечення

Для забезпечення збору метеопараметрів у проекті використано NodeMCU v2 ESP8266. NodeMCU - це плата, яка побудована на базі мікроконтролера ESP8266 і являє собою UART-WiFi модуль з низьким споживанням.

Використання NodeMCU обґрунтоване наявністю вбудованого Wi-Fi модуля, що робить його ідеальним вибором для метеостанції, оскільки для даного проекту передбачається передача даних через бездротову мережу. Мікроконтролер оснащений потужним одноядерним процесором, а також має широкі можливості по підключенню різних датчиків, які працюють по різних інтерфейсам. Для написання коду використовується популярне середовище розробки Arduino IDE з мовою програмування C/C++.

Поєднання даних параметрів забезпечить достатню продуктивність та функціонал для виконання поставленого на нього завдання.

Для вимірювання атмосферного тиску в проекті застосовувався барометр BMP180. BMP180 - це датчик атмосферного тиску, який також має можливість вимірювати температуру. Управляється по шині I2C і є найпопулярнішим датчиком для вимірювання атмосферного тиску, датчик досить точний, надійний та доступний по ціні.

Один із важливих елементів для метеостанції є визначення фактичної швидкості вітру, оскільки від даного параметру залежить спрацювання ЗЗР. Для вимірювання даного параметра застосовувався анемометр ZTS-3000-FSJT китайського виробництва, який є недорогим та популярним.

Для отримання даних про швидкість руху, а також GPS координати обприскувача на полі, застосовувався GPS датчик E108-GN02D.

Для вимірювання відносної вологості та температури можуть застосовувались датчики DHT-11 та DHT-22. Основні технічні характеристики датчиків наведені у таблиці 2.1.

DHT11 та DHT22 - це два різних сенсори температури та вологості, які часто використовуються в проектах IoT, оскільки для багатьох проектів їх

точності вимірів цілком достатньо, тому у даному проєкті буде застосовуватись

2. В промисловому виконанні виробу метеостанції доцільніше використовувати давач DHT-22, оскільки він вимірює температуру в ширшому діапазоні і має більшу точність.

Таблиця 2.1

Основні технічні характеристики давачів DHT-11 та DHT-22[15-17]

Характеристика	DHT-11	DHT-22
Діапазон вимірювання температур, °C	0 до 50 (±2 точність)	-40 до 80 (±0.5 точність)
Діапазон вимірювання вологості, %	20 до 80 (±5 точність)	0 до 100 (±2-5 точність)
Робочий діапазон напруги, V	3.3 - 5	3.3 - 6
Інтерфейс	Цифровий	Цифровий
Характеристика	Кожні 2 секунди	Кожні 2 секунди
Діапазон вимірювання температур, °C	12 x 15 x 5.5	15.1 x 25 x 7.7

2.5. Структура мобільної метеостанції

На рис.2.4 зображена узагальнена схема мобільної метеостанції.

В розробленій системі використовуються:

- 1) nodeMCU;
- 2) давач DHT-22;
- 3) давач DHT-11;
- 4) gps модуль E108-GN02D;
- 5) аналоговий анемометр ZTS-3000-FSJT;
- 6) барометр BMP-180;
- 7) комп'ютер чи смартфон для візуалізації даних;
- 8) лінійний стабілізатор напруги LM317T, який перетворює 12В в 5В.

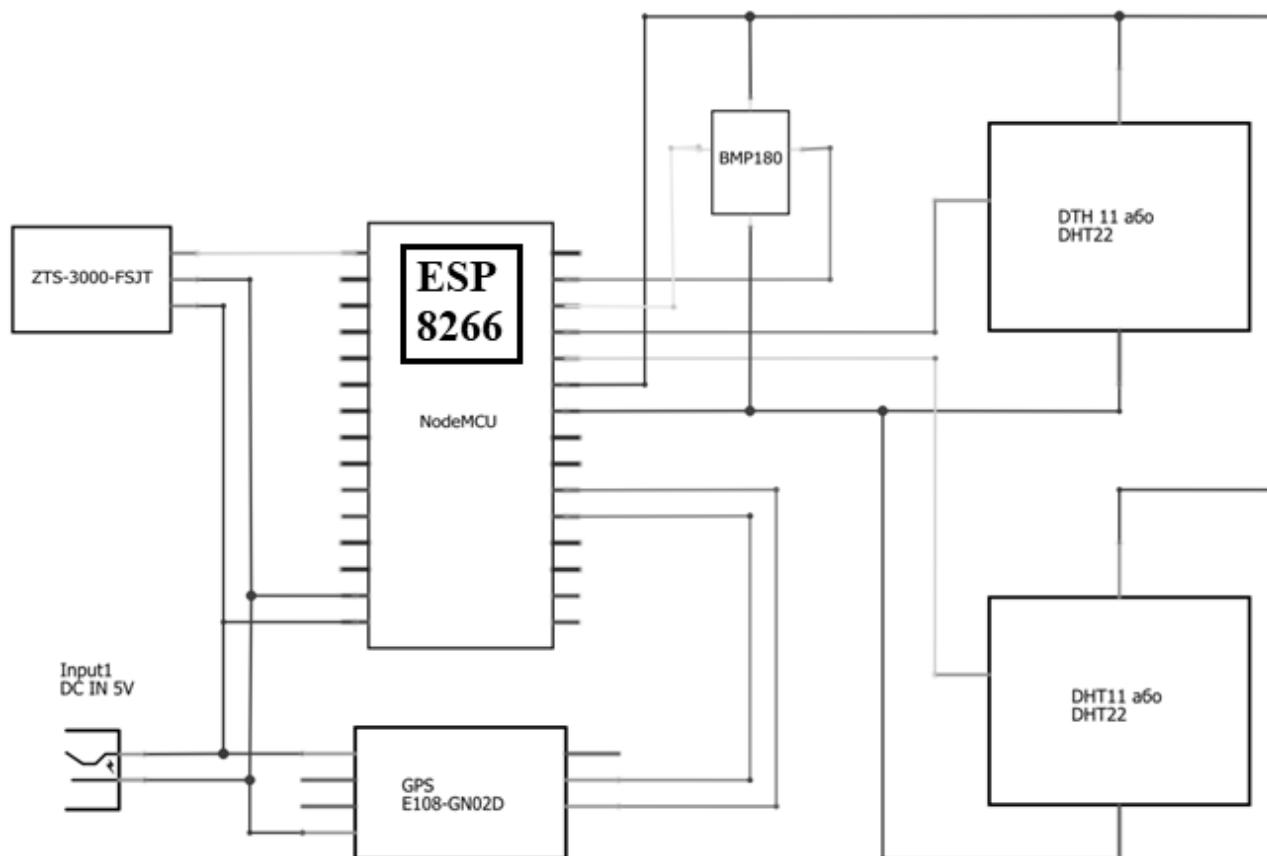


Рис.2.4. Узагальнена схема мобільної метеостанції

Слід зазначити, що в даному проекті використовуються також додатково встановлена кнопка “RESTART” для NodeMCU, яка в майбутньому буде винесена на захисний корпус мікроконтролера. Дане рішення не є обов’язковим для виконання, але з практичної точки зору вона додатково встановлюється.

Для вимірювання швидкості вітру застосовується анемометер. На ринку більшість із них можуть мати живлення 5-10В або 10-30В в залежності від конкретної моделі, схема підключення може бути модифікована по живленню, а також по підключенню. На схемі, яка показана на рис.2.4, зображено рішення коли у нас анемометер працює від 5V для спрощення.

Оскільки у нашому проекті використовується давач ZTS-3000-FSJT [10], що працює від напруги живлення 10-30В, тому до давача анемометра підводиться окрема напруга 12В від акумулятора.

2.6. Висновки до другого розділу

Описано основні засоби та технології, які використовувались для побудови керуючого модуля мобільної метеостанції для обприскувача.

Розроблено загальну архітектуру, а також алгоритмічне забезпечення керуючого модуля мобільної метеостанції для обприскувача, а також наведено принцип роботи метеостанції.

Обґрунтовано вибір апаратного забезпечення на основі NodeMCU v2 ESP8266.

РОЗДІЛ 3

РЕАЛІЗАЦІЯ ПРИЙНЯТИХ РІШЕНЬ ТА ТЕСТУВАННЯ

3.1. Опис програмного забезпечення для мікроконтролера NodeMCU

Відповідно до запропонованого алгоритмічного забезпечення керуючого модуля мобільної метеостанції, а також враховуючи загальну архітектуру системи, розроблено програмне забезпечення.

Основна програма для мікроконтролера NodeMCU починається з підключення бібліотек, які потрібні для роботи (рис.3.1).

```
#include <ESP8266WiFi.h>
#include "uMQTTBroker.h"
#include <Adafruit_BMP085.h>
#include <DHT.h>
#include <TinyGPS++.h>
#include <SoftwareSerial.h>
#include <PubSubClient.h>
```

Рис.3.1. Лістинг коду з підключення бібліотек

Після оголошення бібліотек встановлюються параметри для підключення Wi-Fi мережі, здійснюється присвоєння для датчиків відповідних виводів на мікроконтролері, а також присвоєння константних значень (рис.3.2).

```
uMQTTBroker myBroker;
char ssid[] = "Zanna-S";
char pass[] = "05217847";

#define DHT_PIN D3
#define DHT11_PIN D4
#define BMP_SDA D2
#define BMP_SCL D1

Adafruit_BMP085 bmp;
DHT dht(DHT_PIN, DHT22);
DHT dht11(DHT11_PIN, DHT11);

const int sensorPin = A0;
float voltageMin = 0.645;
float windSpeedMin = 0;
float voltageMax = 3.3;
float windSpeedMax = 61.2;

TinyGPSPlus gps;
static const int RXPin = D8;
static const int TXPin = D7;
SoftwareSerial ss(RXPin, TXPin);
```

Рис.3.2. Лістинг коду з визначенням виводів на мікроконтролері

З метою підключення NodeMCU до Wi-Fi мережі як клієнта або для створення точки доступу Wi-Fi застосовувались функції `void startWiFiClient()` та `startWiFiAP()` (рис.3.3).

```
void startWiFiClient() {
    Serial.println("Connecting to " + (String)ssid);
    WiFi.begin(ssid, pass);

    while (WiFi.status() != WL_CONNECTED) {
        delay(2000);
        Serial.print(".");
    }
    Serial.println("");
    Serial.println("WiFi connected");
    Serial.println("IP address: " + WiFi.localIP().toString());
}

void startWiFiAP() {
    WiFi.softAP(ssid, pass);
    Serial.println("AP started");
    Serial.println("IP address: " + WiFi.softAPIP().toString());
}
```

Рис.3.3. Лістинг коду підключення ESP8266 до Wi-Fi мережі

Наступна функцією у коді є виклик функція `setup()`. Функція `setup()` здійснюється при появі живлення і тільки один раз при запуску пристрою або його перезапуску (рис.3.4).

```
void setup() {
    Serial.begin(115200);
    startWiFiClient();
    myBroker.init();
    dht.begin();
    bmp.begin();
    dht11.begin();
    ss.begin(9600);
    Serial.println("Starting MQTT broker");
    Serial.println("Датчики ініціалізовано");
}
```

Рис.3.4. Лістинг коду функції `setup()`

Після ініціалізації функцією `setup()` виконується функція `loop()`. Для зчитування даних від GPS-модуля E108-GN02D (рис.3.5) використано цикл `while`, який буде виконуватися доки у вхідному буфері `SoftwareSerial (ss)` є доступні дані (`ss.available() > 0`). Це означає, що цикл буде тривати, доки не будуть прочитані всі доступні дані з вхідного буфера. У тілі циклу викликається метод `ss.read()`, який читає один байт з вхідного буфера `SoftwareSerial`. Прочитані дані потім передаються методу `gps.encode()`. Метод `gps.encode()` використовується для обробки байтів, які надходять від GPS-модуля.

```
while (ss.available() > 0) {
  if (gps.encode(ss.read())) {
    publishGPSData();
  }
}
```

Рис.3.5. Лістинг коду для читання даних від GPS-модуля

Якщо `gps.encode()` повертає `true` (тобто, якщо вдалося розпізнати повідомлення GPS, тоді викликається функція `publishGPSData()` (Рис.3.6). Ця функція відповідає за публікацію даних GPS в MQTT брокер.

```
void publishGPSData() {
  myBroker.publish("GPS/Latitude", String(gps.location.lat(), 6).c_str());
  myBroker.publish("GPS/Longitude", String(gps.location.lng(), 6).c_str());
  myBroker.publish("GPS/Altitude", String(gps.altitude.meters()).c_str());
  myBroker.publish("GPS/Speed", String(gps.speed.kmph()).c_str());
  myBroker.publish("GPS/Satellites", String(gps.satellites.value()).c_str());
  delay(1000);
}
```

Рис.3.6. Лістинг коду для публікації даних від GPS-модуля

Наступний елемент у циклі `loop()` застосовує метод керування станами `switch-case` [26]. Реалізація цього методу показана на рис.3.7.

```

unsigned long currentMillisReadSensors;
unsigned long currentMillisPublishData;

switch (currentState) {
  case STATE_READ_SENSORS:
    currentMillisReadSensors = millis();
    if (currentMillisReadSensors - previousMillisReadSensors >= intervalReadSensors) {
      readSensors();
      previousMillisReadSensors = currentMillisReadSensors;
      currentState = STATE_PUBLISH_DATA;
    }
    break;

  case STATE_PUBLISH_DATA:
    currentMillisPublishData = millis();
    if (currentMillisPublishData - previousMillisPublishData >= intervalPublishData) {
      publishData();
      previousMillisPublishData = currentMillisPublishData;
      currentState = STATE_READ_SENSORS;
    }
    break;
}
}

```

Рис.3.7. Лістинг коду для реалізації затримок у читанні даних з давачів та публікації даних

Використання затримок з таймерами станів забезпечує програмі ефективно виконувати зчитування даних та їх публікацію, уникнути блокуючих операцій, які були присутні при використанні стандартних затримок, а також дозволили покращити контроль над часовими інтервалами між операціями читанням і публікації даних.

Наступний фрагмент коду використовується для зчитування даних від 2-х давачів температури та вологості DHT-22 та DHT-11 і давача атмосферного тиску BMP180 (рис.3.8).

```

humidity = dht.readHumidity();
temperature = dht.readTemperature();
pressure = bmp.readPressure() / 100.0;
temperatureBMP = bmp.readTemperature();
humidityDHT11 = dht11.readHumidity();
temperatureDHT11 = dht11.readTemperature();

```

Рис.3.8. Лістинг коду для зчитування даних від 2-х давачів температури та вологості DHT-22 та DHT-11 і давача атмосферного тиску BMP180

З метою отримання вірних даних з давача вітру ZTS-3000-FSJT його спочатку потрібно відкалібрувати. Для цього спочатку необхідно знайти додаткові значення такі як `sensorValue` та `voltage` (рис.3.9), які в подальшому необхідні для застосування методу `mapfloat` (рис.3.10), який дозволяє отримати необхідні калібрувальні константи, показані на рис.3.1.

```
float sensorValue;
float voltage;
float wind_speed;

sensorValue = analogRead(sensorPin);
voltage = (sensorValue / 1024.0) * 3.3;
wind_speed = mapfloat(voltage, voltageMin, voltageMax, windSpeedMin, windSpeedMax);
```

Рис.3.9. Лістинг коду для зчитування даних з давача вітру ZTS-3000-FSJT

```
float mapfloat(float x, float in_min, float in_max, float out_min, float out_max) {
  return (x - in_min) * (out_max - out_min) / (in_max - in_min) + out_min;
}
```

Рис.3.10. Лістинг коду методу `mapfloat`

З метою перевірки коректності отриманих даних з давача вітру, а також використовуючи рекомендації заводу виробника по налаштуванням [18] застосовувалось вивід даних в серіал порт (рис.3.11).

```
Serial.print("Analog Value =");
Serial.println(sensorValue);
Serial.print("Voltage =");
Serial.print(voltage);
Serial.println(" V");
Serial.print("Wind Speed =");
Serial.print(wind_speed);
Serial.println("m/s");
```

Рис.3.11. Лістинг коду для виведення параметрів Analog Value, Voltage, Wind Speed у серіал порт

Визначення параметрів точки роси і ДельтаТ показано на рис.3.12.

```
dewpoint = temperature - (100 - humidity) / 5;
deltaT = temperature - dewpoint;
```

Рис.3.12. Лістинг коду для визначення параметрів Точки роси і ДельтаТ

Лістинг коду для публікація даних на сервер від 2-х датчиків температури та вологості DHT-22 та DHT-11, датчик вітру ZTS-3000-FSJT, датчик атмосферного тиску BMP180 показана на рис.3.13.

```
myBroker.publish("DHT22/Humidity", String(humidity));
myBroker.publish("DHT22/Temperature", String(temperature));
myBroker.publish("BMP180/Pressure", String(pressure));
myBroker.publish("BMP180/Temperature", String(temperatureBMP));
myBroker.publish("DHT11/Humidity", String(humidityDHT11));
myBroker.publish("DHT11/Temperature", String(temperatureDHT11));
myBroker.publish("WindSen/wind_speed", String(wind_speed));
myBroker.publish("DHT11/dewpoint", String(dewpoint));
myBroker.publish("DHT11/deltaT", String(deltaT));
```

Рис.3.13. Лістинг коду для публікації даних від 2-х датчиків температури та вологості DHT-22 та DHT-11, датчик атмосферного тиску BMP180, датчик вітру ZTS-3000-FSJT, а також параметрів Точки роси і ДельтаТ

3.2. Опис програмного коду в Node-RED

3.2.1. Отримання та опрацювання даних з мікроконтролера NodeMCU.

Node-RED - візуальна система програмування для розробки потокових застосунків. Вона зазвичай використовується для створення автоматизованих задач та взаємодії між різними пристроями та службами [21].

У даному рішенні Node-RED забезпечує взаємодію між різними інтернет-пристроями та сервісами, використовуючи візуальне середовище та зв'язуючи різні "ноди" або блоки, які виконують поставлені завдання по заданому алгоритму роботи.

Програма, яка розроблена в застосунку Node-RED була розділена на 3 основні блоки, кожний із яких виконує завдання по певному алгоритму роботи і забезпечує отримання функціоналу, який демонструватиметься в подальшому.

Інстальоване програмне забезпечення Node-RED завантажено на ноутбук з ОС

Windows згідно інструкції на сайті виробника [22], але за необхідністю має можливість завантажуватись на інші ОС [23].

У інсталюваному програмному забезпеченні Node-RED було створено потік, який показаний на рис.3.14. Даний потік забезпечує отримання, відправку, опрацювання та візуалізацію даних, що він отримує від мікроконтролера NodeMCU у вигляді topic в подальшому опрацьовуються і далі надають візуалізацію на моніторі комп'ютера.

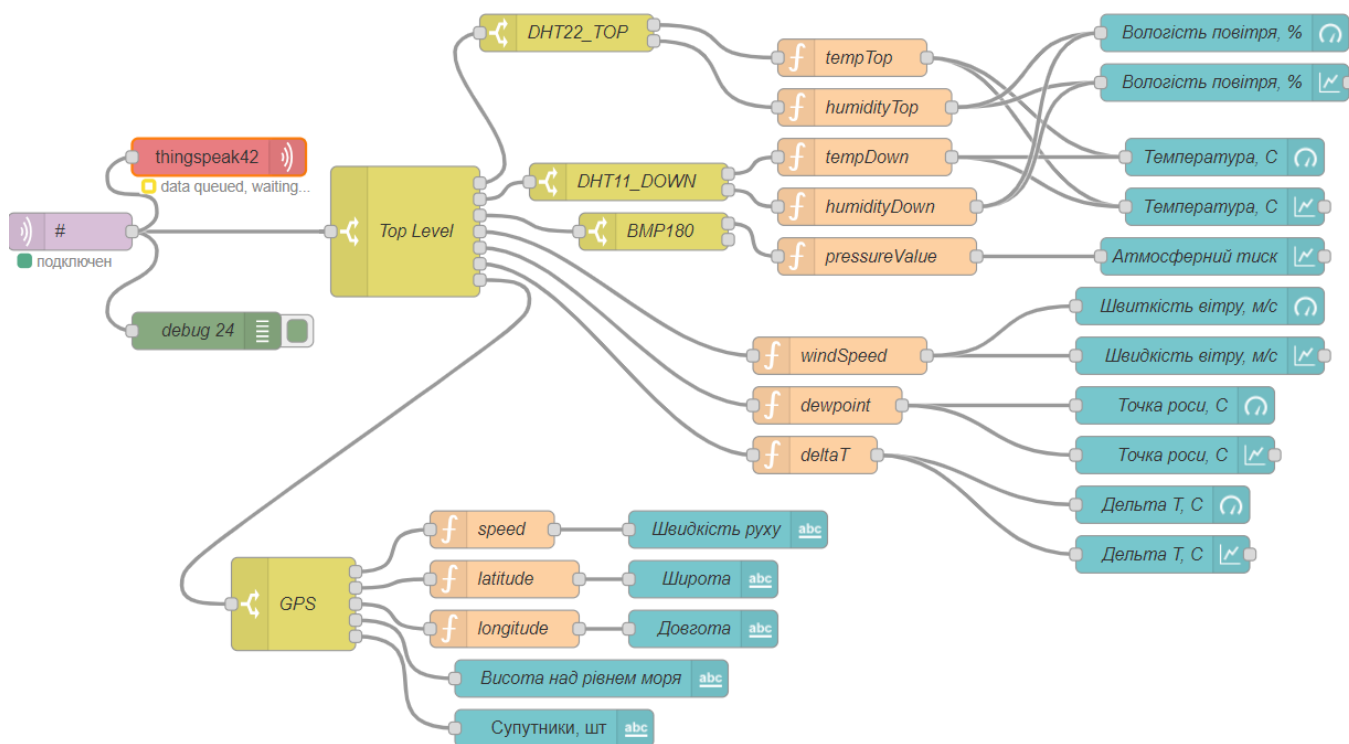


Рис.3.14. Потік отримання, відправки, опрацювання та візуалізації даних в платформі Node-RED

Оскільки вузли такі як DHT22_TOP, DHT11_DOWN, BMP180, GPS мають кілька схожих ознак у темах, то для них використовувались додаткові вузли switch, які розділили додатково основний потік.

З метою подальшого опрацювання даних для кожного елемента додано вузол функції, який отримує значення вологості з об'єкта msg.payload та зберігає його у змінній humidityTop (рис.3.15). Аналогічним чином присвоєна функція була для інших блоків.


```

var humidityTop =msg.payload;
flow.set("humidityTop", humidityTop);
return msg;

```

Рис.3.15. Лістинг коду для налаштування вузла function для елемента humidityTop

Далі данні передавались на вузол chart та gauge, які використовуються для візуалізації даних в дашбордах Node-RED.

Після проведення налаштувань потоків отримано графіки, які показані на рис.3.16. На указаних графіках в режимі реального часу передаються данні від давачів, які під'єднані до мікроконтролера ESP.

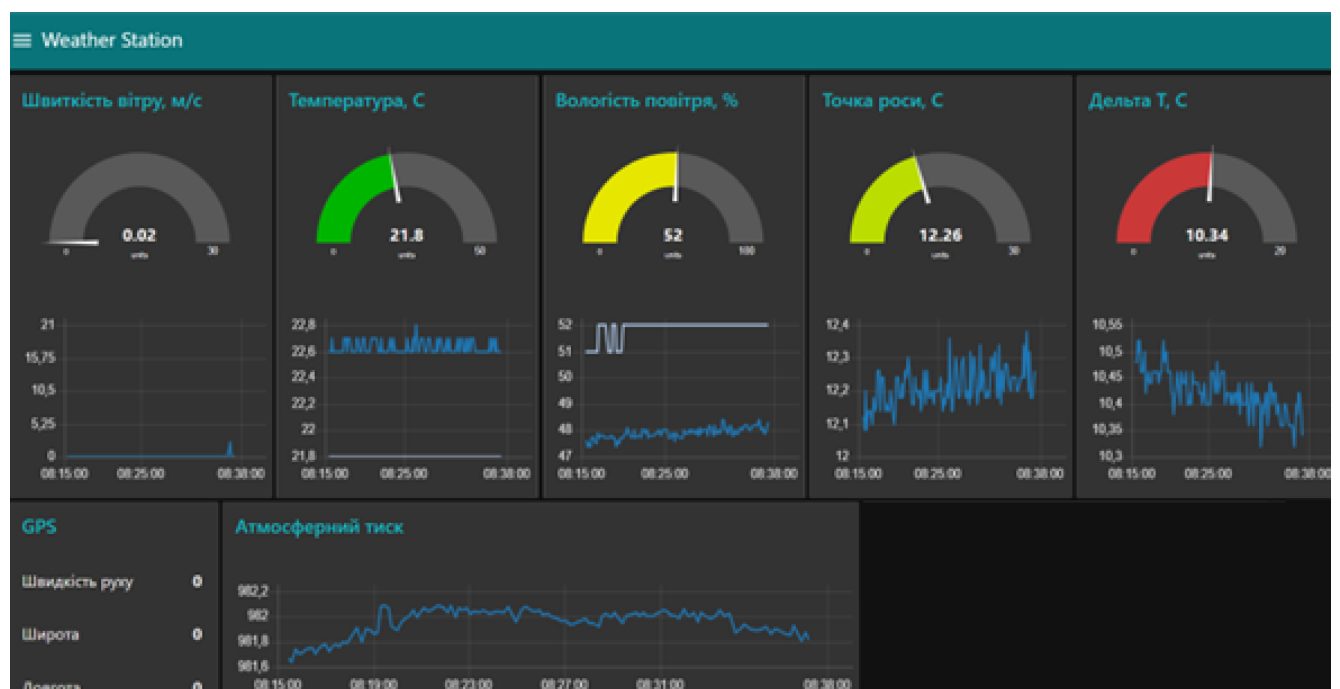


Рис.3.16. Візуалізація даних які отримуються від давачів.

3.2.2. Отримання прогнозу погоди в Node-RED.

З метою отримання прогнозу погоди відповідно до регіону, де буде працювати обприскувач, з даною метеостанцією було створено потік, який показаний на рис.3.17.

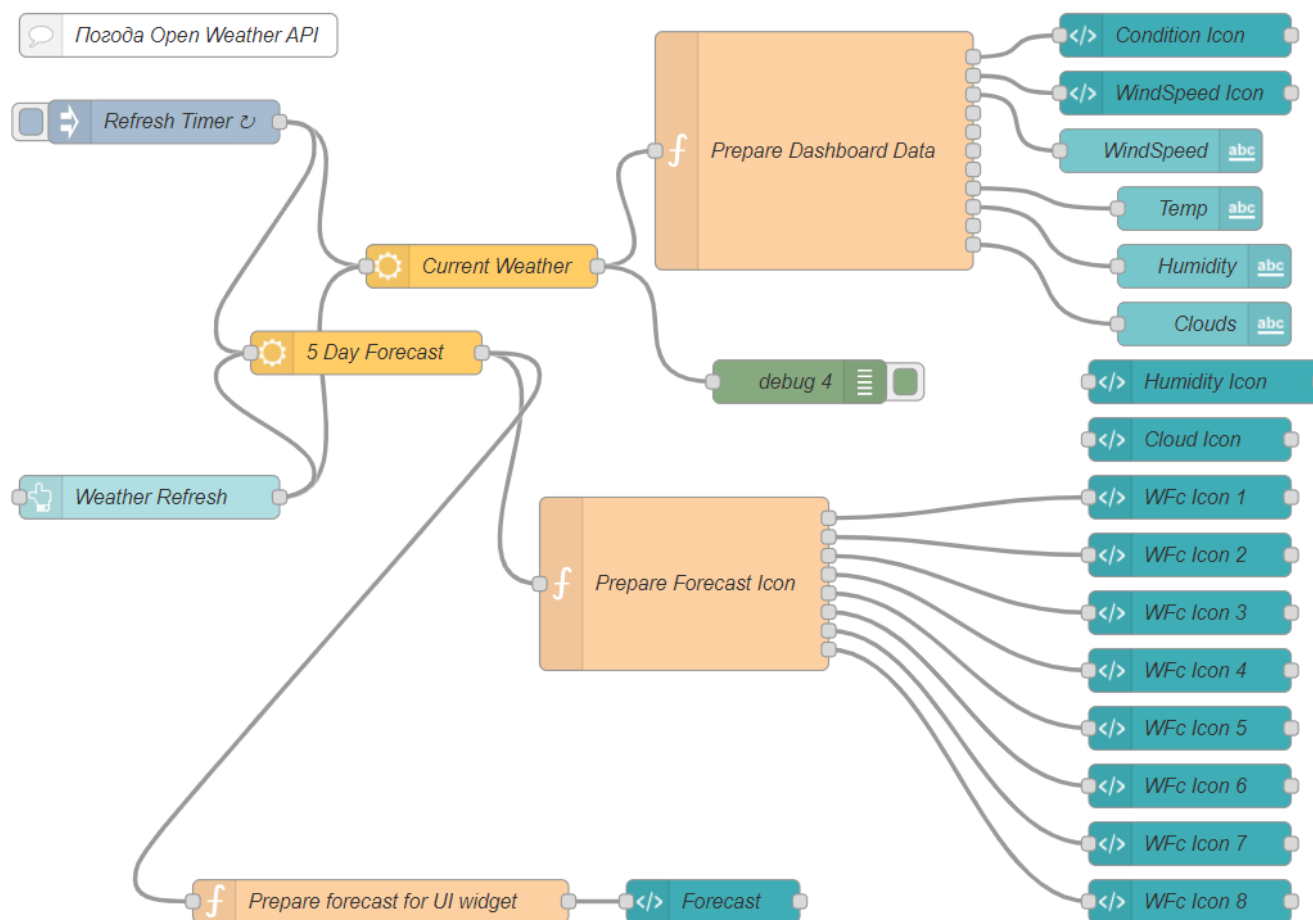


Рис.3.17. Потік прогнозу погоди на кілька годин по API

Для отримання прогнозу погоди є майже готове рішення в каталозі бібліотек node-red. Для того, щоб ним скористуватись спочатку потрібно завантажити бібліотеку node-red-node-openweathermap, яка дозволить отримати відповідну ноду.

Результат отриманого прогнозу погоди показано на рис.3.18.

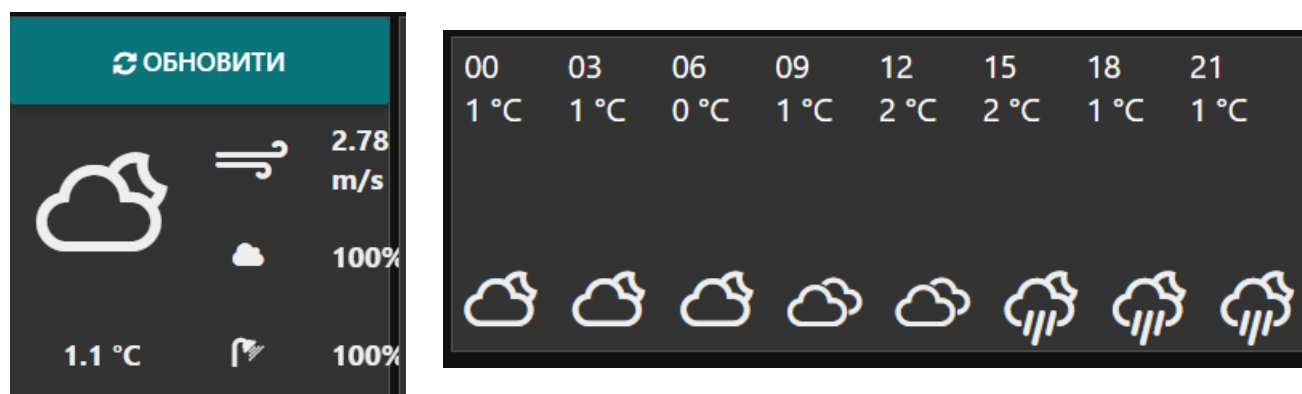


Рис.3.18. Результат отриманого прогнозу погоди в Node-RED

3.2.3. Створення Telegram bot в додатку Node-RED.

Основна мета створення даного рішення полягає в тому, щоб контролер, який слідує як оператор працює на обприскувачі, мав можливість комунікувати з оператором шляхом посиленням команд через Telegram bot, а також надавав рекомендації по налаштуванням обприскувача, якщо фактичні умови на полі змінились.

Повний потік Telegram bot в платформі Node-RED наведено на рис.3.19

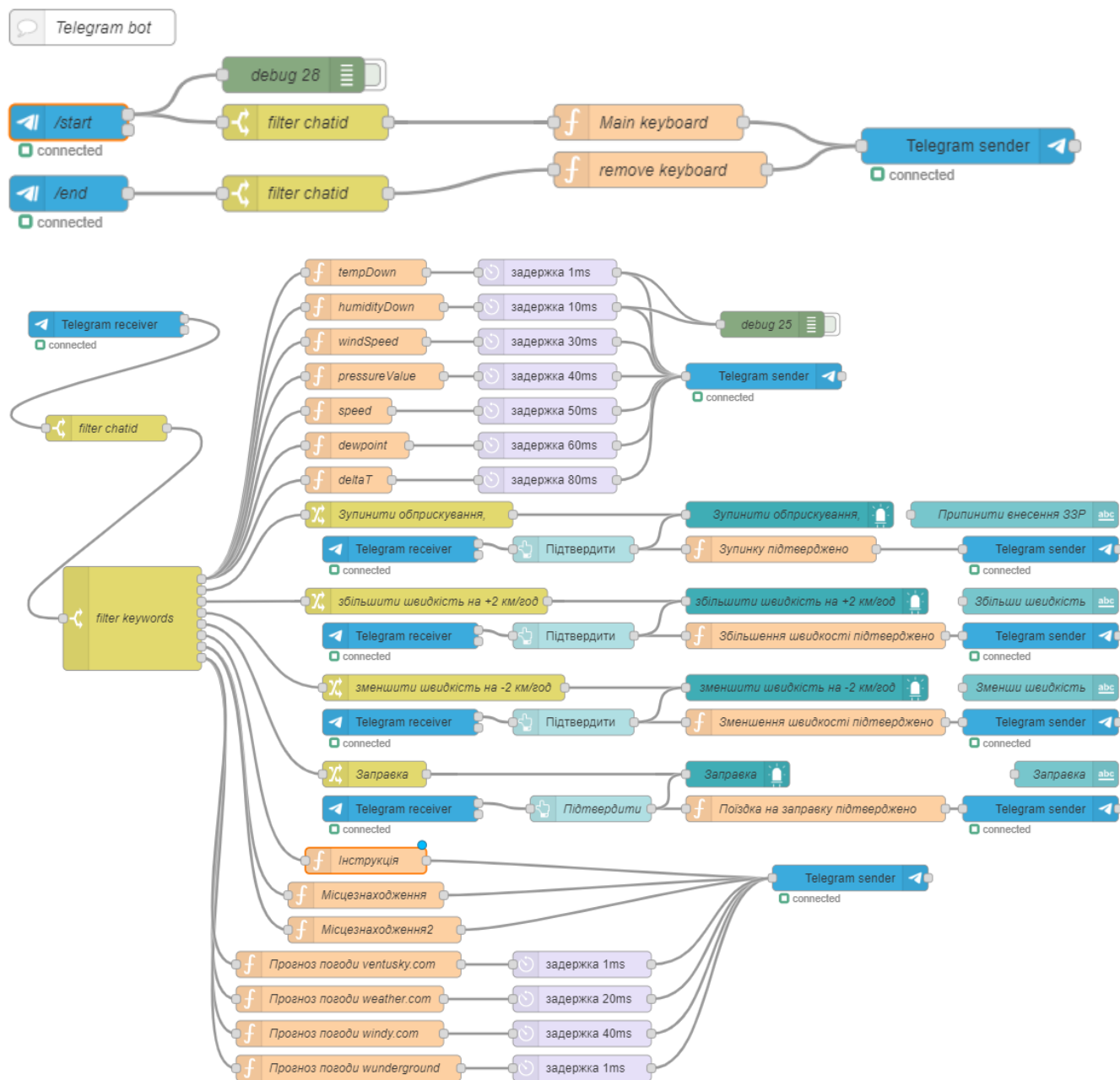


Рис.3.19. Повний потік Telegram bot в додатку Node-RED

Створення Telegram bot починається з того, що спочатку за допомогою BotFather згідно інструкцій на сайті виробника створюється Telegram bot. Під час процедури створення Telegram bot потрібно скопіювати данні такі як Token, Bot-Name, а також chatid. Далі в каталозі бібліотек Node-RED завантажується бібліотека node-red-contrib-telegrambot, яка забезпечує створення вузлів для створення telegram bot. Після цього додається блок під назвою command, у якому додаються основні параметри для створеного бота, а саме Token та Bot-Name, а далі - налаштовується сам вузол command. При цьому для того, щоб дана команда спрацювала в створеному Telegram bot потрібно вести команду /start

Наступний вузол використовувався блок switch який додає властивість для payload.chatId у даному випадку вказується фактичний номер чату до якого застосовуватиметься команда.

Далі команда подається на вузол функції з назвою Main keyboard. У даному блоці (рис.3.20) створюється клавіатура, яка в подальшому відобразиться в боті.

```

var options = {
  reply_markup: JSON.stringify({
    keyboard: [
      ['Параметри', 'Прогноз погоди'],
      ['Зупинити обприскування'],
      ['Збільшити швидкість на +2 км/год', 'Зменшити швидкість на -2 км/год'],
      ['Заправка', 'Місцезнаходження', 'Інструкції']
    ],
    resize_keyboard : true, //auto resize
    one_time_keyboard : false, //once pressed it will not hide
  })
};
var payload = {
  type: "message",
  content: "Start", //Caption of the keyboard
  chatId: msg.payload.chatId, //display keyboard back to the user that send request
  options: options, //refer to options above with keyboard
}
//var msg1 = {payload: payload, ReplyKeyboardRemove}
//a simple way to write is var msg1 = {payload: payload}
var msg1 = {
  payload: payload
}
return msg1 //return msg1 to next module/node

```

Рис.3.20. Лістинг коду блоку вузол функції з назвою Main keyboard

Після даної функції потік даних передається на вузол telegram sender, який

надсилає данні в бот. Для налаштування вузла потрібно тільки вибрати бот.

Аналогічним чином побудований потік для блоку команди /End. Дана команда застосовується для закриття клавіатури, якщо по якійсь із причин вона заважатиме огляду даних в боті. Функція `remove keyboard` наведена на рис. 3.21.

```
var options = {
  reply_markup: JSON.stringify({
    |   remove_keyboard: true //remove keyboard
  })
}
//prepare payload of output
var payload = {
  type: "message",
  content: "End", //caption of keyboard
  chatId: msg.payload.chatId, //display keyboard to the user send
  options: options, //refer to options above with keyboard
}
var msg1 = {payload: payload}
return msg1
```

Рис.3.21. Лістинг коду для налаштування функції `remove keyboard` для команди /End

Зовнішній вигляд створеної клавіатури наведений на рис.3.22. Для завантаження програми в бот використовується команда /start, яка і запустить потік на її створення. В подальшому дана клавіатура постійно буде відкриватись.

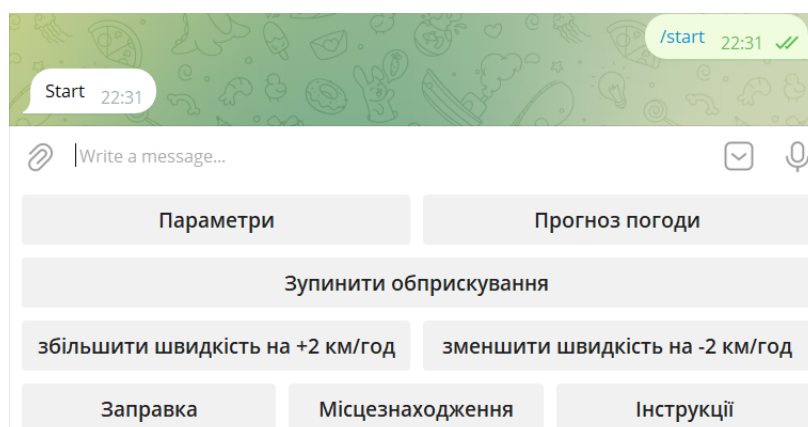


Рис.3.22. Створена клавіатура в телеграм боті.

Клавіатура створена, але вона наразі неактивна і потребує програмування. Блок приймання повідомлень `receiver` забезпечує прийом повідомлень від блоку

sender. Налаштування блоку receiver полягає тільки у виборі назви боту і його токена.

Після блоку receiver потік даних передається до блоку switch з ім'ям filter keywords (рис.3.23). При цьому дані, які передаються від натискання клавіатури, порівнюються з виставленими правилами і якщо є співпадіння посилає даний потік даних на наступний блок.

При натисканні кнопки “Параметри” після блоку switch з ім'ям filter keywords, дані передаються на потік “Параметри”, вузли з функціями, де зчитуються дані з датчиків.

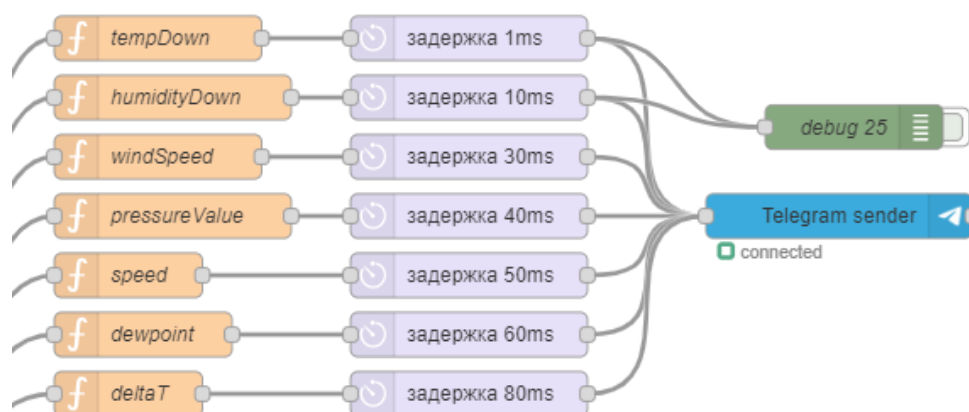


Рис.3.23. Налаштування вузлів функцій для зчитування даних з датчиків

Для структуривання отримання даних в бот застосовуються затримки.

Результат отримання даних після натискання кнопки “Параметри” наведено на рис.3.24.

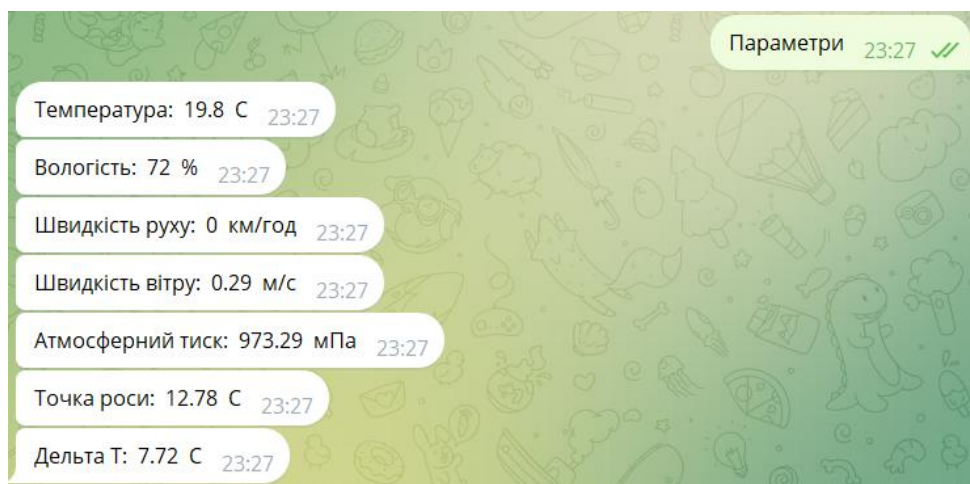


Рис.3.24. Отриманні данні в боті телеграм

При натисканні кнопки “Зупинити обприскування” дані передаються на вузол Change.

Даний вузол застосовується для присвоєння значення для `msg.payload` стану `true`. Указане присвоєння застосовується для включення індикатора на дашборді.

Якщо натиснута кнопка “Зупинити обприскування” на моніторі Node-red застосовується світлова індикація зеленого і червоного кольору (рис.3.25). При цьому червона індикація символізує для оператора обприскувача, що потрібно припинити обприскування, а зелена спрацьовує якщо натиснута кнопка «ПІДТВЕРДИТИ».

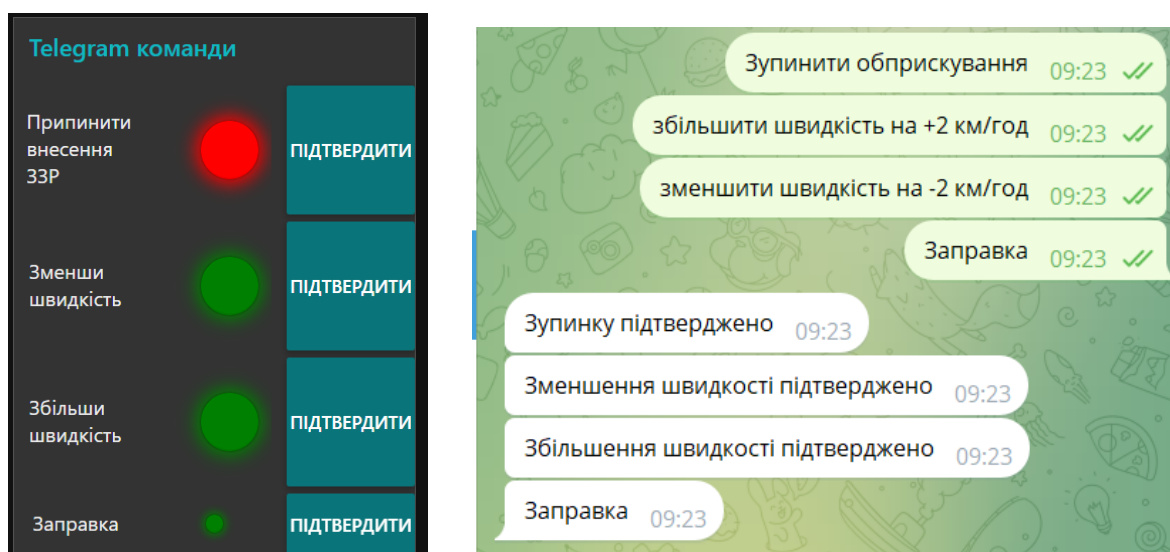


Рис.3.25. Команди від телеграм бота і візуалізація в Node-RED

Для отримання світлової індикації застосовувалась бібліотека `node-red-contrib-ui-led`. Для налаштування кнопок підтвердження застосовувалась нода “Button”.

Функція з ім'ям “Зупинку підтверджено” використовується для надсилання відповіді з дашборду про підтвердження виконання завдання в телеграм бот (рис.3.26).


```
// Ініціалізація об'єкта msg з властивостями
var msg = {
  payload: {}
};

msg.payload.content = "Зупинку підтверджено"
msg.payload.chatId = 722600317;
msg.payload.type = "message";

return msg;
```

Рис.3.26. Лістинг коду для налаштування функції з ім'ям “Зупинку підтверджено”

Аналогічним чином проводилось налаштування інших кнопок комунікації по телеграм для потоків, які показані на рис.3.27

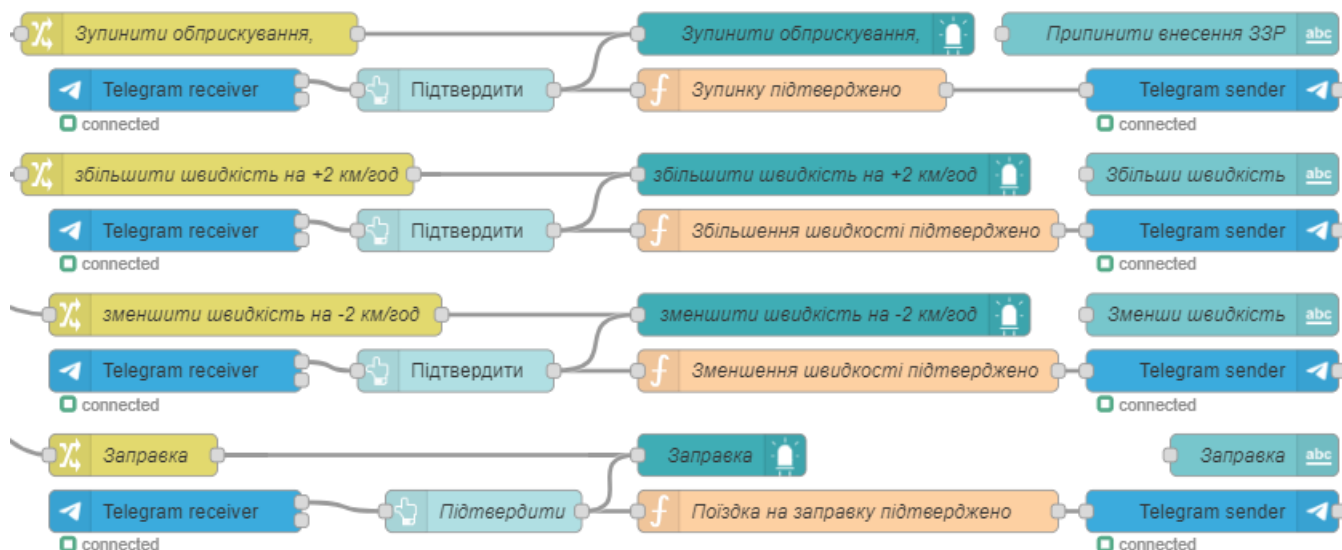


Рис.3.27. Потоки комунікації Node-RED та telegram bot для візуалізації натискання кнопок, а також отримання у відповідь зворотного зв'язку.

Візуалізація повного функціоналу в режимі реального часу, який бачитиме оператор обприскувача в Node-RED, показана на рис.3.28. При цьому незалежно, який монітор буде використовуватись, для візуалізації даних указані графіки можна з легкістю адаптувати під будь який монітор. Стандартний функціонал Node-RED це безпроблем дозволяє зробити.

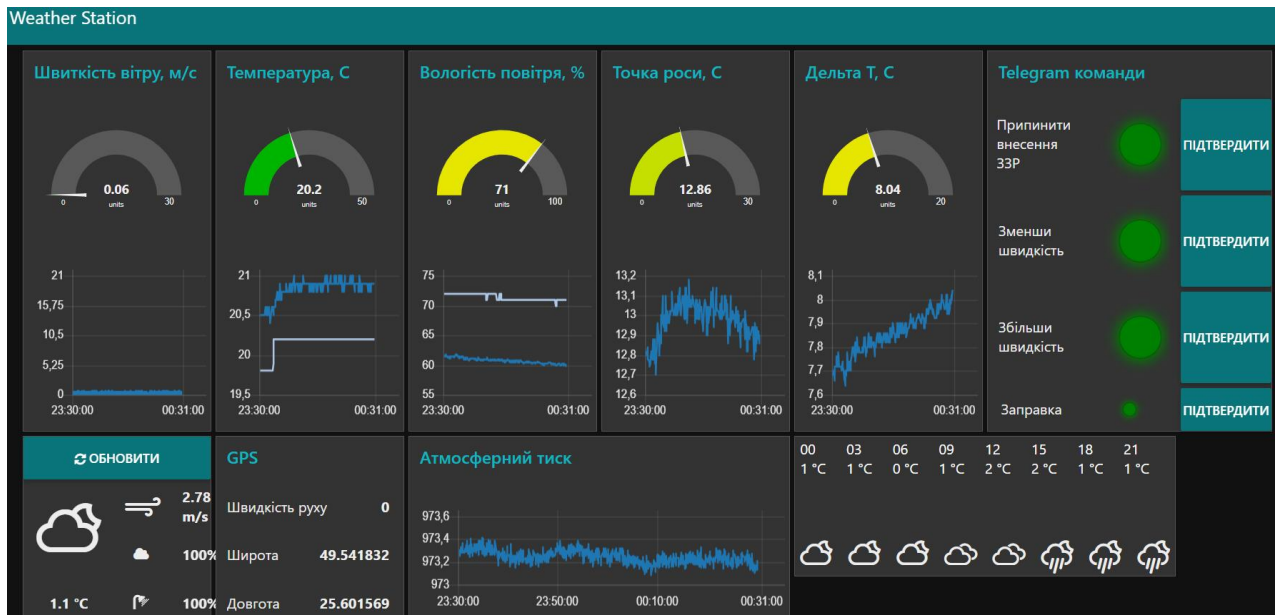


Рис.3.28. Візуалізація повного функціоналу який бачитиме оператор обприскувача в Node-RED

Наступна функціональна кнопка в телеграм боті “Прогноз погоди”.

Лістинг коду блоку функції, який забезпечує отримання прогнозу погоди наведено на рис.3.29.

```
var latitude = flow.get("latitude");
var longitude = flow.get("longitude");

// Створення посилання на прогноз погоди на Weather.com з новими координатами
var weatherForecastLink = "https://weather.com/weather/today/1/" + latitude + "," + longitude;

// Створення повідомлення для відправки в Telegram з посиланням на прогноз погоди
msg.payload.content = "Прогноз погоди: " + weatherForecastLink;
msg.payload.chatId = 722600317;

// Повернення повідомлення
return msg;
```

Рис.3.29. Лістинг коду для отримання прогнозу погоди по GPS

При натисканні на кнопку “Місцезнаходження” для її роботи застосовується потік, який показаний на рис.3.30.



Рис.3.30. Потік для отримання місцезнаходження обприскувача на полі

Лістинг коду блоку функції, який забезпечує отримання місцезнаходження наведено на рис.3.31.

```

var latitude = flow.get("latitude");
var longitude = flow.get("longitude");

// Функція для перетворення десяткових градусів у формат годин/хвилин/секунд
function convertDecimalToDMS(decimal) {
    var degrees = Math.floor(decimal);
    var minutesDecimal = (decimal - degrees) * 60;
    var minutes = Math.floor(minutesDecimal);
    var seconds = Math.round((minutesDecimal - minutes) * 60);
    return degrees + "°" + minutes + "'" + seconds + '"';
}

// Перетворення координат у формат годин/хвилин/секунд
var latitudeDMS = convertDecimalToDMS(latitude);
var longitudeDMS = convertDecimalToDMS(longitude);

// Створення посилання на Google Maps з новими координатами
var mapLink = "https://www.google.com/maps?q=" + latitude + "," + longitude;

// Створення повідомлення для відправки в Telegram з новими координатами
msg.payload.content = "Місцезнаходження: " + latitudeDMS + ", " + longitudeDMS + "\nПосилання на карту: " + mapLink;
msg.payload.chatId = 722600317;

// Повернення повідомлення
return msg;

```

Рис.3.31. Лістинг коду для отримання місцезнаходження по GPS

При натисканні на кнопку “Інструкція” для її роботи застосовується потік, який показаний на рис.3.32.



Рис.3.32. Потік для отримання інструкції

Лістинг коду блоку функції який забезпечує отримання інструкцій користувача наведено на рис.3.33.

```

msg.payload.content = "Інструкція користувача знаходиться за посиланням:
https://docs.google.com/document/d/1feVG5PdUfFmayNBdwQr1T2hMy1JXJ5LX/edit?usp=drive_link&oid=
109644396526812658162&rtpof=true&sd=true";
msg.payload.chatId = 722600317;
return msg;

```

Рис.3.33. Лістинг коду для інструкції користувача

3.3. Збір та аналіз отриманих даних

З метою збору та подальшого аналізу даних з NodeMCU використано рішення ThingSpeak.

Для забезпечення передачі даних використовувалась додаткова бібліотека `node-red-contrib-thingspeak42`, що забезпечує отримання блоку, який дозволить реалізувати передачу даних від MQTT до платформи ThingSpeak (рис. 3.34).

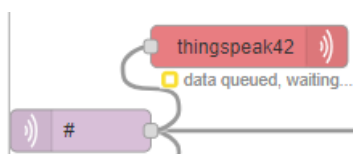


Рис.3.34. Потік для передачі даних з MQTT на ThingSpeak

Отриманий результат на платформі ThingSpeak після проведених налаштувань показаний на рис.3.35.

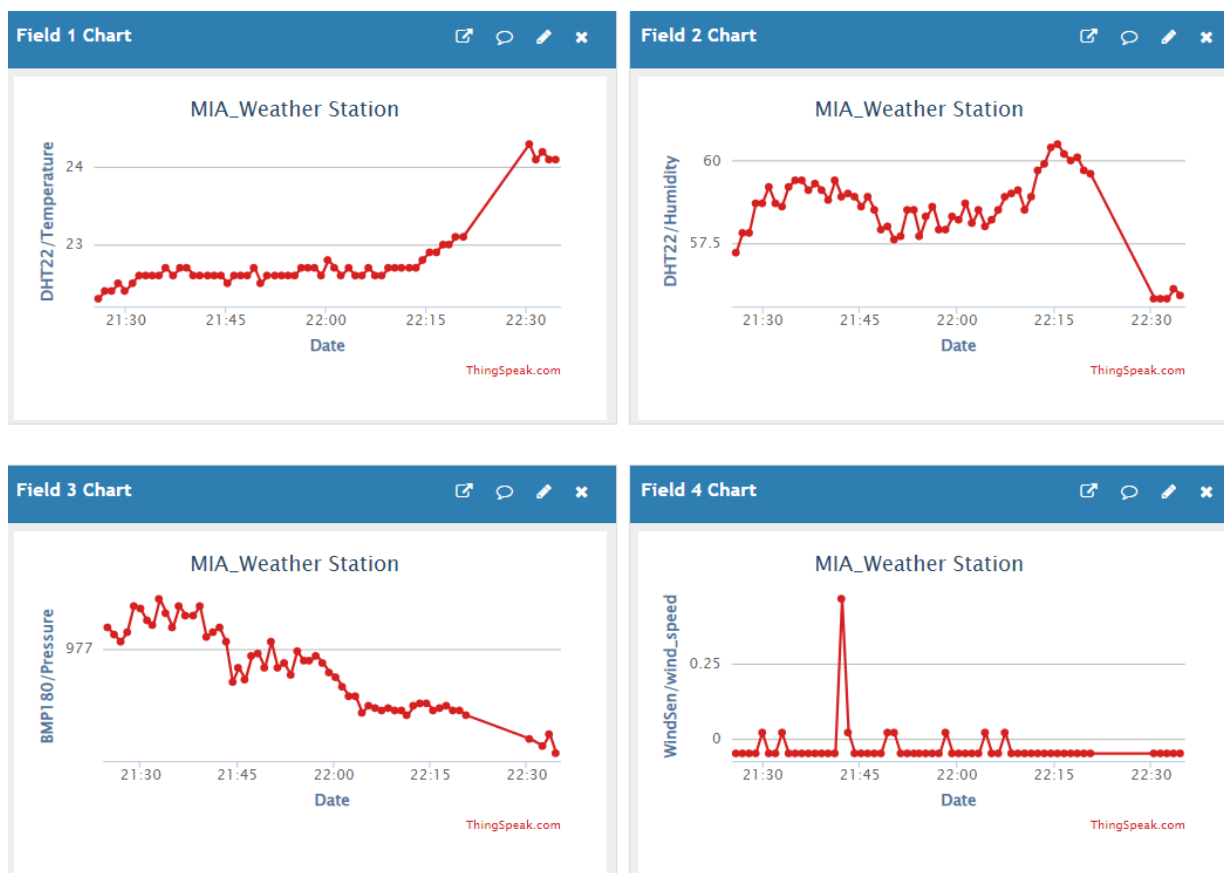


Рис.3.35. Отриманий результат на ThingSpeak

3.4. Тестування керуючого модуля мобільної метеостанції

Керуючий модуль мобільної метеостанції зображена на рис.3.36. Для коректного отримання даних, що стосується визначення інверсії повітря, давачі DHT-22 та DHT-11 мають розташовуватись на 2-х різних висотах, а саме DHT-11 має розташовуватись на висоті 1м від поверхні ґрунту або 0,5м від цільової поверхні обробки, інший давач DHT-22 монтується на висоті 3м.

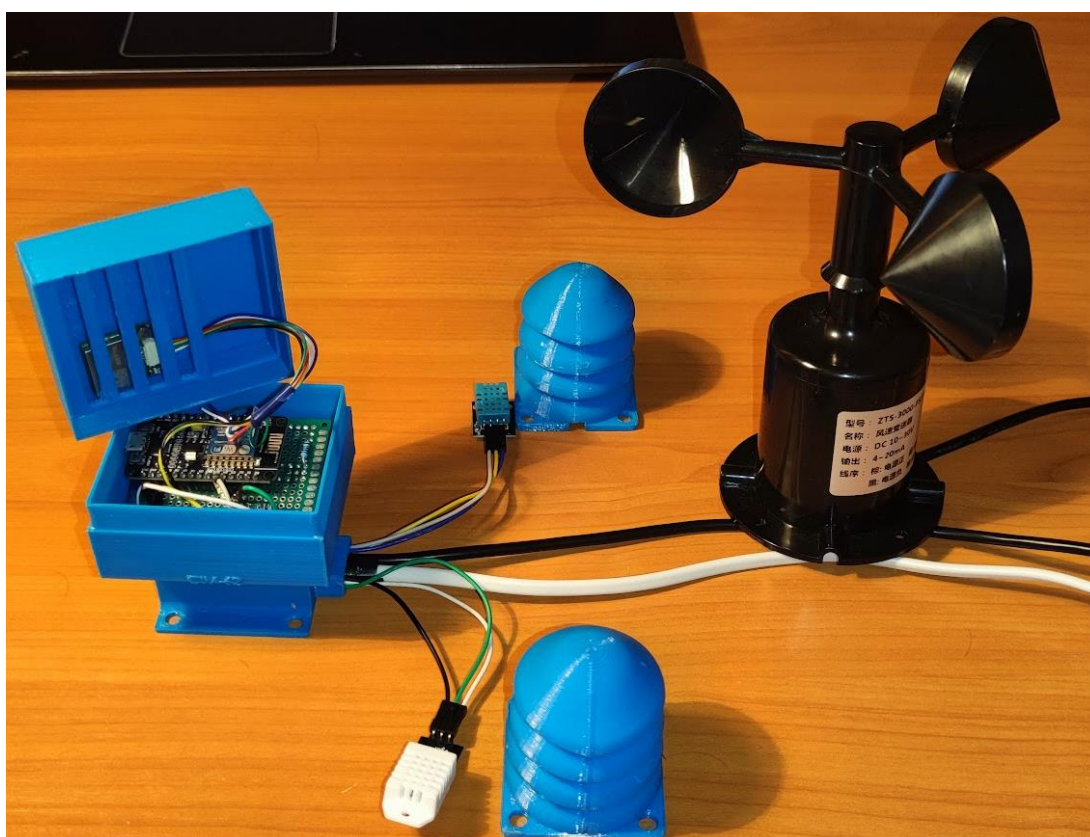


Рис.3.36. Розроблена метеостанція для обприскувача

Під час проведення тестування було встановлено, що за однакових умов, тобто коли давачі знаходяться на однаковій відстані при вимірі температури давачем DHT-22 і DHT-11, було виявлено перепад отриманих фактичних значень від $0,3^{\circ}\text{C}$ до $0,6^{\circ}\text{C}$, що досить суттєво для визначення інверсії. Також проводилась перевірка при застосуванні двох давачів DHT-11, при цьому було зафіксовані відхилення до $0,4^{\circ}\text{C}$, що також є суттєво, оскільки дані будуть некоректні. Тому у промисловому виконанні метеостанції потрібно

використовувати точніші давачі, для прикладу DHT-22, який цілком підходить своїм рівнем похибки вимірів температури у розмірі 0,5%.

Оскільки в кабіні обприскувача можуть бути використанні різні за розмірами монітори, а за їх відсутності смартфон, тому перевірялась можливість адаптації розробленої програмної частини під різні монітори. Візуалізація даних на різних монітора показана на рис.3.37.

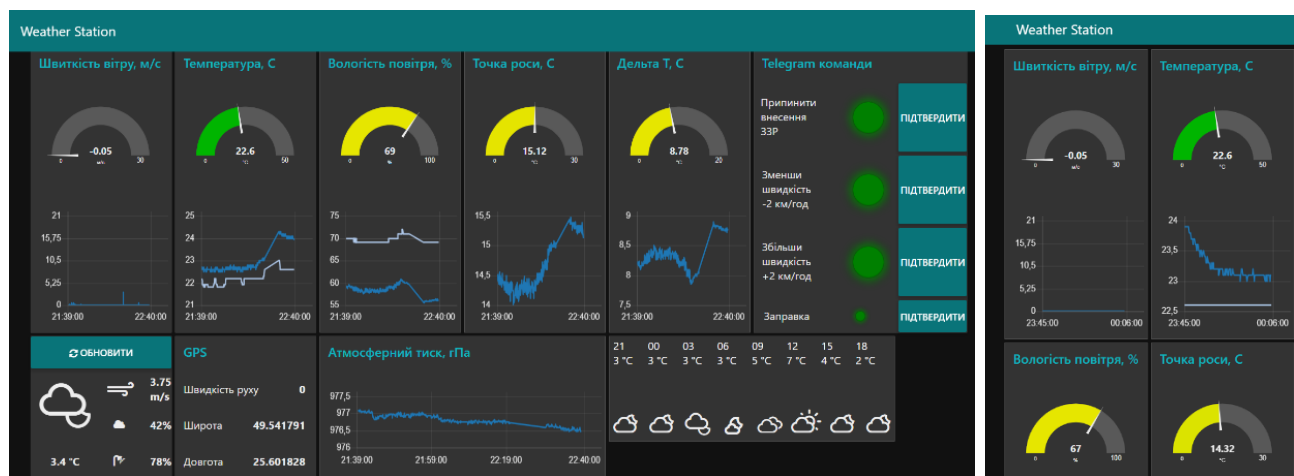


Рис.3.37. Візуалізація даних на монітор ПК та Смартфоні

В процесі тестування розробленого ПЗ для NodeMCU та передачі даних в Node-RED не було виявлено багів чи зависань програми. В майбутньому одним із елементів зменшення надлишковості даних в промисловому виконанні бажано затримки на отримання даних виставляти в розмірі 30с або кожну 1хв. Це зменшить навантаження на систему і, як наслідок, підвищить її надійність.

Також проводилось тестування розробленого телеграм бота, якого основне завдання забезпечити оперативне отримання даних агрономом чи контролером господарства про фактичні дані на полі, дані прогнозу погоди, а також, місцезнаходження обприскувача, він може швидко давати команди для оператора обприскувача. Зразок отримання фактичних даних по полю при натисканні на кнопку “Параметри” продемонстрована на рис.3.38.

В процесі тестування було виявлено, що потребує додаткового калібрування давача вітру, оскільки в стані спокою він показує від’ємні значення, тому було змінено константи для давача вітру.

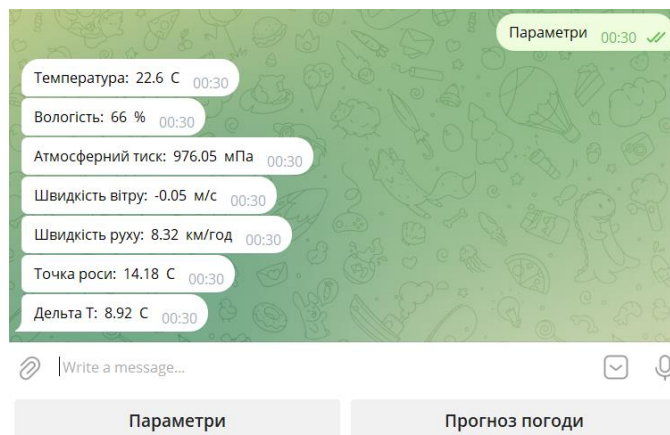


Рис.3.38. Отримані дані при натисканні кнопки “Параметри”

При тестуванні кнопки “Прогноз погоди” виявлено, що для сайтів weather.com, windy.com, wunderground.com для коректного показу погоди потрібно координати широти та довготи розділяти “;”, а для сайту ventusky.com використовувати “;”.

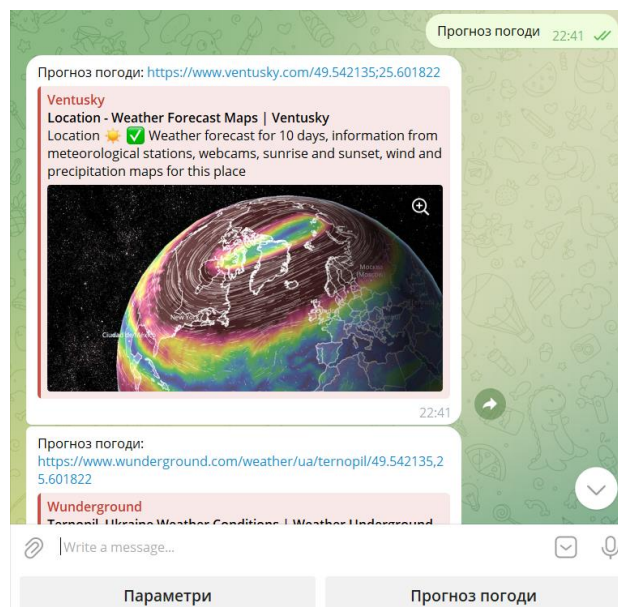


Рис.3.39. Отримані дані при натисканні кнопки “Прогноз погоди”

Тестуючи алгоритм роботи кнопок “Зупинити обприскування”, “збільшити швидкість +2км/год”, “зменшити швидкість -2км/год” та “Заправка” отримано позитивний результат при тестуванні. Зображення з процедури тестування наведені на рис.3.40 - 3.42.

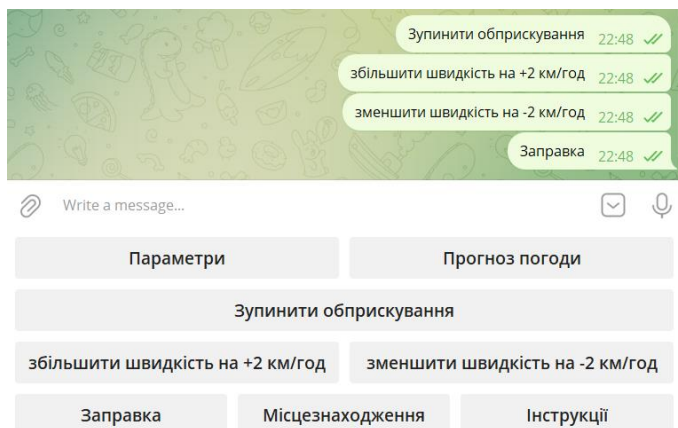


Рис.3.40. Повідомлення в боті про натискання на кнопки “Зупинити обприскування”, “збільшити швидкість +2км/год”, “зменшити швидкість - 2км/год” та “Заправка”

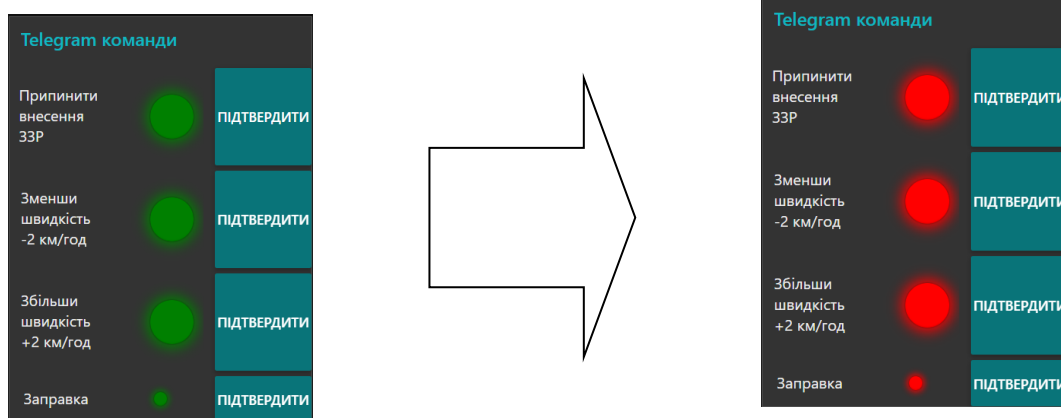


Рис.3.41. Зміна світлового індикатора кнопок “Зупинити обприскування”, “збільшити швидкість +2км/год”, “зменшити швидкість -2км/год” та “Заправка” при натисканні в Telegram bot

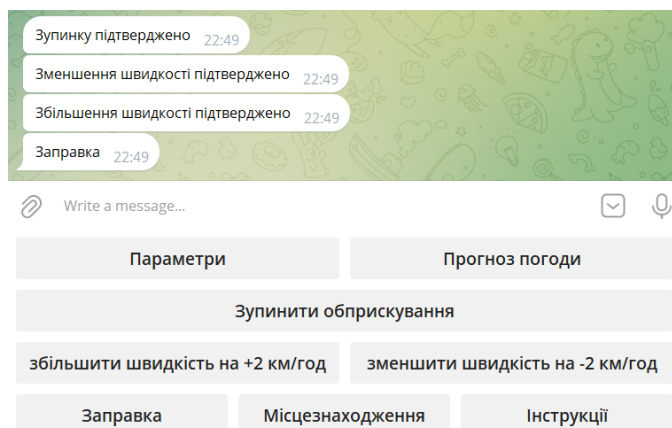


Рис.3.42. Надання відповіді при натисканні в дашбордах Node-red кнопок “ПІДТВЕРДИТИ”

При тестуванні кнопки “Місцезнаходження”, яка показана на рис.3.43. встановлено, що місцезнаходження визначається коректно, при цьому для десктопові версії комфортніше користуватись Google maps, а для мобільних пристороїв OpenStreetMap.

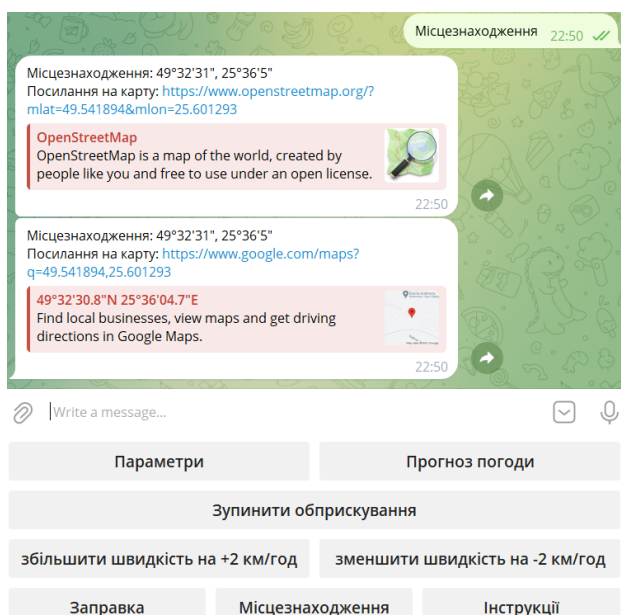


Рис.3.43. Отримані дані при натисканні кнопки “Місцезнаходження”

Також проводилось тестування для кнопки “Інструкції”, яка показала коректну роботу. Дані з тестування показані на рис.3.44.

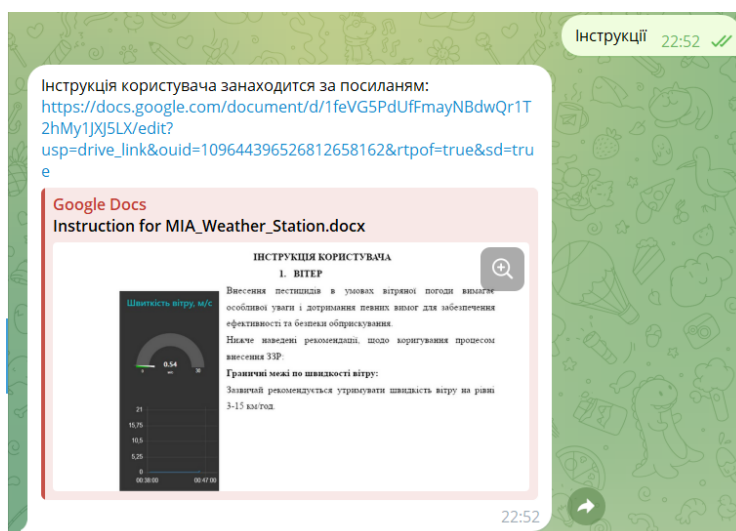


Рис.3.44. Отримані дані при натисканні кнопки “Інструкції”

В майбутньому планується додати функціонал передачі коротких текстових повідомлень на дашборд Node-red, а також отримання треку переміщень обприскувача на полі.

3.5. Висновки до третього розділу

Детально розписано основний програмний код для мікроконтролера NodeMCU обґрунтуванням вибраних методів та основних функцій.

В середовищі Node-RED розроблено потік, який забезпечує отримання, опрацювання та візуалізацію метеоданих з мікроконтролера ESP8266, а також розроблено потік, який забезпечує комунікацію Node-RED з телеграм ботом.

Розроблено телеграм бот, який дозволяє посилати команди з телеграм бота на Node-RED, а саме команди: “ Параметри ”, “Прогноз погоди”, “Зупинити обприскування”, “збільшити швидкість на +2 км/год”, “зменшити швидкість на -2 км/год”, “Заправка”, “Місцезнаходження” та “Інструкції” при цьому указані команди опрацьовуються по розробленому алгоритму і далі надають зворотній зв'язок для бота та дашборда Node-RED.

Проведено тестування розробленого керуючого модуля мобільної метеостанції для обприскувача, яка підтвердила ефективність застосованих алгоритмів, а також дозволила усунути неточності в отриманих даних.

РОЗДІЛ 4

ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1. Охорона праці

Мета кваліфікаційної роботи магістра полягає у розробці алгоритмічного та програмно-апаратного забезпечення керуючого модуля мобільної метеостанції для обприскувача. Під час розв'язання задач дослідження, враховано вимоги з охорони праці, пожежної та електробезпеки та техніки безпеки.

Основні документами які регулюють питання по охорони праці для користувачів ПК є :

1. НПАОП 0.00-7.15-18 «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями»;
2. ДБН В.2.5-56:2014 «Системи протипожежного захисту»;
3. Інструкція з охорони праці під час експлуатації електронно-обчислювальних машин, затверджена наказом Міністерства доходів і зборів України від 5 вересня 2013 р. № 443;
4. Вимоги до приміщень, згідно з [35, 36], щодо розташування робочого місця передбачають виконання наступних вимог: мінімальна площа, яка виділяється на одне робоче місце повинна становити мінімум $6,0 \text{ м}^2$, при об'ємі – мінімум $20,0 \text{ м}^3$; – розташування робочих місць користувачів ПК заборонено у цокольних або підвальних приміщеннях.

Виконання теоретичної та практичної частини проекту передбачало використання комп'ютерної техніки та обладнання з низькими напругами та силами струму. Зокрема, для блоку живлення плати ESP8266 використовувалася напруга живлення 5 В, при цьому на платі використовувались можливі номінали напруги на рівні 5 В і 3,3 В, що не представляло небезпеки для користувачів та розробника системи.

Для регламентації пожежної безпеки використано вимоги "Типового

положення про інструктажі, спеціальне навчання та перевірку знань з питань пожежної безпеки на підприємствах, в установах та організаціях України", що затверджене постановою Кабінету міністрів України від 26 червня 2013 р. № 444. Для захисту від негативного впливу екранів дотримано вимог Закону України "Про затвердження Вимог щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями" та НПАОП 0.00-7.15-18, затвердженого наказом Міністерства соціальної політики України 14.02.2018 N207.

Робоче місце виконання кваліфікаційної роботи та проектування комп'ютерної системи відповідає вимогам організаційних, ергономічних та пожежної безпеки, враховуючи регламентацію електробезпеки відповідно до Правил безпечної експлуатації електроустановок споживачів (НПАОП 40.1-1.21-98) від 09.01.98 та інших визначених стандартів [26, 27].

Вимоги до електробезпеки при проектуванні компонентів комп'ютерної системи для моніторингу метеорологічних параметрів дотримано, використовуючи безпроводні технології передавання даних і напругу живлення в діапазоні 3,3 В і 5 В. Це спрямовано на зменшення ризику ураження струмом при контакті з мережею чи в аварійних ситуаціях.

Щодо пожежної безпеки будівлі, використання і експлуатації приміщення відповідає вимогам державних будівельних норм "Пожежна безпека об'єктів будівництва" (наказ Держбуду України від 03.12.2002 N 88) та правил пожежної безпеки України (наказ Міністерства України з питань надзвичайних ситуацій від 19.10.2004 N 126).

Приміщення, де розташовані робочі місця з ПК, відповідає санітарним нормам ДСанПіН 3.3.2-007-98. Забезпечено оптимальні значення параметрів мікроклімату, враховуючи вимоги нормативних документів.

Освітлення в приміщенні з ПК відповідає ДБН В.2.5-28-2018, забезпечуючи як природне, так і штучне освітлення згідно з встановленими нормами. Штучне освітлення виконано за допомогою системи загального освітлення з використанням люмінесцентних ламп типу ЛБ.

При розробці проекту комп'ютерної системи для моніторингу метеорологічних параметрів на основі IoT дотримано вимог щодо техніки безпеки: не виконувався самостійний ремонт ПК, не вносились конструктивні зміни в апаратне забезпечення, використовувалися лише матеріали, що стосувалися розробки системи.

Для забезпечення безпечної експлуатації інформаційних технологій та мереж дотримано вимог СТУ EN 60950-1:2015 "Обладнання інформаційних технологій. Безпека. Частина 1. Загальні вимоги".

Нормативно-правові акти у сфері захисту рослин [31]:

1. Закон України «Про захист рослин»;
2. Закон України «Про пестициди і агрохімікати»;
3. Закон України «Про основні засади державного нагляду (контролю) у сфері господарської діяльності»;
4. Постанова Кабінету Міністрів України від 9 травня 2023 р. № 458 «Про затвердження Порядку одержання посвідчення про право роботи з пестицидами»;
5. Постанова КМУ від 2 листопада 1995 р. № 881 «Про затвердження Порядку державного обліку наявності та використання пестицидів і агрохімікатів»;
6. Постанова Кабінету Міністрів України від 23 січня 2019 р. № 34 «Про затвердження критеріїв, за якими оцінюється ступінь ризику від провадження господарської діяльності у сфері захисту рослин та визначається періодичність здійснення планових заходів державного нагляду (контролю) Державною службою з питань безпечності харчових продуктів та захисту споживачів»;
7. Наказ Міністерства аграрної політики та продовольства України від 07.05.2019 № 242 «Про затвердження уніфікованих форм актів, що складаються за результатами проведення планових (позапланових) заходів державного нагляду (контролю) щодо додержання суб'єктом господарювання вимог законодавства у сферах карантину та захисту рослин»;

8. Наказ Міністерства аграрної політики та продовольства України від 06.09.2013 № 534 «Про затвердження Порядку оформлення Головним державним фітосанітарним інспектором України, головними державними фітосанітарними інспекторами а Автономній Республіці Крим, місті Києві, областях, їх заступниками та державними фітосанітарними інспекторами матеріалів про адміністративні правопорушення»;

9. Наказ Міністерства аграрної політики та продовольства України від 13.02.2013 № 96 «Про затвердження розмірів плати за послуги з питань ветеринарної медицини, захисту рослин, охорони прав на сорти рослин, які надаються органами та установами, що входять до сфери управління Державної ветеринарної та фітосанітарної служби України».

4.2. Фактори виробничого середовища і їх вплив на життєдіяльність промислово-виробничого персоналу

Термін «виробниче середовище» - комплекс організаційно-технічних параметрів, а також архітектурно-конструктивних рішень, метою яких є створення комфортних умов праці персоналу та забезпечення стабільної роботи обладнання шляхом раціонального поєднання робіт операції з роботою машин і пристроїв [32]. Виробниче середовище не має прямого впливу на виробничі процеси, але воно сильно впливає на працездатність персоналу, а також має знімати втому та запобігати захворюванням.

Технологічний процес зберігання та обробки інформації вимагає високої культури виробництва та особливої чистоти повітря у виробничому середовищі. Водночас пил і гази негативно впливають на точність і надійність електронного обладнання та якість технічних пристроїв.

Підприємства з обробки даних потребують захисту від пилу, надійної звукоізоляції між виробничими приміщеннями, а також ідеальних санітарно-гігієнічних умов (мікроклімат, освітлення, опалення, вентиляція тощо).

Джерелами шуму при цьому виді роботи є самі комп'ютери (вентилятори в стійках, принтери тощо), пристрої вентиляції та кондиціонування та інше обладнання.

У виробничих приміщеннях, де робота здійснюється за допомогою відеотерміналів (ВДТ) і персональних комп'ютерів, рівень шуму на робочих місцях не повинен перевищувати значень, встановлених санітарними нормами щодо допустимих рівнів шуму для таких робіт.

При виконанні основних робіт на ВДТ і ПК (диспетчерська, операторська, лічильні кабінки і контрольні пункти, комп'ютерна техніка та ін.) рівень шуму на робочому місці не повинен перевищувати 50 дБА.

Рівень шуму не повинен перевищувати 60 дБА в середовищі, де інженери та техніки виконують лабораторний, аналітичний або вимірювальний контроль.

У приміщеннях, де знаходяться комп'ютери (без екранів), рівень шуму не повинен перевищувати 65 дБА. Рівень шуму на робочих місцях у приміщеннях з шумними блоками ЕОМ відповідає СанПіН 2.2.2. 542-96 не має перевищувати 75 дБА.

Наступні фактори які можуть негативно вплинути на людину, є параметри мікроклімату. Параметрами мікроклімату виробничого середовища є сукупність температури, відносної вологості та швидкості руху повітря.

Дані параметри суттєво впливають на функціональну активність людини, її самопочуття, здоров'я, а також на надійність комп'ютерної техніки.

Великий вплив на мікроклімат мають джерела тепла. До них відносяться: комп'ютерна техніка, освітлювальні прилади, обслуговуючий персонал і сонячна радіація. Приміщення з великою кількістю комп'ютерів мають найбільше сумарне тепловиділення, а основним опалювальним обладнанням у цих приміщеннях є комп'ютери, які в середньому виділяють 80% загального тепла. Тепловиділення від освітлювальних приладів становить 12%, від обслуговуючого персоналу 1%, від сонячного випромінювання 6%. Тепловий потік через непрозорі захисні конструкції становить 1%.

Відносна вологість повітря має великий вплив на організм людини і роботу обладнання. За вологості повітря до 40% основа магнітної стрічки стає

крихкою, підвищується знос на магнітних головок, погіршується ізоляція на кабелях, виникає статична напруга при переміщенні комп'ютерного середовища.

Кількість пилу в приміщеннях комп'ютерної техніки не повинна перевищувати $0,5 \text{ мг/м}^3$. Забороняється палити, оскільки частинки попелу, що осідають на поверхні магнітних носіїв, можуть призвести до несправності комп'ютера.

Обладнання, рівень шуму якого перевищує норматив, необхідно встановлювати поза приміщенням, де знаходяться ВДТ і ЕОМ. Шум у машинних відділеннях знижується за рахунок зменшення шуму від джерел і спеціальних архітектурно-конструкційних рішень.

Додаткові заходи зниження шуму в машинних відділеннях можуть включати:

- пристрій підвісної стелі, що виконує роль звукопоглинаючого екрана;
- використання матеріалів, які максимально поглинають звук
- зменшення площі скління та віконних прорізів;
- встановлення особливо шумних пристроїв на еластичні килимки;
- використання на робочих місцях звукопоглинальних екранів;
- Використання простих штор з щільної тканини, які складаються в складки і розвішуються на відстані 15-20 см від огорожі. Ширина штори повинна бути в два рази більше ширини вікна.

4.3. Забезпечення електробезпеки користувачів ПК.

Комплексним нормативним документом, спрямованим на забезпечення безпеки користувачів ПК, є «Державні санітарні правила і норми роботи з візуальними терміналами (ВДТ) електронно-обчислювальних машин».

Особлива увага приділяється тому, щоб знизити вплив шкідливих і небезпечних факторів на працівників, які працюють з відеотерміналами. Положення визначають гігієнічні та ергономічні вимоги до організації робочих місць і приміщень, параметри виробничого середовища, що дають змогу

запобігти порушенням здоров'я окремих користувачів комп'ютерів. Користувачі ПК працюють з пристроями, які постійно підключені до джерела живлення.

При організації робочого місця необхідно ретельно перевірити безпеку підключення до електричної мережі. При роботі з комп'ютерною технікою користувачі можуть зіткнутися з такими видами ризиків [33]:

- енергетичний ризик (спричинений коротким замиканням: опіки, електрична дуга, розряд розплавленого металу);
- ураження електричним струмом;
- термічний ризик (вплив високої температури);
- пожежна небезпека;
- механічний ризик (удари рухомих частин, падіння, травми внаслідок розрізання гострих частин елементів конструкції);
- радіаційний ризик (звукове, височастотне, інфрачервоне, ультрафіолетове та іонізуюче випромінювання, а також високоінтенсивне когерентне видиме світло (лазерне випромінювання) вплив);

Під час експлуатації електрообладнання можливе порушення цілісності ізоляції проводів, кабелів, обмоток машин та інших частин, що знаходяться під напругою струмоведучих частин, тобто струмоведучих частин. зменшується. Це неминуче викликає напругу в нерухомих частинах обладнання, внаслідок чого на працівника діє струм.

Електробезпека працівників служби забезпечується системою організаційно-технічних заходів і процедур. Одним із методів особистого захисту при спробі самоізоляції є використання захисної основи.

Коротке замикання на землю – це випадкове електричне з'єднання з електрообладнанням, яке знаходиться під напругою. При падінні обірваного дроту на землю, при порушенні ізоляції обладнання може виникнути коротке замикання на землю через появу зв'язку між струмоведучими частинами і корпусом заземлення або конструктивними частинами обладнання. У всіх цих випадках струм від струмоведучих частин передається на землю через контактні з землею електроди. Спеціальні металеві електроди називають масами.

Захисне заземлення – це навмисне з'єднання із землею за допомогою

заземлювального пристрою металевих невідповідних частин обладнання, які можуть опинитися під напругою внаслідок порушення ізоляції електроустановки [33].

Відповідно до СанПіН 2.2.2/2.4.1340-03 «Гігієнічні вимоги до персональних електронно-обчислювальних машин та організації роботи» будівлі, в яких розміщені робочі місця з ЕОМ, повинні мати безпечне заземлення (нуль) відповідно до технічних вимог. Експлуатація електроустановок та комп'ютерної техніки.

Комп'ютер зазвичай підключається через блок живлення або блок живлення, який має мережевий фільтр, його конденсатори призначені для замикання нульового проводу, а також триходові, відповідні високочастотним каналам перешкод. Наземна електромережа. У цьому випадку до розетки необхідно підключити три дроти: один фазний, другий нульовий робочий і третій нульовий провід (НЗП). Нульовий захисний провід повинен бути з'єднаний з нульовим провідником мережі.

В іншому випадку (якщо НЗП нікуди не підключається) в корпусі системного блоку може з'явитися напруга близько 110 АС. Це відбувається тому, що конденсатори фільтра діють як ємнісні ділянки напруги [33].

У подальшому при використанні персонального комп'ютера слід дотримуватися наступних рекомендацій [34]:

- постійно контролювати надійність підключення трипровідних розеток;
 - додатково підключити системний блок до НЗП, наприклад підключити провід під гвинт блоку живлення;
 - Рекомендується підключати екран через відповідний пристрій (якщо є тільки однофазна двопровідна мережа). При цьому мережевий фільтр і всі силові кабелі повинні знаходитися якомога далі від оператора в компактному положенні позаду робочого місця;
 - Не підключайте корпус комп'ютера до парового або водяного нагріву.
- Якщо блок живлення комп'ютера несправний, батареї можуть бути під напругою;
- Не розміщуйте центральний блок у дуже вологому або запиленому місці,

на підлозі, близько до ніг оператора;

- Не торкайтеся екрана монітора та клавіатури одночасно (можливий високий електростатичний потенціал);

- щоб уникнути ураження електричним струмом, забороняється торкатися задньої панелі системного блоку та змінювати роз'єми працюючої периферії комп'ютера;

- ПК повинен бути встановлений тільки на жорстку стійку опору, яка також виключає випадкові удари центрального блоку;

- Не рекомендується встановлювати ПК та його клавіатуру на поверхні, де накопичується статична електрика (органічне скло та поліровані лакові поверхні);

- температура повітря в приміщенні допускається в межах 20-25 градусів Цельсія, вологість до 75%; Неприпустимі різкі перепади температури;

- не допускається надмірна запиленість повітря приміщення (не більше 1 мг/м³ при максимальному розмірі частинок 3 мкм); обов'язкове щоденне вологе прибирання приміщень;

- Екран, фільтри біля екрана, клавіатуру та інші частини персонального комп'ютера необхідно щодня протирати вологою ганчіркою [34].

4.5. Висновки до четвертого розділу

В даному розділі описані питання щодо забезпечення безпеки в надзвичайних ситуаціях та охорони праці. Опрацьована інформація, що до основних вимог до приміщень та робочих місць з ПК з метою забезпечення комфортних умов праці.

Наведені основні нормативно-правові документи у сфері захисту рослин.

Надано відповідь, що до основних факторів виробничого середовища і їх вплив на життєдіяльність промислово-виробничого персоналу, а також розкрито питання, що до забезпечення електробезпеки користувачів ПК.

ВИСНОВКИ

У кваліфікаційній роботі розв'язана актуальна науково-технічна задача, яка полягала у підвищенні ефективності спрацювання пестицидів шляхом розробки алгоритмічного та програмно-апаратного забезпечення керуючого модуля мобільної метеостанції для обприскувача. В процесі виконання отримано наступні результати:

1. Проведено аналіз основних факторів, які впливають на ефективність внесення пестицидів із яких було виявлено, що найбільший вплив на якість спрацювання пестицидів мають метеоумови на полі, а саме: відносна вологість повітря, температура повітря, швидкість вітру, кількість опадів, дельта Т та температурні інверсії.

2. В результаті аналізу ринку встановлено, що є потреба у доступних за ціною мобільних метеостанцій для обприскувачів, які матимуть можливість оперативно надавати рекомендації для оператора обприскувача.

3. Розроблено загальну архітектуру комп'ютеризованої системи, яка дозволяє покращити якість внесення пестицидів завдяки отриманню прогнозованих та фактичних метеорологічних даних на полі обробки та реалізації оперативної комунікації між агрономом та механізатором.

4. Розроблено алгоритмічне та програмне забезпечення керуючого модуля мобільної метеостанції для обприскувача.

5. Розроблено телеграм бот, який дозволяє отримати метеорологічні параметри на полі обробки і на основі отриманих даних скоректувати налаштування обприскувача для забезпечення найбільшої ефективності обприскування.

6. Розроблено прототип керуючого модуля мобільної метеостанції, а також проведено його тестування.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Гузь М., Сінько В., Марченко В. Обприскування: на що слід звернути особливу увагу. 2017. URL: <https://agroexpert.ua/obpriskuvanna-na-so-slid-zvernuti-osoblivu-uvagu> (дата звернення: 24.09.2023).
2. Сторчоус І. Швидкість чи якість обприскування? – Більше не доведеться вибирати. 2023. URL: <https://www.growhow.in.ua/shvydkist-chy-iakest-obpryskuvannia-bilshe-ne-dovedetsia-vybyraty/> (дата звернення: 24.09.2023).
3. Гончаров О. БІЛЬША КРАПЛЯ – МЕНШЕ ЗНЕСЕННЯ. 2020. URL: <https://agroexpert.ua/bilsha-kraplia-menshe-znesennia/> (дата звернення: 24.09.2023).
4. ОПТИМАЛЬНІ ТЕМПЕРАТУРИ ДЛЯ ЗАСТОСУВАННЯ ПЕСТИЦИДІВ. 2020. URL: <https://posivna.com.ua/ua/zamitky-ahronoma/optimalni-temperaturi-dlya-zastosuvannya-pestitsidiv> (дата звернення: 24.11.2023).
5. John Deere Mobile Weather. 2023. URL: <https://www.deere.com/en/technology-products/precision-ag-technology/field-and-water-management/mobile-weather/> (дата звернення: 24.09.2023).
6. Wolf.T. Spraying Weather. 2022. URL: <https://sprayers101.com/spraying-weather> (дата звернення: 24.09.2023).
7. WatchDog Sprayer Station. 2023. URL: <https://www.specmeters.com/weather-monitoring/weather-stations/sprayer-station/watchdog-sprayer-station/> (дата звернення: 24.10.2023).
8. ПРОФЕСІЙНА МЕТЕОСТАНЦІЯ (WIFI) MISOL HP2550. 2023. URL: <https://thermolab.net.ua/ua/p1634319646-professionalnaya-meteostantsiya-wifi.html> (дата звернення: 24.10.2023).
9. МЕТЕОТРЕК МОБІ. 2023. URL: <https://www.meteotrek.ua/equipment/tproduct/353440524-366263547451-meteotrek-mobi> (дата звернення: 24.09.2023).
10. Шелест М.С., Комісар Є.О., Зубко В.М. Мобільна агрометеостанція для обприскувачів. Вісник Сумського національного аграрного університету:

науковий журнал. Суми : СНАУ, 2020. Вип. 2 (40). С. 3-7.

11. ESP8266 WIFI NODEMCU V2 CP2102 ESP-12E ARDUINO. 2023. URL: https://vroda.co.ua/product/esp8266-wifi-nodemcu-v2-cp2102-esp-12e-arduino_10839656422 (дата звернення: 24.10.2023).

12. WiFi Плата NodeMCU V2 ESP8266 (CP210). 2023. URL: <https://arduino.ua/ru/prod1495-wifi-plata-nodemcu-v2-esp8266-cp2102> (дата звернення: 24.10.2023).

13. Барометр BMP180 - датчик атмосферного тиску з функцією вимірювання температури. 2023. URL: <https://radiovolt.in.ua/ua/p1168909664-barometr-bmp180-datchik.html> (дата звернення: 28.10.2023).

14. Data sheet BMP180 Digital pressure sensor. 2023. URL: <https://www.alldatasheet.com/datasheet-pdf/pdf/514264/TI1/BMP180.html> (дата звернення: 28.10.2023).

15. DHT11–Temperature and Humidity Sensor. 2023. URL: <https://components101.com/sensors/dht11-temperature-sensor> (дата звернення: 28.10.2023).

16. Digital-output relative humidity & temperature sensor/module

17. DHT22 (DHT22 also named as AM2302). 2023. URL: <https://www.sparkfun.com/datasheets/Sensors/Temperature/DHT22.pdf> (дата звернення: 10.11.2023).

18. ZTS-3000-FSJT-*-*-1 聚碳酸酯风速传感器 产品使用手册. 2023. URL: <http://save.ztsiot.com/%E4%BA%A7%E5%93%81%E8%AF%B4%E6%98%8E%E4%B9%A6/ZTS/%E6%B0%94%E8%B1%A1%E5%86%9C%E4%B8%9A/%E8%81%9A%E7%A2%B3%E9%A3%8E%E9%80%9F%E4%BC%A0%E6%84%9F%E5%99%A8> (дата звернення: 10.11.2023).

19. Wind Speed Sensor - Weather Station - Anemometer - PR-3000-FSJT-V05. 2023. URL: <https://gleantronics.ie/en/products/wind-speed-sensor-weather-station-anemometer-pr-3000-fsjt-v05-4515.html> (дата звернення: 10.11.2023).

20. E108-GN02D. 2023. URL: <https://www.ebyte.com/en/product-view-news.html?id=1125> (дата звернення: 10.11.2023).

21. Node-RED. 2023. URL: <https://uk.wikipedia.org/wiki/Node-RED> (дата звернення: 10.11.2023).
22. Running on Windows. 2023. URL: <https://nodered.org/docs/getting-started/windows>(дата звернення: 10.11.2023).
23. Getting Started. 2023. URL: <https://nodered.org/docs/getting-started/> (дата звернення: 10.11.2023).
24. Dashboard with current conditions and forecast. 2023. URL: <https://flows.nodered.org/flow/b5b7d5da14d24e71de447e6aa290937e/in/rekJJdGYekDB> (дата звернення: 20.11.2023).
25. IoT Analytics – ThingSpeak Internet of Things. URL: <https://thingspeak.com/> (дата звернення: 02.11.2023).
26. Оператор switch. 2021. URL: <https://acode.com.ua/urok-68-operator-switch/> (дата звернення: 02.11.2023).
27. Оконський М.В., Лупенко С.А., Паламар А.М. Комп'ютерна система для моніторингу метеорологічних параметрів на основі IoT. Актуальні задачі сучасних технологій : збірник тез доповідей X міжнародної науково-технічної конференції молодих учених та студентів. Тернопіль: ТНТУ. 2021. С. 112.
28. Охорона праці в офісі. Вимоги до робочого місця офісного працівника. URL: <https://gc.ua/uk/oxorona-praci-v-ofisi-vimogi-do-robochogomiscya-ofisnogopracivnika/> (дата звернення: 26.11.2023).
29. Желібо Є. П., Сагайдак І. С. Безпека життєдіяльності. Навчальний посібник для аудиторної та практичної роботи. К.:ЕКОМЕН. 2011. 200 с. 26.
30. Депутат О. П., Коваленко І. В., Мужик І. С. Цивільна оборона. Навчальний посібник. Львів: Афіша. 2000. 336 с.
31. Нормативно-правові акти у сфері захисту рослин 2023. URL: <https://dpss.gov.ua/fitosanitariya-kontrol-u-sferi-nasinnictva-ta-rozsadnictva/fitosanitarnij-kontrol/normativno-pravovi-akti-u-sferi-zahistu-roslin> (дата звернення: 26.11.2023).
32. Микитишин А.Г., Митник М.М., Стухляк П.Д., Пасічник В.В. Комп'ютерні мережі. Книга 1. Львів, «Магнолія 2006». 2013. 256 с.

33. Микитишин А.Г., Митник М.М., Стухляк П.Д. Комплексна безпека інформаційних мережевих систем: навчальний посібник. Тернопіль: Вид-во ТНТУ імені Івана Пулюя, 2016. 256 с.

34. ДБН в.2.5-27-2006 захисні заходи електробезпеки в електроустановках будинків і споруд. Видання офіційне. Мінбуд України. Київ, 2006 156 с. (дата звернення 10.12.2023).

35. Муштин Д.І. Мобільна метеостанція для обприскувача. Матеріали XII Міжнародної науково-практичної конференції молодих учених та студентів Тернопільського національного технічного університету імені Івана Пулюя «Актуальні задачі сучасних технологій» (6-7 грудня 2023 року). Тернопіль: ТНТУ. 2023. С. 431.

36. Осухівська Г.М., Муштин Д.І. Комп'ютеризована система контролю за метеоданими для обприскувача. Матеріали XI науково-технічної конференції Тернопільського національного технічного університету імені Івана Пулюя «Інформаційні моделі, системи та технології» (13-14 грудня 2023 року). Тернопіль: ТНТУ. 2023. С. 166.

37. Лупенко С.А., Луцик Н.С., Луцків А.М., Осухівська Г.М., Тиш Є.В. Методичні вказівки до виконання кваліфікаційної роботи магістра для студентів спеціальності 123 «Комп'ютерна інженерія» другого (магістерського) рівня вищої освіти усіх форм навчання. Тернопіль, ТНТУ. 2021. 34 с.

38. Ларіоник Р.В., Луцик Н.С., Паламар А.М. Система для моніторингу якості атмосферного повітря на базі IoT. Матеріали IX науково-технічної конференції "Інформаційні моделі, системи та технології" Тернопільського національного технічного університету імені Івана Пулюя (Тернопіль, 8–9 грудня 2021 року), Тернопіль: ТНТУ. 2021. С. 116.

39. Оконський М.В., Лупенко С.А., Паламар А.М. Інформаційно-вимірювальна система для контролю метеорологічних параметрів на основі Інтернету речей. Матеріали IX науково-технічної конференції "Інформаційні моделі, системи та технології" Тернопільського національного технічного університету імені Івана Пулюя (Тернопіль, 8–9 грудня 2021 року), Тернопіль: ТНТУ, 2021. С. 118.

Додаток А

Опубліковані тези конференцій за темою кваліфікаційної роботи магістра

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Тернопільський національний технічний університет імені Івана Пулюя (Україна)
Університет імені П'єра і Марії Кюрі (Франція)
Маріборський університет (Словенія)
Технічний університет у Кошице (Словаччина)
Вільнюський технічний університет ім. Гедимінаса (Литва)
Міжнародний університет цивільної авіації (Марокко)
Наукове товариство ім. Т.Шевченка

АКТУАЛЬНІ ЗАДАЧІ СУЧАСНИХ ТЕХНОЛОГІЙ

Збірник
тез доповідей

**ХІІ Міжнародної науково-практичної
конференції молодих учених та студентів**
6-7 грудня 2023 року



УКРАЇНА
ТЕРНОПІЛЬ – 2023

38.	Т. Крамар ДЕЦЕНТРАЛІЗОВАНЕ АВТОМАТИЧНЕ ПІДКЛЮЧЕННЯ ПУНКТИВ НЕЗЛАМНОСТІ ПІД ЧАС ВІДКЛЮЧЕНЬ У ЗИМІ 2023 В ПРИФРОНТОВИХ ЗОНАХ УКРАЇНИ	415
39.	Б. Б. Млинко, О. П. Стефанюк АНАЛІЗ ВИКОРИСТАННЯ ІГРОВИХ РУШІВ ДЛЯ СТВОРЕННЯ ЦИФРОВИХ ДВІЙНИКІВ НА ОСНОВІ СИСТЕМНОГО ПІДХОДУ	417
40.	Н. М. Коцюк, В. Д. Тимошук, Ю. О. Момоток, Н. С. Луцк СИСТЕМА РЕЗЕРВУВАННЯ ТРАФІКУ НА ОСНОВІ МІКРОТК	419
41.	В. В. Василюшин, В. Д. Тимошук, Н. Ю. Кітчак, Н. С. Луцк АНАЛІЗ ХАРАКТЕРИСТИК ТА ЗАСТОСУВАННЯ МІКРОКОНТРОЛЕРІВ АТТІNY85, АТМЕGA8, RP2040	420
42.	А. М. Ковтко, Н. В. Лещук, І. Р. Козбур, І. В. Коноваленко АНАЛІЗ ЕФЕКТИВНОСТІ СИСТЕМ АВТОМАТИЗОВАНОГО ТЕСТУВАННЯ ПРОГРАМНИХ ПРОДУКТІВ	421
43.	О. Ю. Замора, А. В. Немеришин, І. Р. Козбур, О. Р. Дмитрів АНАЛІЗ МЕРЕЖЕВИХ СИСТЕМ АВТОМАТИЗОВАНОГО УПРАВЛІННЯ З ВИКОРИСТАННЯМ ПРОТОКОЛІВ МНОЖИННОГО ДОСТУПУ	423
44.	М. В. Дрогобицький, Н. С. Луцк, А. М. Паламар КОМП'ЮТЕРНА СИСТЕМА ДЛЯ ДИСТАНЦІЙНОГО КОНТРОЛЮ РІВНЯ ШУМУ НАВКОЛИШНЬОГО СЕРЕДОВИЩА	425
45.	І. В. Лилик, А. М. Паламар КОМП'ЮТЕРНА СИСТЕМА ДИСТАНЦІЙНОГО КОНТРОЛЮ ІНТЕНСИВНОСТІ УЛЬТРАФІОЛЕТОВОГО ВИПРОМІНЮВАННЯ	426
46.	А. М. Паламар, Д. С. Сомін, В. П. Волоський КОМП'ЮТЕРНА СИСТЕМА ДЛЯ ВІДДАЛЕНОГО СПОСТЕРЕЖЕННЯ ЗА РІВНЕМ НАСИЧЕННЯ КИСНЕМ КРОВІ ЛЮДИНИ	427
47.	М. В. Криховецький МЕТОДИ ВИЯВЛЕННЯ ДРОНІВ НА БАЗІ НЕЙРОННИХ МЕРЕЖ	428
48.	Д. І. Муштин МОБІЛЬНА МЕТЕОСТАНЦІЯ ДЛЯ ОБПРИСКУВАЧА	431
49.	Л. Є. Мосій, І. В. Струтинська, Г. В. Козбур РОЛЬ КОМП'ЮТЕРНО-ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ У ЦИФРОВІЙ ТРАНСФОРМАЦІЇ ЕКОНОМІКИ.	432
50.	О. Є. Подвисоцький; Н. Б. Стадник МЕТОДИ БІОМЕТРИЧНОЇ ІДЕНТИФІКАЦІЇ В РОЗУМНОМУ БУДИНКУ	435
51.	А. М. Паламар, Р. О. Романчук КОМП'ЮТЕРНА СИСТЕМА ДЛЯ ВІДДАЛЕНОГО КОНТРОЛЮ РІВНЯ ЗАБРУДНЕННЯ ПОВІТРЯ ПИЛОМ	436
52.	Є. В. Тиш, Р. І. Шалапай ТИПИ ВИМОГ ДО КОМП'ЮТЕРНИХ СИСТЕМ І МЕТОДИ ЇХ ВИЯВЛЕННЯ	437
53.	А. М. Луцків, С. В. Макогон НЕЙРОМЕРЕЖЕВІ ПІДХОДИ ДО ПЕРЕТВОРЕННЯ ТЕКСТОВИХ ПОВІДОМЛЕНЬ В АУДИОПОТІК	438
54.	В. В. Яцишин канд. І. М. Кучма ПОБУДОВА ОНТОЛОГІЙ ЯК СПОСІБ ЕФЕКТИВНОГО	439

УДК 004.75:631.1

Д. І. Муштин, Ph.D

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

МОБІЛЬНА МЕТЕОСТАНЦІЯ ДЛЯ ОБПРИСКУВАЧА

D. I. Mushtyn, Ph.D

MOBILE WEATHER STATION FOR A SPRAYER

Одним із шляхів підвищення якості спрацювання пестицидів крім контролю за погодними умовами є налаштування обприскувача відповідно до умов на кожному полі обробки. Вдалий підбір розміру краплі, швидкості руху та норми внесення пестицидів під фактичні умови поля дозволяє як знизити виробничі затрати, так і покращити ефективність від обприскування [1, 2], а будь-яке відхилення від умов - призводить до зниження ефективності обробки [3].

Отримати інформацію про фактичні умови на кожному полі для здійснення підбору найоптимальніших параметрів для обприскувача дозволяють дані з метеостанцій. Але, враховуючи динамічну зміну умов, найдоцільніше встановлювати метеостанції на обприскувачі.

На сьогоднішній день на українському ринку запропоновано ряд мобільних метеостанцій [4-6], але вони є або досить дорогими, або не дозволяють отримати необхідні дані. Саме тому ринок потребує доступну за вартістю метеостанцію для обприскувача із простим функціоналом та можливістю надання рекомендацій для оператора, які параметри потрібно змінити для підвищення якості внесення пестицидів. При цьому необхідно розраховувати такі параметри як Delta T, а також інверсію повітря.

Реалізувати керуючий модуль мобільної метеостанції доцільно здійснити на базі мікроконтролера ESP8266, який збирає, опрацьовує та передає дані від 2-х датчиків температури та вологості DHT-22 або DHT-11, датчика вітру ZTS-3000-FSJT, датчика атмосферного тиску BMP180, а також GPS приймача E108-GN02D. Для обміну даними використано MQTT, а для опрацювання даних, їх візуалізацію та отримання необхідної інформації про погодні умови на кожному конкретному полі - Node-RED.

Такі рішення дозволять реалізувати недорогу, портативну та функціональну мобільну метеостанцію, яку можна встановити на будь-якому самохідному і причіпному обприскувачі.

Література

1. Сторчоус І. Швидкість чи якість обприскування? – Більше не доведеться вибирати. 2023. URL: <https://www.growhow.in.ua/shvydkist-chy-iyakist-obpryskuvannia-bilshe-ne-dovedetsia-vybyrati/> (дата звернення: 24.09.2023).
2. Гончаров О. Більша крапля – менше знесення. 2020. URL: <https://agroexpert.ua/bilsha-kraplia-menshe-znesennia/> (дата звернення: 24.09.2023).
3. Гузь М., Сінько В., Марченко В. Обприскування: на що слід звернути особливу увагу. 2017. URL: <https://agroexpert.ua/obpriskuvanna-na-so-slid-zvernuti-osoblivu-uvagu> (дата звернення: 24.09.2023).
4. Wolf.T. Spraying Weather. 2022. URL: <https://sprayers101.com/spraying-weather> (дата звернення: 24.09.2023).
5. John Deere Mobile Weather. 2023. URL: <https://www.deere.com/en/technology-products/precision-ag-technology/field-and-water-management/mobile-weather/> (дата звернення: 24.09.2023).
6. МЕТЕОТРЕК MOBI 2023. URL: <https://www.meteotrek.ua/equipment/tproduct/353440524-366263547451-meteotrek-mobi> (дата звернення: 24.09.2023).

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТЕРНОПЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ
УНІВЕРСИТЕТ ІМЕНІ ІВАНА ПУЛЮЯ**

МАТЕРІАЛИ

XI НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ

**«ІНФОРМАЦІЙНІ МОДЕЛІ,
СИСТЕМИ ТА ТЕХНОЛОГІЇ»**



13-14 грудня 2023 року

**ТЕРНОПЛЬ
2023**

- Ясній О.П., Крисюк І.В.**
ФАКТОРИ ВПЛИВУ НА НАДІЙНІСТЬ КОМП'ЮТЕРНИХ СИСТЕМ В ПРОЦЕСІ ЇХ РОЗРОБКИ
Yasniy O.P., Krysiuk I.V.
EFFECTS RELIABILITY FACTORS OF COMPUTER SYSTEMS IN THE PROCESS OF THEIR DEVELOPMENT 161
- Василь Яцишин, Іван Кучма**
КЛАСИФІКАЦІЯ ОНТОЛОГІЙ В ПРОЦЕСІ МОДЕЛЮВАННЯ КОМП'ЮТЕРНИХ МЕРЕЖ
Vasyl Yatsyshyn, Ivan Kuchma
CLASSIFICATION OF ONTOLOGIES IN THE PROCESS OF COMPUTER NETWORK MODELING 162
- І.В. Лилик, А.М. Паламар**
КОМП'ЮТЕРИЗОВАНА СИСТЕМА МОНІТОРИНГУ РІВНЯ УЛЬТРАФІОЛЕТОВОГО ВИПРОМІНЮВАННЯ НА ОСНОВІ ІНТЕРНЕТУ РЕЧЕЙ
I.V. Lylyk, A.M. Palamar
COMPUTERIZED ULTRAVIOLET RADIATION LEVEL MONITORING SYSTEM BASED ON THE INTERNET OF THINGS 163
- Андрій Луцків, Сергій Макогон**
ТИПИ АРХІТЕКТУР НЕЙРОННИХ МЕРЕЖ ДЛЯ ПЕРЕТВОРЕННЯ ТЕКСТОВИХ ПОВІДОМЛЕНЬ У ЗВУКОВИЙ ПОТІК
Andriy Lutskiv, Serhii Makohon
TYPES OF NEURAL NETWORK ARCHITECTURES FOR TEXT TO SPEECH 164
- Андрій Луцків, Юрій Мельничук**
МУЛЬТИАГЕНТНА ОРГАНІЗАЦІЯ СЕРВЕРА ОНЛАЙН АУКЦІОНІВ
Andriy Lutskiv, Yuriy Melnychuk
MULTI-AGENCY ONLINE AUCTION SERVER ORGANIZATION 165
- Галина Осухівська, Денис Муштин**
КОМП'ЮТЕРИЗОВАНА СИСТЕМА КОНТРОЛЮ ЗА МЕТЕОДАНИМИ ДЛЯ ОБПРИСКУВАЧА
Halyna Osukhivska, Denys Mushtyn
COMPUTERIZED METEODATA CONTROL SYSTEM FOR SPRAYER 166
- Т.А. Озарків; Р.О. Жаровський**
МЕТОД ОПТИМІЗАЦІЇ EIGRP ПРОТОКОЛУ ДЛЯ ПІДВИЩЕННЯ ПРОДУКТИВНОСТІ ПЕРЕДАЧІ ДАНИХ В КОМП'ЮТЕРНИХ МЕРЕЖАХ
T. A. Ozarkiv; R.O. Zharovskyi
THE METHOD OF OPTIMIZING THE EIGRP PROTOCOL TO INCREASE THE PRODUCTIVITY OF DATA TRANSMISSION IN COMPUTER NETWORKS 167
- Андрій Луцків, Андрій Островський**
ОРГАНІЗАЦІЯ ДОСТУПУ ДО МОДЕЛІ GPT-3 ЗАСОБАМИ МОВИ PYTHON
Andriy Lutskiv, Andriy Ostrovskiy
ORGANIZING ACCESS TO THE GPT-3 MODEL USING PYTHON 168
- А.М. Паламар, Р.О. Романчук, М.В. Дрогобицький**
КОМП'ЮТЕРИЗОВАНА СИСТЕМА ДЛЯ ДИСТАНЦІЙНОГО КОНТРОЛЮ РІВНЯ КОНЦЕНТРАЦІЇ ПИЛУ НА ОСНОВІ ІНТЕРНЕТУ РЕЧЕЙ
A.M. Palamar, R.O. Romanchuk, M.V. Drohobytskyi
COMPUTERIZED SYSTEM FOR REMOTE MONITORING OF DUST CONCENTRATION LEVEL BASED ON THE INTERNET OF THINGS 169
- Ярослав Панчишин**
СТРУКТУРА СИСТЕМИ КОНТРОЛЮ ПАРАМЕТРІВ МІКРОКЛІМАТУ МІНІ-ТЕПЛИЦІ
Yaroslav Panchyshyn
STRUCTURE OF THE MINI-GREENHOUSE MICROCLIMATE PARAMETER CONTROL SYSTEM 170

УДК 681.518.3

Галина Осухівська, канд. техн. наук, доц.; Денис Муштин, Ph.D

Тернопільський національний технічний університет імені Івана Пулюя, Україна

КОМП'ЮТЕРИЗОВАНА СИСТЕМА КОНТРОЛЮ ЗА МЕТЕОДАНИМИ ДЛЯ ОБПРИСКУВАЧА

Halyna Osukhivska, Ph.D, Assoc. Prof; Denys Mushtyn, Ph.D

COMPUTERIZED METEODATA CONTROL SYSTEM FOR SPRAYER

Розробка комп'ютеризованої системи контролю за метеоданими для обприскувача, яка би враховувала фактичні умови поля, прогноз погоди від найближчих метеостанцій, а також надавала рекомендації оператору обприскувача про підбір розміру краплі, швидкості руху та норми внесення пестицидів є важливим та актуальним завданням.

В запропонованій системі керуючий модуль реалізовано на базі мікроконтролера ESP8266, для обміну даними використано MQTT, для опрацювання отриманих даних та їх візуалізації - Node-RED. Також в систему додано Telegram bot, який здійснює інформування про фактичні дані, які є на полі, а також надсилає команди оператору обприскувача про доцільність збільшення чи зменшення швидкості руху, або припинення обприскування.

На Рисунку 1 наведено загальну архітектуру комп'ютеризованої системи контролю за метеоданими для обприскувача, на Рисунку 2 - приклад візуалізації даних на моніторі комп'ютера в платформі Node-red.

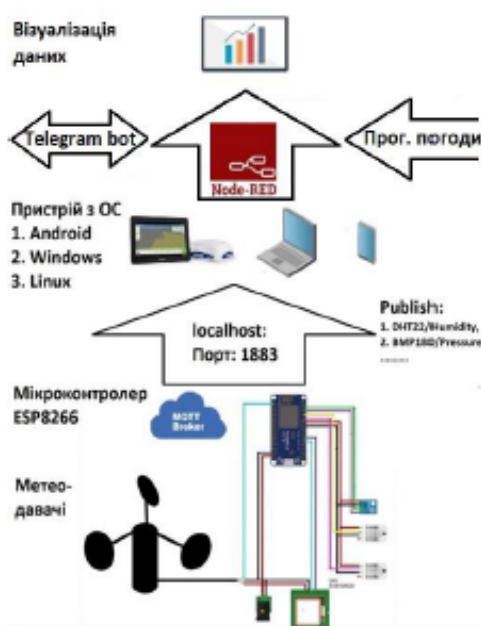


Рисунок 1. Загальна архітектура комп'ютеризованої системи контролю за метеоданими для обприскувача



Рисунок 2. Зразок візуалізації даних на моніторі комп'ютера

Таким чином, запропонована комп'ютеризована система забезпечує контроль за фактичними метеоданими на полі, враховує прогноз погоди, а також, на основі зібраних даних, надає рекомендації оператору щодо підвищення ефективності роботи обприскувача.

Додаток Б

Лістинг коду

```
#include <ESP8266WiFi.h>
#include "uMQTTBroker.h"
#include <Adafruit_BMP085.h>
#include <DHT.h>
#include <TinyGPS++.h>
#include <SoftwareSerial.h>
#include <PubSubClient.h>

uMQTTBroker myBroker;

char ssid[] = "*****";
char pass[] = "*****";

#define DHT_PIN D3
#define DHT11_PIN D4
#define BMP_SDA D2
#define BMP_SCL D1

Adafruit_BMP085 bmp;
DHT dht(DHT_PIN, DHT22);
DHT dht11(DHT11_PIN, DHT11);

const int sensorPin = A0;
float voltageMin = 0.65;
float windSpeedMin = 0;
float voltageMax = 3.3;
float windSpeedMax = 61.2;

static const int RXPin = D8;
static const int TXPin = D7;
TinyGPSPlus gps;
SoftwareSerial ss(RXPin, TXPin);

// Таймери та стани
unsigned long previousMillisReadSensors = 0;
const long intervalReadSensors = 4000;
unsigned long previousMillisPublishData = 0;
const long intervalPublishData = 8000;

enum State {
    STATE_READ_SENSORS,
    STATE_PUBLISH_DATA
};

State currentState = STATE_READ_SENSORS;

// Declare global variables here
```

```

float humidity;
float temperature;
float pressure;
float temperatureBMP;
float humidityDHT11;
float temperatureDHT11;
float sensorValue;
float voltage;
float wind_speed;
float dewpoint;
float deltaT;

void startWiFiClient() {
  Serial.println("Connecting to " + (String)ssid);
  WiFi.begin(ssid, pass);

  while (WiFi.status() != WL_CONNECTED) {
    delay(2000);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: " + WiFi.localIP().toString());
}

void startWiFiAP() {
  WiFi.softAP(ssid, pass);
  Serial.println("AP started");
  Serial.println("IP address: " + WiFi.softAPIP().toString());
}

void readSensors() {
  // Assign values to global variables
  humidity = dht.readHumidity();
  temperature = dht.readTemperature();
  pressure = bmp.readPressure() / 100.0;
  temperatureBMP = bmp.readTemperature();
  humidityDHT11 = dht11.readHumidity();
  temperatureDHT11 = dht11.readTemperature();
  sensorValue = analogRead(sensorPin);
  voltage = (sensorValue / 1024.0) * 3.3;
  wind_speed = mapfloat(voltage, voltageMin, voltageMax, windSpeedMin, windSpeedMax);
  dewpoint = temperature - (100 - humidity) / 5;
  deltaT = temperature - dewpoint;
}

void publishData() {
  // Ваш код для публікації даних
  Serial.print("Analog Value =");
  Serial.println(sensorValue);
  Serial.print("Voltage =");

```

```

Serial.print(voltage);
Serial.println(" V");
Serial.print("Wind Speed =");
Serial.print(wind_speed);
Serial.println("m/s");
myBroker.publish("DHT22/Humidity", String(humidity));
myBroker.publish("DHT22/Temperature", String(temperature));
myBroker.publish("BMP180/Pressure", String(pressure));
myBroker.publish("BMP180/Temperature", String(temperatureBMP));
myBroker.publish("DHT11/Humidity", String(humidityDHT11));
myBroker.publish("DHT11/Temperature", String(temperatureDHT11));
myBroker.publish("WindSen/wind_speed", String(wind_speed));
myBroker.publish("DHT11/dewpoint", String(dewpoint));
myBroker.publish("DHT11/deltaT", String(deltaT));
}

void publishGPSData() {
  myBroker.publish("GPS/Latitude", String(gps.location.lat(), 6).c_str());
  myBroker.publish("GPS/Longitude", String(gps.location.lng(), 6).c_str());
  myBroker.publish("GPS/Altitude", String(gps.altitude.meters()).c_str());
  myBroker.publish("GPS/Speed", String(gps.speed.kmph()).c_str());
  myBroker.publish("GPS/Date", String(gps.date.year()) + "." + gps.date.month() + "." +
gps.date.day());
  myBroker.publish("GPS/Time", String(gps.time.hour()) + ":" + gps.time.minute());
  myBroker.publish("GPS/Satellites", String(gps.satellites.value()).c_str());
  delay(1000);
}

void setup() {
  Serial.begin(115200);
  startWiFiClient();
  myBroker.init();
  dht.begin();
  bmp.begin();
  dht11.begin();
  ss.begin(9600);
  Serial.println("Starting MQTT broker");
  Serial.println("Датчики ініціалізовано");
}

void loop() {
  // GPS
  while (ss.available() > 0) {
    if (gps.encode(ss.read())) {
      publishGPSData();
    }
  }
}

unsigned long currentMillisReadSensors;
unsigned long currentMillisPublishData;

```



```
switch (currentState) {
  case STATE_READ_SENSORS:
    currentMillisReadSensors = millis();
    if (currentMillisReadSensors - previousMillisReadSensors >= intervalReadSensors) {
      readSensors();
      previousMillisReadSensors = currentMillisReadSensors;
      currentState = STATE_PUBLISH_DATA;
    }
    break;

  case STATE_PUBLISH_DATA:
    currentMillisPublishData = millis();
    if (currentMillisPublishData - previousMillisPublishData >= intervalPublishData) {
      publishData();
      previousMillisPublishData = currentMillisPublishData;
      currentState = STATE_READ_SENSORS;
    }
    break;
}
}

float mapfloat(float x, float in_min, float in_max, float out_min, float out_max) {
  return (x - in_min) * (out_max - out_min) / (in_max - in_min) + out_min;
}
```