

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ВОДНОГО
ГОСПОДАРСТВА ТА ПРИРОДОКОРИСТУВАННЯ**

Навчально-науковий інститут кібернетики,
інформаційних технологій та інженерії
Кафедра комп'ютерних наук та прикладної математики

“До захисту допущена”
Завідувач кафедри комп'ютерних
наук та прикладної математики
Турбал Юрій Васильович
_____ 20__ р.

КВАЛІФІКАЦІЙНА РОБОТА

**«ПРОЄКТУВАННЯ ТА РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ
ІНТЕЛЕКТУАЛЬНОГО НАЗЕМНОГО ДРОНУ-ДЕТЕКТОРА АКУСТИЧНИХ АНОМАЛІЙ З
ЕЛЕМЕНТАМИ АВТОНОМНОГО КЕРУВАННЯ ТА ПЕРЕДАЧЕЮ ДАНИХ»**

Виконав: Федоров Ілля Олександрович _____
(прізвище, ім'я, по батькові) (підпис)

група ПЗ-41

Керівник: Зубик Я. Я. _____
(науковий ступінь, вчене звання, посада, прізвище, ініціали) (підпис)

Рівне 2025

Національний університет водного господарства та природокористування

Навчально науковий інститут автоматичної, кібернетики та обчислювальної техніки

Кафедра комп'ютерних наук та прикладної математики

Освітньо-кваліфікаційний рівень **бакалавр**

Галузь знань 12 «Інформаційні технології»

Спеціальність 121 «Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ

Зав. Кафедри д.т.н., проф.

_____ Турбал Ю. В.

«__» _____ 2025 року

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

Федорову Іллі Олександровичу

1. Тема роботи *Проектування та розробка програмного забезпечення для інтелектуального наземного дрону-детектора акустичних аномалій з елементами автономного керування та передачею даних*

керівник Зубик Ярослав Ярославович, ст. викл.

затверджені наказом вищого навчального закладу від 18 квітня 2025 року С № 463.

2. Термін подання роботи студентом 30 травня 2025 р.

3. Вихідні дані: технічні засоби для виявлення акустичних аномалій і передачі даних у мобільних системах. Архітектури програмних систем для керування наземним дроном та веб-інтерфейсів, засоби тестування функціональності.

4. Зміст розрахунково-пояснювальної записки: Описано етапи проектування, вибір компонентів і реалізацію програмного забезпечення для наземного дрону з елементами автономного керування. Надано технічні характеристики, схеми підключення, логіку веб-керування та механізм реагування на акустичні події. Проведено тестування функціональності в умовах, наближених до реального застосування.

5.Консультанти розділів роботи(проекту)

Розділ	Прізвище, ініціали, посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв
Розділ 1	Зубик Я.Я.	05.03.25	07.03.25
Розділ 2	Зубик Я.Я.	02.04.25	04.04.25
Розділ 3	Зубик Я.Я.	01.05.25	03.05.25

7. Дата видачі завдання : 05.03.2025 р.

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Огляд літературних джерел з тематики автономних дронів та методів виявлення акустичних аномалій	05.03.25 – 12.03.25	Виконав
2	Вибір технічних компонентів і модулів, формування загальної архітектури системи	13.03.25 – 19.03.25	Виконав
3	Розробка схеми підключення компонентів та збірка апаратної частини	20.03.25 – 26.03.25	Виконав
4	Програмування ESP32, реалізація логіки керування та веб-інтерфейсу	27.03.25 – 07.04.25	Виконав
5	Інтеграція сенсорів, обробка подій та автономного реагування	08.04.25 – 17.04.25	Виконав
6	Проведення тестування в умовах, наближених до реального застосування	18.04.25 – 25.04.25	Виконав
7	Оформлення пояснювальної записки, підготовка виступу та графічних матеріалів до захисту	26.04.25 – 05.06.25	Виконав

Студент

Федоров І.О.

(підпис)

(прізвище та ініціали)

Керівник роботи

Зубик Я. Я.

(підпис)

(прізвище та ініціали)

ЗМІСТ

РЕФЕРАТ	6
ВСТУП	7
РОЗДІЛ 1	
ТЕОРЕТИЧНІ ОСНОВИ РОЗРОБКИ ІНТЕЛЕКТУАЛЬНИХ НАЗЕМНИХ СИСТЕМ ІЗ СЕНСОРНОЮ ВЗАЄМОДІЄЮ	9
1.1. Сучасні підходи до створення мобільних роботизованих систем	9
1.2. Особливості побудови наземного дрону з сенсорним аналізом.....	10
1.3. Мікроконтролер ESP32 як основа мобільної платформи	11
1.4. Реалізація дистанційного керування через веб-інтерфейс.....	13
1.5. Теоретичні основи роботи GPS-модуля та акустичного сенсора	14
1.5.1. Основи функціонування GPS-модуля.....	14
1.5.2. Мікрофонний модуль і принципи виявлення акустичних подій ...	16
РОЗДІЛ 2	
ПРОЄКТУВАННЯ АПАРАТНО-ПРОГРАМНОЇ ЧАСТИНИ НАЗЕМНОГО ДРОНУ	18
2.1. Загальна архітектура системи	18
2.2. Вибір апаратних компонентів	19
2.3. Схема з'єднань та підключення компонентів	24
2.4. Програмна архітектура та структура коду.....	27
2.5. Реалізація логіки керування рухом.....	29
2.6. Реалізація Wi-Fi з'єднання та веб-інтерфейсу	31
2.7. Інтеграція сенсорів GPS та мікрофону.....	33
2.8. Інструкція зі збирання та налагодження системи.....	35

2.9. Типові проблеми реалізації та шляхи їх усунення	36
РОЗДІЛ 3	
ТЕСТУВАННЯ І ОЦІНКА РЕЗУЛЬТАТІВ РОБОТИ СИСТЕМИ	40
3.1. Умови проведення випробувань.....	40
3.2. Перевірка функціональності керування.....	41
3.3. Робота модуля GPS	43
3.4. Реакція на акустичні аномалії.....	44
3.5. Сценарії використання та поведінка системи	46
3.6. Інструкція користувача.....	48
3.7. Оцінка ефективності системи	49
ВИСНОВКИ	52
ВИКОРИСТАНІ ДЖЕРЕЛА	54

РЕФЕРАТ

Кваліфікаційна робота: 55 с., 9 рисунків, 10 джерел.

Мета роботи: Розробка програмного забезпечення для інтелектуального наземного дрону-детектора акустичних аномалій, що забезпечує елементи автономного керування, передачу координат у відповідь на зовнішні звукові події та дистанційну взаємодію з користувачем через вбудований веб-інтерфейс.

Об'єкт дослідження – мобільні роботизовані платформи з елементами сенсорного аналізу та дистанційного керування.

Предмет дослідження – технічні та програмні рішення, що використовуються для реалізації взаємодії між користувачем і роботизованою системою, а також методи зчитування й обробки акустичних та просторових даних.

Методи вивчення – аналіз електронних компонентів, побудова апаратних схем, програмування в середовищі Arduino IDE, дослідження протоколів локальної бездротової передачі даних, експериментальне тестування.

У роботі проведено аналіз теоретичних і практичних основ побудови наземних мобільних систем. Вивчено можливості використання мікроконтролера ESP32 для реалізації Wi-Fi зв'язку та локального веб-інтерфейсу. Реалізовано систему керування з інтеграцією GPS-модуля та акустичного сенсора для виявлення змін у навколишньому середовищі. Проведено тестування в різних умовах з аналізом ефективності взаємодії між модулями та оцінкою точності сенсорних даних.

Ключові слова: НАЗЕМНИЙ ДРОН, ESP32, Wi-Fi, ВЕБ-ІНТЕРФЕЙС, GPS, АКУСТИЧНИЙ СЕНСОР, МОБІЛЬНА ПЛАТФОРМА, ДИСТАНЦІЙНЕ КЕРУВАННЯ.

ВСТУП

У сучасних умовах розвитку інформаційних технологій зростає попит на мобільні, компактні пристрої з можливістю взаємодії з навколишнім середовищем у режимі реального часу. Особливий інтерес викликають наземні дрони як платформи для реалізації таких систем — зокрема, в контексті виявлення подій, що супроводжуються акустичними аномаліями, та передачі інформації на мобільні пристрої користувача.

Актуальність теми полягає у потребі створення універсальних систем мобільного моніторингу, здатних працювати в режимі дистанційного керування з одночасним відстеженням акустичних змін. Такі системи можуть бути застосовані у сферах безпеки, пошуково-рятувальних операцій, охорони периметру, екологічного моніторингу тощо. У багатьох випадках важливим є не лише пересування пристроєм, а й здатність реагувати на зміни в середовищі без участі оператора, що створює передумови для впровадження елементів автономного реагування.

Метою роботи є проєктування та розробка програмного забезпечення для наземного дрону-детектора, яке забезпечує інтелектуальне керування, обробку даних з GPS- та мікрофонного сенсорів, а також реагування на акустичні події з можливістю віддаленої взаємодії через веб-інтерфейс.

У межах виконання кваліфікаційної роботи передбачено послідовне дослідження сучасних підходів до побудови мобільних платформ із дистанційним керуванням і частковою автономією, зокрема зосереджено увагу на аналізі існуючих рішень та архітектур подібних систем. Проєкт передбачає розробку апаратної частини на основі мікроконтролера ESP32, інтеграцію акустичного сенсора для виявлення звукових аномалій, реалізацію механізмів передачі даних на клієнтський пристрій через Wi-Fi, а також програмування логіки реагування на події, що включає зупинку руху, обробку координат і сповіщення користувача. Завершальним етапом виступає тестування розробленої

системи в різноманітних умовах з подальшим аналізом її стабільності та ефективності.

Об'єкт дослідження — мобільна сенсорна система з елементами віддаленого керування.

Предмет дослідження — алгоритми обробки акустичних сигналів, логіка реагування на події та програмна реалізація передачі даних у реальному часі.

Наукова новизна полягає у поєднанні простого конструктора з системою сенсорного моніторингу, в якому реалізовано елементи автономного реагування — таких як автоматична зупинка або сповіщення при виявленні аномалії без участі користувача.

Практичне значення полягає в можливості використання пристрою як прототипу для розробки реальних систем безпеки, патрулювання, моніторингу шуму на будівельних майданчиках, військових полігонах, у приміщеннях та на відкритій місцевості.

Методи дослідження: використання мікроконтролера ESP32, мікрофону, GPS-модуля, алгоритмів порогового виявлення шуму, Wi-Fi комунікацій та тестування реакції системи на акустичні події. Робота складається зі вступу, трьох розділів, висновків, списку використаних джерел і додатків. У першому розділі розглянуто теоретичні основи та сучасні підходи. У другому — описано апаратну реалізацію. У третьому — представлено програмну частину та тестування.

РОЗДІЛ 1

ТЕОРЕТИЧНІ ОСНОВИ РОЗРОБКИ ІНТЕЛЕКТУАЛЬНИХ НАЗЕМНИХ СИСТЕМ ІЗ СЕНСОРНОЮ ВЗАЄМОДІЄЮ

1.1. Сучасні підходи до створення мобільних роботизованих систем

У сучасному світі роботизовані мобільні платформи посідають важливе місце в розвитку науково-технічного прогресу, оскільки дають змогу автоматизувати широкий спектр завдань, які раніше вимагали фізичної участі людини. Їх застосування охоплює різноманітні галузі: від промислових процесів, логістики й складських операцій — до екологічного моніторингу, пошуково-рятувальних місій, сільського господарства, охоронних функцій та навіть участі у військових і розвідувальних операціях. Саме універсальність та адаптивність таких систем забезпечує їх швидке впровадження у реальні завдання.

Залежно від середовища використання, мобільні платформи умовно поділяються на наземні, повітряні (аероботи чи дрони), підводні або комбіновані (амфібійні). Наземні системи, на відміну від повітряних, мають низку переваг, серед яких — відносна простота конструкції, нижча вартість виробництва, стабільність пересування на горизонтальних поверхнях, менше енергоспоживання та більша вантажопідйомність. Ці фактори роблять наземні дрони особливо зручними для побудови систем автономного патрулювання, виявлення аномалій, дослідження територій та іншого функціонального навантаження, яке не потребує вертикального переміщення.

З появою мікроконтролерів нового покоління (таких як ESP32, Arduino Nano, STM32) та мікрокомп'ютерів на кшталт Raspberry Pi, розробка мобільних платформ стала доступною не лише для промислових компаній, а й для інженерів-ентузіастів, студентів, дослідницьких лабораторій. Завдяки широкій підтримці бібліотек, наявності документації, відкритому програмному

забезпеченню та спільнотам розробників, створення автономних або напіваавтономних роботизованих систем значно спростилося.

Ключовою особливістю сучасних підходів є інтеграція сенсорних модулів, бездротових інтерфейсів (Wi-Fi, Bluetooth, LoRa) та простих алгоритмів прийняття рішень, які забезпечують не лише дистанційне керування пристроями, а й можливість автономної поведінки в залежності від зовнішніх факторів. Усе це дає змогу формувати нове покоління мобільних інтелектуальних систем, здатних до самостійного функціонування у змінному середовищі.

1.2. Особливості побудови наземного дрону з сенсорним аналізом

Наземні роботизовані системи, що виконують аналіз навколишнього середовища, проєктуються як комплекс взаємопов'язаних підсистем, кожна з яких виконує специфічні функції, необхідні для автономної або напіваавтономної роботи пристрою. Такий підхід дозволяє досягти високої модульності, спрощує процес налагодження і подальшої модернізації системи, а також забезпечує чітке логічне розділення функціональності між окремими елементами.

Ключову роль у подібних розробках відіграє механічна платформа, на якій базується вся система. Вона включає виконавчі механізми, що забезпечують пересування дрона в просторі — як правило, це електродвигуни, підключені через драйвер, та серводвигун, який відповідає за зміну напрямку руху. Конфігурація цих компонентів впливає на стабільність пересування, можливість маневрування та тип поверхні, по якій може переміщуватись пристрій.

Не менш важливим є програмно-апаратний блок обробки даних. Саме тут здійснюється прийом інформації з різних сенсорів: акустичних, температурних, візуальних, інфрачервоних тощо. У межах цього дослідження основну увагу приділено обробці звукових даних, що надходять з мікрофонного модуля. На основі зафіксованих параметрів шуму система здатна ідентифікувати аномальні

події, що можуть свідчити про присутність рухомого об'єкта, небезпечну ситуацію або інший зовнішній вплив.

Ще однією невід'ємною складовою сучасного дрону є підсистема зв'язку, що забезпечує обмін даними між пристроєм і користувачем або зовнішніми системами. Для передавання команд, а також потенційного зчитування координат і стану пристрою можуть використовуватися різні протоколи — зокрема Wi-Fi, Bluetooth, GSM або LoRa. У даному проєкті реалізовано використання бездротового модуля Wi-Fi, інтегрованого в мікроконтролер ESP32, який виступає в ролі точки доступу. Така архітектура дозволяє користувачу підключатися до пристрою без участі сторонніх мереж, що значно підвищує автономність та безпеку функціонування системи.

Завдяки поєднанню трьох зазначених функціональних блоків — переміщення, обробки даних та комунікації — формується гнучка і функціональна система, здатна виконувати базові інтелектуальні дії та реагувати на зміну параметрів зовнішнього середовища в режимі реального часу. Це відкриває широкі можливості для застосування таких дронів у сферах моніторингу, охорони територій, технічного нагляду та досліджень.

1.3. Мікроконтролер ESP32 як основа мобільної платформи

Мікроконтролер ESP32 є одним із найпотужніших і найпопулярніших рішень для реалізації вбудованих систем з підтримкою бездротового зв'язку. Завдяки своїй високій продуктивності, великій кількості доступних пінів введення/виведення, а також широкій підтримці периферійних інтерфейсів, він став платформою вибору для розробників інтелектуальних IoT-пристроїв, систем автоматизації та мобільних роботизованих платформ.

На відміну від попередника ESP8266, мікроконтролер ESP32 має двоядерний процесор Tensilica Xtensa LX6, що дозволяє виконувати паралельні обчислювальні задачі, у тому числі роботу з веб-інтерфейсом, управління рухом,

обробку сенсорних даних і фонові процеси. Це особливо важливо для мобільних систем, які мають працювати в режимі реального часу і обробляти декілька потоків даних одночасно без втрати стабільності.

До стандартного функціоналу ESP32 входить вбудований модуль Wi-Fi, який підтримує режими як точки доступу (AP), так і клієнта (STA), а також можливість одночасної роботи в обох режимах. Підтримка повного стеку TCP/IP забезпечує стабільний обмін даними з іншими пристроями або користувацьким інтерфейсом. Додатково ESP32 має вбудований Bluetooth (Classic і BLE), що відкриває можливості для розширення каналу зв'язку, хоча в цьому проєкті він не використовується.

Платформа підтримує запуск легкого веб-сервера безпосередньо на мікроконтролері. Це дозволяє реалізовувати інтуїтивно зрозумілий інтерфейс керування, до якого користувач підключається через Wi-Fi, не використовуючи зовнішніх роутерів або інтернет-з'єднання. Зі сторони програмування, ESP32 має широку сумісність із середовищем Arduino IDE та підтримує численні бібліотеки, що значно спрощує реалізацію навіть складних функцій, таких як обробка сигналів із сенсорів або робота з GPS-модулями через UART.

У межах даного проєкту ESP32 виконує низку важливих функцій, зокрема керування електродвигунами, які відповідають за переміщення платформи. Це реалізовано через цифрові пін-коди, що подають сигнали на драйвер двигунів. Крім того, ESP32 здійснює управління серводвигуном, що відповідає за зміну напрямку руху. Окремий блок реалізовано для запуску та обслуговування веб-інтерфейсу, який є основним каналом взаємодії між пристроєм і користувачем.

Додатково мікроконтролер використовується для зчитування даних із сенсорів. До нього підключено мікрофонний модуль, що дозволяє виявляти зміни звукового фону, а також GPS-модуль, який передає координати у стандартному форматі. Таким чином, ESP32 виконує функції обчислювального ядра, комунікаційного інтерфейсу та центра управління для всієї мобільної системи.

1.4. Реалізація дистанційного керування через веб-інтерфейс

Ефективна взаємодія користувача з мобільною платформою є важливою умовою для досягнення функціональної автономії системи. В даному проєкті реалізовано дистанційне керування через вбудований веб-інтерфейс, який розміщується безпосередньо на мікроконтролері ESP32. Такий підхід дозволяє забезпечити максимально автономну роботу пристрою без потреби у зовнішніх мережах або доступі до Інтернету, що є особливо важливим у польових умовах, при виконанні завдань у віддалених районах або в ситуаціях, коли стабільне інтернет-з'єднання відсутнє.

Після увімкнення живлення мікроконтролер автоматично ініціалізує створення Wi-Fi точки доступу. Користувач може підключитися до цієї мережі за допомогою смартфона, планшета чи ноутбука. Адреса веб-інтерфейсу, як правило, має вигляд локальної IP-адреси (192.168.1.101), яка відкривається в будь-якому стандартному браузері. У результаті відкривається веб-сторінка з інтерактивним інтерфейсом керування платформою.

Інтерфейс розроблено з використанням технологій HTML, CSS та JavaScript, що забезпечує як адаптивність зовнішнього вигляду під різні розміри екранів, так і реактивність у роботі. Користувач бачить панель керування з набором кнопок, які відповідають за напрямки руху: вперед, назад, вліво, вправо, а також зупинку. Реалізація кнопок передбачає не лише одноразову подачу команди, але й повторювану передачу сигналу з певною періодичністю (наприклад, кожні 500 мс) при утриманні кнопки. Це дає змогу досягти плавного, інтуїтивного управління, подібного до принципу роботи фізичних геймпадів.

Передача команд відбувається за допомогою HTTP GET-запитів, які генеруються скриптами JavaScript при взаємодії з елементами інтерфейсу. Ці запити надсилаються на вбудований веб-сервер ESP32, де обробляються функціями мікроконтролера. У залежності від вмісту запиту, мікроконтролер

виконує відповідну дію — подає сигнали на драйвер двигуна або змінює положення серводвигуна.

Завдяки такій реалізації керування забезпечується максимальна зручність та мобільність для користувача. Взаємодія з платформою можлива без інсталяції додаткового програмного забезпечення, виключно за допомогою браузера, що розширює коло потенційних пристроїв керування. Така архітектура ідеально підходить для освітніх проєктів, прототипів польових систем, а також для мобільних технічних рішень, які повинні зберігати простоту у використанні та незалежність від складної інфраструктури зв'язку.

1.5. Теоретичні основи роботи GPS-модуля та акустичного сенсора

У мобільних платформах, які здійснюють орієнтацію у просторі та реагують на зовнішні подразники, важливу роль відіграють сенсорні модулі, що забезпечують збір та обробку даних про навколишнє середовище. Зокрема, модулі глобального позиціонування (GPS) використовуються для визначення координат пристрою, що дозволяє реалізувати просторову логіку поведінки, фіксацію маршрутів і контроль меж зон дії. Водночас мікрофонні сенсори забезпечують виявлення змін у звуковому середовищі, дозволяючи системі реагувати на акустичні аномалії, які можуть бути пов'язані з рухом об'єктів, голосовими командами або потенційними загрозами. У цьому пункті розглянуто теоретичні основи функціонування зазначених сенсорів, їхню архітектуру та принципи роботи в контексті інтеграції до мобільної роботизованої системи.

1.5.1. Основи функціонування GPS-модуля

Система глобального позиціонування (GPS) є одним із найпоширеніших інструментів для визначення просторового розташування об'єкта. Принцип її роботи ґрунтується на обміні сигналами між приймачем і групою супутників, що

обертаються навколо Землі. Кожен супутник транслює унікальний радіосигнал із часовою міткою та власним положенням у просторі. Приймач, аналізуючи затримку сигналів від кількох супутників, обчислює власні координати у вигляді широти, довготи та (за потреби) висоти.

У даному проєкті використовується модуль NEO-6M — один із найпоширеніших GPS-приймачів, сумісних з Arduino та ESP32. Цей модуль забезпечує оновлення координат із частотою до одного разу на секунду та здатен досягати точності позиціонування в межах 2,5 метрів за умови прямої видимості неба. Він підключається до мікроконтролера через UART-інтерфейс, що дозволяє ефективно передавати великі обсяги даних у форматі NMEA — стандартного текстового протоколу, який використовується для передачі координат, швидкості, напрямку руху та іншої навігаційної інформації.

Інтеграція GPS у мобільну платформу значно розширює її функціональність. Зокрема, пристрій набуває можливості не лише передавати координати користувачу через серійну консоль або веб-інтерфейс, але й самостійно приймати рішення на основі геоданих. Наприклад, система може фіксувати положення в момент виявлення акустичної аномалії або обмежувати власний маршрут у межах заданої геозони. Також реалізація такої логіки дозволяє записувати координати в пам'ять або передавати їх на зовнішній сервер для подальшого аналізу.

Важливим аспектом роботи GPS-модуля є час його ініціалізації — так званий cold start, що може тривати до 30–40 секунд. У сприятливих умовах, за наявності попередньо збережених ефемерид, час доступу до перших координат може знижуватись до 10–15 секунд. В умовах обмеженої видимості (наприклад, у приміщеннях або біля щільних забудов) якість сигналу суттєво знижується, що слід враховувати при експлуатації системи.

Таким чином, GPS-модуль у складі даної роботизованої платформи виконує не лише навігаційну функцію, а й виступає інструментом для

збереження інформації про події в просторі, підвищуючи автономність і контекстну обізнаність системи.

1.5.2. Мікрофонний модуль і принципи виявлення акустичних подій

У системах моніторингу навколишнього середовища важливу роль відіграє здатність пристрою реагувати на звукові сигнали. З цією метою в даній роботі використовується простий, але ефективний компонент — мікрофонний модуль KY-038, що дозволяє фіксувати зміни рівня гучності поблизу пристрою. Основна його функція полягає у виявленні акустичних аномалій, які можуть свідчити про присутність людини, тварини, рух техніки або інші зовнішні події.

Модуль містить конденсаторний мікрофон, підсилювач і компаратор. У базовій конфігурації застосовується аналоговий вихід, який передає безперервний сигнал, пропорційний інтенсивності звуку. Цей сигнал зчитується на аналоговому вході мікроконтролера ESP32. Оскільки ESP32 підтримує кілька ADC-каналів, його апаратна база дозволяє в режимі реального часу проводити аналіз амплітуди звукового сигналу без значного навантаження на процесор.

Алгоритм обробки даних з мікрофонного модуля передбачає безперервне зчитування напруги з аналогового піну, після чого отримане значення порівнюється з попередньо встановленим порогом. У разі перевищення заданого рівня інтенсивності, система активує певну логіку реагування. Це може бути як миттєва зупинка руху дрона, так і ініціація додаткових дій, наприклад, запис GPS-координат, зміна маршруту або виведення повідомлення користувачу.

Завдяки простоті реалізації та швидкій реакції, такий сенсор є ефективним рішенням для проєктів, де не потрібен точний спектральний аналіз звуку, але необхідна базова індикація зовнішньої активності. У подальшому система може бути доповнена більш складними мікрофонами з цифровим виходом або підключенням до спеціалізованих мікросхем, що дозволять розпізнавати тип звукових подій — наприклад, людську мову, крик або технічні звуки.

Таким чином, наявність мікрофонного модуля перетворює мобільну платформу на сенсорну систему, здатну реагувати на навколишні зміни, підвищуючи її інтелектуальність та пристосованість до динамічного середовища.

РОЗДІЛ 2

ПРОЄКТУВАННЯ АПАРАТНО-ПРОГРАМНОЇ ЧАСТИНИ НАЗЕМНОГО ДРОНУ

2.1. Загальна архітектура системи

Побудова ефективної мобільної платформи вимагає реалізації цілісної апаратно-програмної архітектури, в якій усі компоненти — виконавчі механізми, сенсори, блок живлення і система керування — взаємодіють як єдиний функціональний комплекс. У даному проєкті розроблено концепцію наземного дрону, який здатен самостійно пересуватись, аналізувати оточення та реагувати на зміну умов середовища через сенсорну підсистему.

Центром усієї системи виступає мікроконтролер ESP32, який виконує функції координації роботи всіх модулів. Він об'єднує у собі три ключові ролі: обробку сигналів від сенсорів, керування руховою частиною та організацію зв'язку з користувачем. У контексті архітектури ESP32 відіграє роль ядра, що пов'язує периферійні модулі між собою та забезпечує узгоджену взаємодію в реальному часі.

Живлення пристрою здійснюється автономно — від павербанку з виходом 5V. Через макетну плату напруга подається на всі елементи, що забезпечує гнучкість під час монтажу й дозволяє легко змінювати або оновлювати компоненти.

Механічна частина представлена електродвигуном постійного струму, що відповідає за пересування, і сервоприводом, який змінює напрямок руху платформи. Для керування двигуном використовується модуль L298N, що приймає сигнали від ESP32 і дозволяє регулювати напрямок та швидкість обертання. Серводвигун також отримує команди від мікроконтролера, змінюючи кут повороту передньої осі дрона.

Сенсорна підсистема включає два основні модулі: GPS-модуль NEO-6M, який передає координати пристрою, та мікрофонний модуль, що фіксує зміни

інтенсивності навколишнього звуку. Обидва модулі працюють у реальному часі та постійно надсилають дані до ESP32. Обробка цих даних дозволяє системі ініціювати дії — наприклад, автоматичну зупинку при виявленні шуму або фіксацію місцезнаходження на момент події.

Зв'язок з користувачем реалізується через вбудований веб-інтерфейс, який запускається на самому мікроконтролері. Після старту система створює Wi-Fi точку доступу, до якої можна підключитися з будь-якого пристрою з браузером. Через цей інтерфейс користувач має можливість дистанційно керувати рухом, а в майбутньому — переглядати дані з сенсорів і координати.

У цілому архітектура системи орієнтована на модульність і масштабованість: кожен компонент виконує чітко визначену роль, а структура з'єднань дозволяє зручно адаптувати або вдосконалювати платформу. Такий підхід забезпечує надійну роботу в режимі часткової автономії та закладає основу для реалізації більш складних функцій у майбутньому.

2.2. Вибір апаратних компонентів

Формування надійної та функціонально збалансованої апаратної частини мобільної платформи є ключовим етапом у реалізації проєкту. Від правильного підбору компонентів залежить стабільність роботи системи, енергоефективність, масштабованість, а також можливість її подальшого вдосконалення. У ході розробки було прийнято рішення використовувати лише перевірені, доступні за вартістю й сумісні між собою модулі, що добре підтримуються в середовищі Arduino IDE.

Центральною ланкою системи є мікроконтролер ESP32. На відміну від багатьох інших рішень, таких як Arduino Uno чи ESP8266, ESP32 поєднує в собі компактність, високу продуктивність та широкий набір периферійних інтерфейсів. Вбудований модуль Wi-Fi дозволяє уникнути потреби у додаткових адаптерах, а двоядерний процесор із підтримкою багатозадачності забезпечує

плавну роботу як веб-сервера, так і систем обробки даних. Крім того, ESP32 має достатню кількість цифрових і аналогових пінів, що дозволяє одночасно підключити й керувати декількома модулями без необхідності в мультиплексорах.

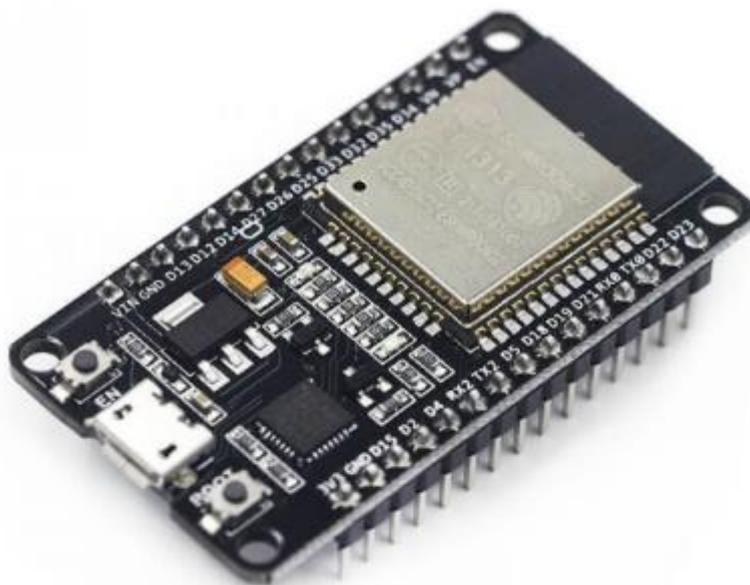


Рисунок 2.1. Плата ESP32

Для реалізації механіки руху платформи використовується модуль керування двигунами L298N. Цей драйвер дозволяє керувати двома двигунами постійного струму, змінюючи як напрямок, так і швидкість обертання за рахунок подачі сигналів із мікроконтролера. Перевагою L298N є здатність працювати з двигунами, які споживають до 2А струму на канал, а також наявність вбудованих захистів. Його використання дозволяє досягти стабільного пересування платформи навіть при змінних навантаженнях або зміні напрямку.

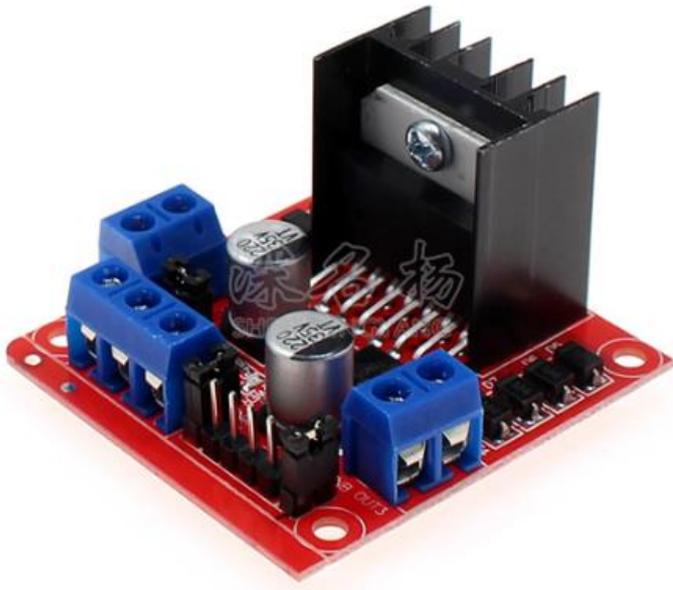


Рисунок 2.2. Модуль керування двигунами L298N

Для зміни напрямку руху дрона реалізовано керування передніми колесами за допомогою серводвигуна. Незважаючи на те, що модель серво в даному проєкті не ідентифікована, її функціонування базується на стандартному принципі керування через широтно-імпульсну модуляцію (PWM). Серводвигун змінює кут повороту передньої осі, дозволяючи дрону здійснювати маневри на місцевості. Простота підключення до ESP32 та широке програмне забезпечення для роботи з сервоприводами робить цей підхід універсальним для проєктів з обмеженим бюджетом.



Рисунок 2.3. Прототип дрона з вбудованим серводвигуном

З метою реалізації просторової орієнтації пристрою було обрано GPS-модуль NEO-6M. Цей модуль є оптимальним за співвідношенням ціна–якість і дозволяє отримувати координати з частотою до 1 Гц. Він підтримує протокол NMEA, що робить його сумісним з Arduino IDE та багатьма іншими середовищами. Завдяки UART-з'єднанню передача даних до ESP32 є стабільною навіть при високій частоті оновлення. Таким чином, пристрій може не лише передавати своє місцезнаходження користувачеві, але й самостійно аналізувати позицію для створення логіки реагування.



Рисунок 2.4. GPS-модуль NEO-6M

Акустичний сенсор KY-038 використовується для моніторингу звукового фону навколо дрона. Він дозволяє визначити зміну рівня шуму, що може бути пов'язано з появою сторонніх об'єктів, руху людей або іншими подіями. Хоча цей сенсор не має високої точності або здатності до спектрального аналізу, його достатньо для базового рівня детекції. Аналоговий сигнал із модуля зчитується на відповідному вході ESP32 і порівнюється із заданим порогом, що дозволяє реалізувати просту, але ефективну логіку реагування на зміну акустичного середовища.

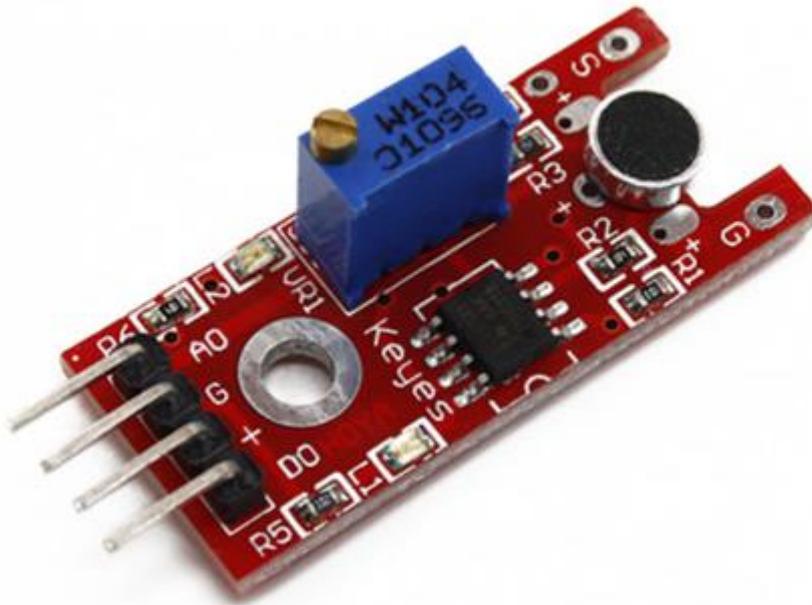


Рисунок 2.5. Акустичний сенсор KY-038

Живлення всієї системи забезпечується через стандартний павербанк із виходом 5V. Це рішення дозволяє зробити платформу мобільною та незалежною від зовнішніх джерел енергії. Павербанк забезпечує стабільне живлення для всіх модулів, включно з ESP32, двигунами та сенсорами, а також забезпечує зручність перезарядки, що є особливо важливим для практичної експлуатації.

Таким чином, вибрані апаратні компоненти забезпечують повний спектр необхідної функціональності: від обробки інформації до руху, позиціонування й моніторингу зовнішніх факторів. Усі модулі сумісні між собою та підтримуються в обраному програмному середовищі, що забезпечує стабільність і гнучкість системи загалом.

2.3. Схема з'єднань та підключення компонентів

Для забезпечення повноцінної взаємодії між окремими апаратними компонентами мобільного дрону реалізовано схему підключення на макетній

платі (breadboard), яка дозволяє легко змінювати конфігурацію під час прототипування та тестування. Макетна плата використовується як базовий вузол маршрутизації сигналів, з'єднуючи модулі через гнучкі провідники із мікроконтролером ESP32, що виступає центральним обчислювальним та керуючим елементом системи.

Мікроконтролер ESP32 отримує живлення безпосередньо від павербанку через порт USB або VCC-пін із стабілізованою напругою 5V. Подібне джерело живлення є зручним рішенням для мобільних платформ, оскільки поєднує портативність із достатньою ємністю для підтримки безперервної роботи системи протягом тривалого часу. Усі підключені модулі живляться через ту саму лінію живлення, яка розподіляється через макетну плату.

Для реалізації руху наземної платформи використовується електродвигун постійного струму, керований за допомогою драйвера L298N. Вихідні пін-коди ESP32 підключаються до відповідних входів драйвера IN1 та IN2, що дозволяє регулювати напрямок обертання двигуна. Додатково забезпечується керування швидкістю через подачу сигналу PWM на пін ENA. Виходи OUT1 і OUT2 модуля L298N з'єднані безпосередньо з електродвигуном, утворюючи повноцінний ланцюг приводу.

Серводвигун підключено до одного з PWM-каналів ESP32, оскільки серво потребує сигналу широтно-імпульсної модуляції для точного регулювання кута повороту. В даному проєкті він відповідає за управління передньою віссю платформи та зміною напрямку руху. PWM-сигнал формується на частоті 50 Гц, а кут обертання контролюється шляхом зміни тривалості імпульсу від 500 до 2400 мкс, що є стандартним діапазоном для більшості сервомоторів.

Мікрофонний модуль підключається до одного з аналогових входів ESP32, що дає змогу проводити безперервне зчитування рівня напруги, пропорційної гучності навколишнього середовища. У коді реалізується постійна перевірка значень ADC (аналогово-цифрового перетворення) та їх порівняння з порогом, визначеним емпірично.

GPS-модуль NEO-6M, відповідальний за геопозиціонування, підключено до ESP32 через UART-інтерфейс. Передача даних відбувається через пари TX/RX відповідно до логіки: TX модуля під'єднується до RX ESP32, і навпаки. Комунікація відбувається зі швидкістю 9600 бод, що є типовим значенням для даних типів модулів і забезпечує стабільне оновлення координат із частотою до одного разу на секунду. Для обробки отриманих даних використовується бібліотека TinyGPS++, яка дозволяє витягати корисну інформацію (широту, довготу, швидкість, час) у зручному форматі.

Усі елементи з'єднані через загальний контур землі (GND), що забезпечує стабільність роботи, запобігає появі різниці потенціалів між модулями та мінімізує ризик перешкод у сигнальних лініях. Надійний контакт усіх GND-виводів є критично важливим, оскільки порушення загальної землі може спричинити хаотичну поведінку сервоприводів, неправильні показники з сенсорів або нестабільну роботу модуля зв'язку.

Сумарно схема з'єднань забезпечує чіткий розподіл функцій і стабільну взаємодію між усіма компонентами дрону. Структурна прозорість підключень дозволяє без труднощів розширювати функціональність проєкту — наприклад, додавати нові сенсори, дисплеї, модулі зберігання даних або бездротової

передачі, не змінюючи базову архітектуру.

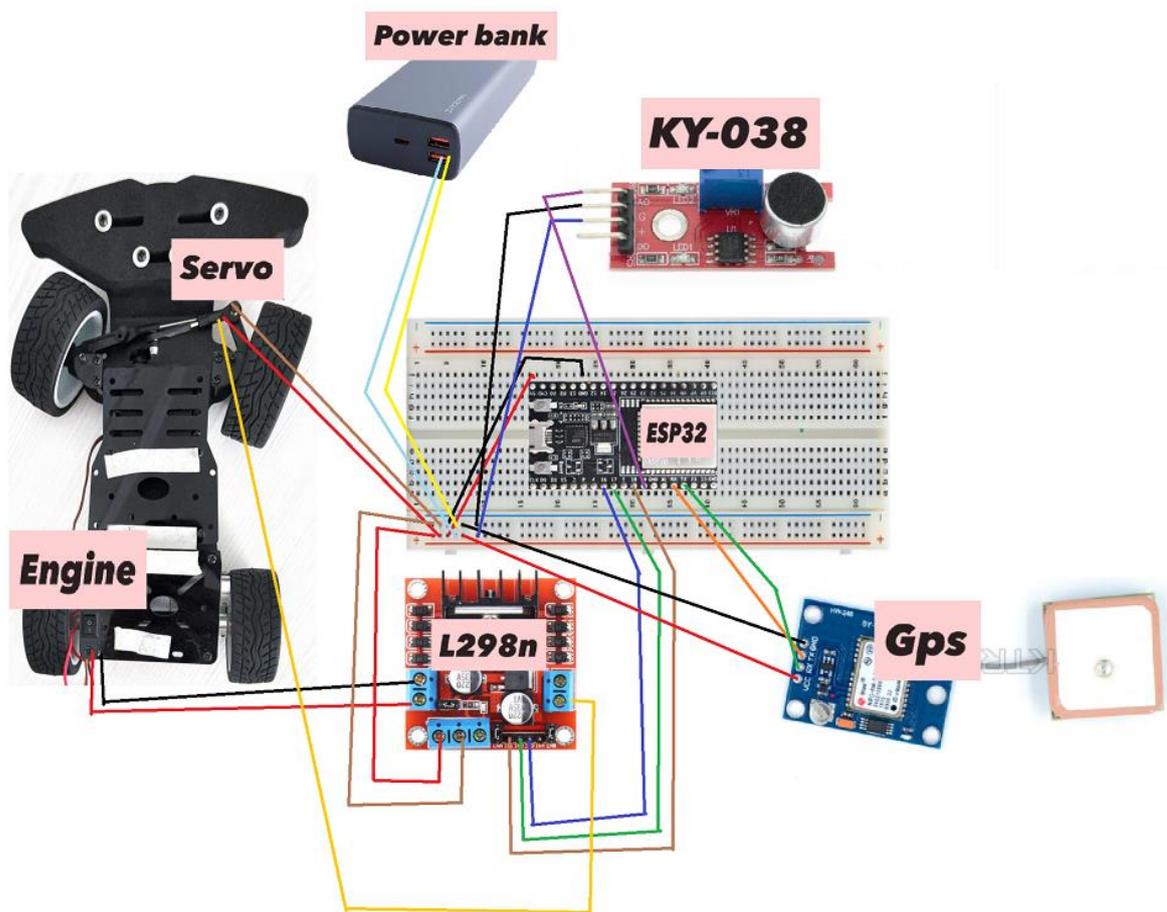


Рис. 2.6. Зображення схеми підключень

2.4. Програмна архітектура та структура коду

Програмне забезпечення для функціонування мобільної платформи реалізовано в середовищі Arduino IDE з використанням відповідних бібліотек для ESP32. Архітектура програмного коду побудована на принципах модульності, де кожен логічний блок виконує окрему функцію, а їхня взаємодія забезпечує цілісну роботу системи. Такий підхід дозволяє легко масштабувати проект, розширювати його функціональність або вносити зміни без порушення стабільності основного алгоритму.

На етапі ініціалізації відбувається налаштування апаратної частини — задаються режими роботи пінів мікроконтролера (INPUT, OUTPUT),

запускається PWM для керування серводвигуном, виділяються таймери, необхідні для стабільної роботи сервомеханіки, та ініціалізується послідовний порт для налагодження. Особлива увага приділяється конфігурації Wi-Fi. ESP32 запускається у режимі точки доступу (Access Point), створюючи незалежну мережу, до якої може підключитися користувач. За потреби також реалізована можливість підключення до наявної Wi-Fi мережі в режимі клієнта (STA), однак основний сценарій передбачає автономну локальну взаємодію без участі зовнішнього інтернет-з'єднання.

Одразу після налаштування мережевих параметрів ініціалізується вбудований веб-сервер. Сервер слухає стандартний порт 80 і обробляє HTTP-запити, що надходять від браузера користувача. Найбільш важливими є два маршрути: головна сторінка інтерфейсу ("/") та обробник команд руху ("/moveDirection"). Веб-інтерфейс, написаний на HTML із використанням CSS для стилізації та JavaScript для логіки, вбудовано безпосередньо в пам'ять мікроконтролера у вигляді гав-рядка, що дозволяє працювати без окремих файлів чи зовнішнього сервера.

Клієнтський інтерфейс дозволяє відправляти команди до сервера за допомогою HTTP GET-запитів. Зокрема, при натисканні на одну з кнопок керування (вперед, назад, вліво, вправо або стоп), генерується відповідний запит, який містить параметр напрямку. Серверна частина обробляє цей запит, розпізнає переданий параметр, і викликає відповідну функцію керування рухом. Це дає змогу реалізувати реакцію в режимі реального часу без значної затримки між натисканням кнопки і реакцією пристрою.

Функції керування реалізовані окремо для кожного напрямку руху. Вони змінюють стани пінів, що відповідають за драйвер двигуна, та задають положення серводвигуна через метод *myservo.write()*. Наприклад, при русі вперед двигун обертається у прямому напрямку, а серво встановлюється у нейтральне положення (90 градусів). Для повороту серво змінює свій кут

(наприклад, 50 для лівого повороту або 130 для правого), при цьому двигун продовжує обертання у відповідному напрямку.

Окремим напрямком, передбаченим у подальшому розвитку системи, є інтеграція обробки GPS-координат і акустичних сигналів. Після зчитування даних із GPS-модуля, координати можуть або передаватися користувачу, або використовуватися для внутрішньої логіки — наприклад, зупинки при виході за межі геозони. Аналогічно, рівень сигналу з мікрофонного модуля може оброблятися як тригер для ініціації подій. Усе це може бути реалізовано як у вигляді фонових процесів у *loop()*, так і через апаратні переривання або окремі задачі у випадку використання FreeRTOS (який підтримується ESP32).

Таким чином, програмна архітектура дрона поєднує в собі простоту реалізації з гнучкістю розширення. Вона дозволяє ефективно взаємодіяти з апаратною частиною, реалізовувати інтерфейс користувача без додаткових серверів або додатків і забезпечує стабільну реакцію на події в режимі реального часу.

2.5. Реалізація логіки керування рухом

Функціональність мобільної платформи базується на здатності точно та оперативно реагувати на команди, що надходять від користувача через веб-інтерфейс. Для цього у структурі програмного забезпечення реалізовано набір функцій, що відповідають за керування виконавчими механізмами: електродвигуном і серводвигуном. Ключова роль у реалізації цієї логіки належить обробці HTTP-запитів, які кодують напрямок руху в параметрах.

Загальна ідея побудови логіки полягає у зміні стану цифрових виходів ESP32 відповідно до вхідної команди. Коли користувач натискає відповідну кнопку в інтерфейсі, JavaScript-скрипт формує HTTP GET-запит, наприклад */moveDirection?dirr=up*. Мікроконтролер, отримавши цей запит, викликає відповідну функцію — *moveForward()*, *moveLeft()* тощо. Кожна з функцій містить

низку інструкцій для керування пін-кодами, підключеними до драйвера двигуна L298N, а також команду для встановлення положення сервоприводу через PWM.

При русі вперед, функція *moveForward()* встановлює один з пінів двигуна в стан HIGH, а інший — у LOW, що забезпечує обертання валу у напрямку прямого ходу. Одночасно серводвигун виставляється в нейтральне положення (90 градусів), яке відповідає прямолінійному русі. Такий підхід дозволяє не лише фізично рухатись вперед, але й мінімізує механічне тертя, зменшуючи навантаження на двигун.

Для реалізації повороту система використовує комбіновану зміну керування: серводвигун змінює свій кут (наприклад, до 50 градусів для повороту вліво або 130 — вправо), при цьому основний двигун продовжує працювати у напрямку обертання. Така комбінація дозволяє здійснювати маневрування без складної системи керування коліс, імітуючи поведінку транспортного засобу із переднім кермовим приводом.

Функція *moveStop()* реалізує зупинку дрона шляхом подачі логічного нуля (LOW) на обидва керуючих пін-коди, що зупиняє двигун. Крім того, серво повертається у початкове положення, готуючи систему до наступної дії. Така поведінка дозволяє забезпечити безпечне гальмування та мінімізувати випадковий рух у разі втрати з'єднання або припинення команди.

Усі функції руху супроводжуються незначною програмною затримкою (delay), яка вводиться для стабілізації роботи двигуна та плавності переходів між станами. Впровадження циклічної передачі команд (через інтервал, наприклад, 500 мс) дозволяє платформі залишатись у вибраному стані руху, поки утримується кнопка на інтерфейсі, що нагадує принцип роботи фізичних контролерів.

Загальна логіка керування сформована з урахуванням мінімальної кількості апаратних ресурсів, високої надійності та простоти реалізації. Вона дозволяє досягти базового рівня маневреності, необхідного для навігації у відкритому просторі або приміщенні, та формує фундамент для подальшого

вдосконалення, зокрема введення адаптивного керування на основі даних із сенсорів або розширених сценаріїв руху.

2.6. Реалізація Wi-Fi з'єднання та веб-інтерфейсу

Одним із головних завдань у розробці системи є забезпечення надійної, стабільної та зручної для користувача взаємодії з мобільною платформою. У межах цього проекту реалізовано підхід, що не вимагає підключення до глобальних мереж або додаткових маршрутизаторів — управління здійснюється повністю автономно завдяки можливостям ESP32 створювати власну Wi-Fi мережу.

Після подачі живлення мікроконтролер переходить у режим точки доступу (Access Point), ініціалізуючи власну Wi-Fi мережу з унікальним іменем (SSID) та паролем, визначеним у коді. Така точка доступу працює в режимі ізольованої локальної мережі, не потребує зовнішнього підключення до Інтернету та дозволяє безпосередньо взаємодіяти з користувацькими пристроями — смартфоном, планшетом або ноутбуком. Підключившись до цієї мережі, користувач уводиться у браузері локальну IP-адресу — це 192.168.1.101 — і потрапляє на головну сторінку керування.

Інтерфейс веб-сторінки реалізовано з використанням стандартних технологій HTML, CSS і JavaScript. Завдяки цьому сторінка є кросбраузерною та адаптивною, не потребує спеціальних клієнтів чи мобільних застосунків, що значно спрощує процес керування. Візуально інтерфейс містить кнопки для кожного напрямку руху: вперед, назад, вліво, вправо та зупинка. Дизайн кнопок забезпечує чіткий зворотній зв'язок при натисканні (зміна кольору або стилю), що покращує користувацький досвід.

При взаємодії з кнопками генеруються HTTP GET-запити — кожна дія відповідає певному параметру, який додається до URL. Наприклад, при натисканні кнопки «вперед» браузер надсилає запит за адресою

192.168.1.101/moveDirection?dirr=up. Мікроконтролер, що виконує роль сервера, приймає запит і викликає відповідну функцію з керування рухом. Обробка запитів відбувається в основному циклі програми *loop()*, паралельно з іншими процесами, зокрема зчитуванням даних із сенсорів.

Щоб забезпечити безперервність руху при утриманні кнопки, реалізовано повторну передачу команд з інтервалом у 500 мілісекунд. Це дає змогу дрону зберігати напрямок руху до моменту, поки користувач не припинить натискання. Таким чином, поведінка нагадує класичну роботу геймпада або пульта дистанційного керування, що значно полегшує управління в режимі реального часу.

Додатковою перевагою такого підходу є повна автономність та захищеність від зовнішніх мережових впливів. Система не залежить від стабільності інтернет-з'єднання, а передача даних відбувається безпосередньо в межах локального з'єднання. Це дозволяє застосовувати пристрій у польових умовах, технічних зонах, лабораторіях, де підключення до Wi-Fi або мобільного інтернету неможливе чи небажане.

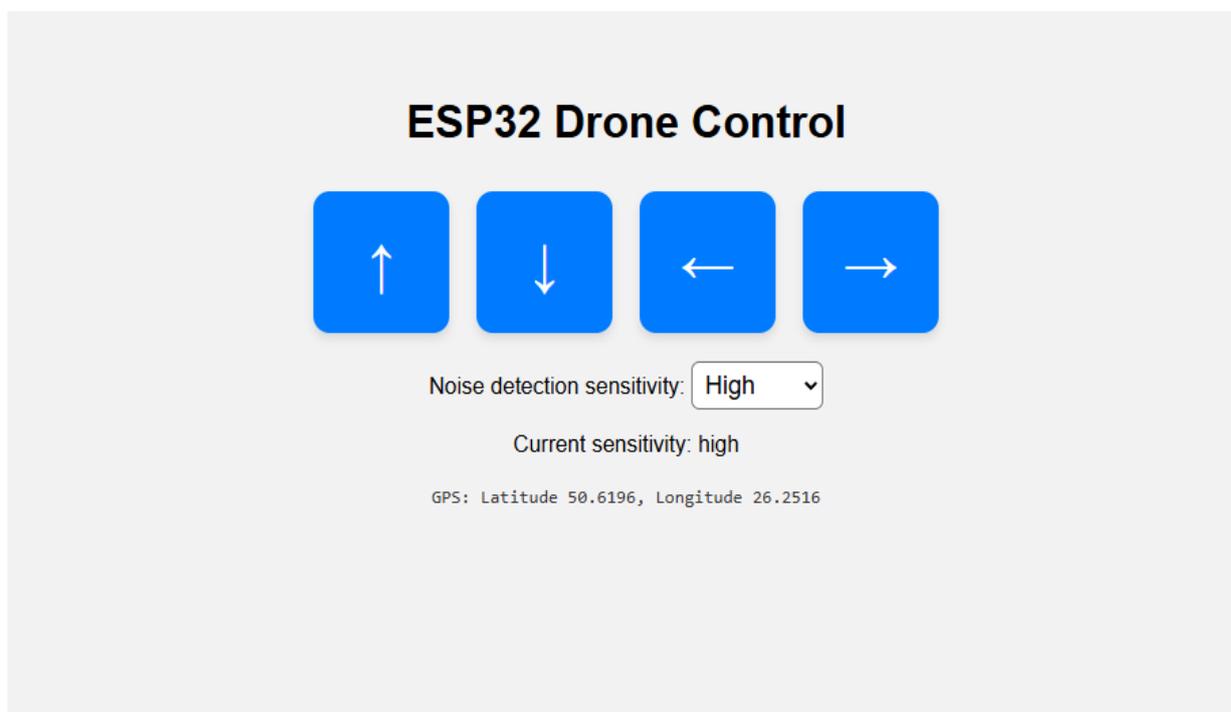


Рис. 2.7. Вигляд веб-інтерфейсу

Таким чином, реалізація Wi-Fi-з'єднання на базі ESP32 у поєднанні з інтегрованим веб-інтерфейсом забезпечує ефективну, просту й інтуїтивну модель взаємодії з мобільною платформою без використання додаткового обладнання чи програмного забезпечення. Такий підхід може бути адаптований для багатьох інших пристроїв з обмеженими ресурсами, які потребують простого механізму дистанційного контролю.

2.7. Інтеграція сенсорів GPS та мікрофону

Одним із ключових напрямів у проектуванні мобільної платформи є підвищення її обізнаності про просторове розташування та навколишнє середовище. Для цього до складу системи інтегровано два основні сенсори: GPS-модуль для супутникової навігації та мікрофонний модуль для фіксації акустичних аномалій. Їхня взаємодія з мікроконтролером ESP32 реалізована на апаратному рівні через відповідні інтерфейси — UART та аналогові входи — і керується за допомогою бібліотек та програмних обробників.

GPS-модуль NEO-6M підключається до UART-порту ESP32, що забезпечує стабільний і надійний обмін даними в реальному часі. Координати надходять у форматі NMEA, з якого шляхом парсингу виділяються широта, довгота, час, висота, швидкість та інші параметри. Для зручності в роботі використовується бібліотека TinyGPS++ — вона дозволяє легко витягати дані та інтегрувати їх у систему логіки управління. Наприклад, на основі отриманих координат пристрій може визначити, чи перебуває він у дозволеній зоні, та при виході за її межі ініціювати аварійну зупинку. Подібна функціональність може бути використана як елемент віртуального бар'єру або електронного периметра (геозони), що особливо корисно у випадках патрулювання, охорони або картографування територій.

Окрім цього, координати можуть зберігатися в пам'яті пристрою чи передаватися в консоль користувача як лог-фіксація подій. Наприклад, при

виявленні аномалії система автоматично записує місце її виникнення, що дозволяє користувачу згодом проаналізувати активність у певних точках маршруту. Збереження координат дає змогу створювати маршрутну карту або навіть будувати аналітику поведінки дрона на місцевості.

Що стосується мікрофонного модуля, він підключається до одного з аналогових входів ESP32 та забезпечує безперервне зчитування рівня гучності. Сигнал, що надходить із сенсора, являє собою напругу, пропорційну інтенсивності навколишнього звуку. У програмному коді реалізовано алгоритм порівняння цього значення з заздалегідь визначеним порогом. При його перевищенні ініціюється відповідна подія — найчастіше зупинка дрона з одночасним записом координат і виведенням повідомлення у консоль.

Подібна логіка дозволяє вважати платформу умовно чутливою до навколишнього середовища, що відкриває нові сценарії її використання. Наприклад, вона може бути застосована в охоронних цілях для виявлення стороннього втручання, в екологічному моніторингу для фіксації аномальних звуків, або в рятувальних операціях для реагування на голос чи шум, що може вказувати на місце перебування людини.

У перспективі система може бути доповнена алгоритмами спектрального аналізу сигналу, які дозволять класифікувати тип джерела шуму — наприклад, розрізняти людський голос, звук автомобіля або постріл. Додатково можлива реалізація визначення напрямку звуку за допомогою масиву мікрофонів (beamforming), що дозволить точніше локалізувати джерело сигналу та скоригувати маршрут руху дрона відповідним чином.

Інтеграція GPS-модуля та мікрофонного сенсора значно розширює функціональність пристрою, перетворюючи його з простої рухомої платформи на адаптивну сенсорну систему, здатну взаємодіяти з оточенням і приймати рішення на основі змін у навколишньому середовищі. Такий підхід відкриває широкі можливості для подальшого розвитку платформи в напрямку

повноцінної автономії, інтелектуального аналізу даних і взаємодії з іншими інформаційними системами.

2.8. Інструкція зі збирання та налагодження системи

Ефективна реалізація мобільної платформи передбачає не лише правильний підбір апаратних компонентів, а й їхнє точне з'єднання, коректне програмування та тестування функціональності в реальних умовах. У цьому підпункті описано послідовність дій, необхідну для повного збирання та запуску системи, а також налагодження її базових функцій.

Першим етапом є фізичне розміщення усіх електронних модулів на макетній платі або корпусній основі дрона. Центральне місце займає мікроконтролер ESP32, до якого необхідно під'єднати інші компоненти відповідно до попередньо підготовленої схеми з'єднань. До цифрових пінів ESP32 підключаються виводи драйвера L298N, що відповідають за керування напрямком обертання моторів. Окремий вихід PWM використовується для управління серводвигуном, що змінює напрям руху платформи. GPS-модуль під'єднується через інтерфейс UART (пари TX/RX), а мікрофонний модуль — до одного з аналогових входів мікроконтролера.

Для стабільного живлення усіх компонентів рекомендується використовувати стандартний павербанк із виходом 5V, який підключається до ESP32 через USB-кабель. У разі потреби допускається живлення через стабілізатор напруги, підключений до виводу VIN. Всі модулі мають бути з'єднані по GND для забезпечення єдиного потенціалу, що критично важливо для правильної роботи аналогових сигналів та уникнення паразитних перешкод.

Наступним кроком є програмування мікроконтролера. Для цього необхідно відкрити середовище розробки Arduino IDE, вибрати плату ESP32, підключити пристрій до ПК за допомогою USB та завантажити прошивку. Код проекту має містити блоки ініціалізації, створення точки доступу Wi-Fi, запуску

веб-сервера, обробки HTTP-запитів, а також функції керування рухом, серводвигуном і логікою обробки сенсорних даних. У разі першого підключення може знадобитися встановлення додаткових бібліотек (ESPAsyncWebServer, TinyGPS++, ESP32Servo та інші).

Після завантаження коду ESP32 автоматично створить власну Wi-Fi мережу. Користувач повинен підключитися до цієї мережі з будь-якого пристрою, що має модуль Wi-Fi та браузер, і ввести в адресний рядок IP-адресу 192.168.1.101. Після цього відкриється веб-інтерфейс керування, в якому відображаються інтерактивні кнопки для напрямків руху та зупинки.

Фінальним етапом є тестування системи. У першу чергу перевіряється коректність реакції платформи на кожну з команд: рух уперед, назад, повороти та зупинка. Серводвигун повинен точно змінювати кут, а двигуни — стабільно запускатися у відповідному напрямку. Далі слід перевірити функцію виявлення звуку: при створенні гучного клацання, хлопку або голосного звуку система має зафіксувати зміну сигналу з мікрофону та ініціювати відповідну подію - зупинку руху. У разі інтеграції GPS-модуля доцільно перевірити оновлення координат та їх правильне відображення у серійній консолі.

Дотримання описаної послідовності дій дозволяє забезпечити стабільну роботу системи з мінімальними витратами на відладку та модифікації. Крім того, інструкція може бути використана як основа для подальшого створення повноцінної документації або технічного керівництва до експлуатації пристрою.

2.9. Типові проблеми реалізації та шляхи їх усунення

У процесі розробки та тестування мобільної платформи на базі ESP32 було виявлено низку характерних проблем, що виникали як на етапі збирання апаратної частини, так і в процесі програмної інтеграції модулів. Аналіз цих проблем і своєчасне знаходження шляхів їхнього усунення відіграли важливу роль у стабілізації роботи системи та підвищенні її надійності.

Однією з перших складностей стала нестабільність живлення. У тестових умовах використовувалися різні джерела напруги — від комп'ютерного USB-порту до зовнішніх адаптерів. Було помічено, що при змінному навантаженні або запуску двигунів відбувалося незначне падіння напруги, що призводило до самовільного перезавантаження ESP32 або порушення зв'язку з сенсорами. Проблема вирішувалася через використання якісного павербанку з перевіреною стабільністю вихідної напруги, а також шляхом додавання конденсаторів на лінії живлення для згладжування пульсацій.

Іншою поширеною проблемою було некоректне зчитування координат з GPS-модуля. Виявилось, що при неправильному підключенні TX/RX контактів обмін даними не відбувається, незважаючи на фізичну справність обох пристроїв. Причиною було стандартне правило інверсії: передавальний пін TX одного модуля має бути підключений до приймального RX іншого. Після інверсії з'єднань та підтвердження відповідності швидкості передачі (9600 бод) зв'язок було повністю відновлено.

Окремо слід відзначити проблему, пов'язану з роботою сервомотора при запуску системи. У деяких випадках, одразу після подачі живлення, сервопривід різко відхилився у випадковому напрямку, що створювало ризик механічного пошкодження. Причиною виявилася відсутність чітко визначеної початкової команди. Після додавання у функцію *setup()* ініціалізації сервомотора з установкою у центральне положення (90 градусів), а також затримки для стабілізації PWM-сигналу, проблема була повністю усунена.

Ще один виклик стосувався стабільності Wi-Fi-з'єднання. При спробах роботи в режимі клієнта (STA), тобто підключення до зовнішнього маршрутизатора, виявлялася чутливість до перешкод і загальне зниження швидкості реакції. Це пояснюється тим, що в зоні з низьким рівнем сигналу або перевантаженою мережею ESP32 втрачав зв'язок або працював із затримками. Вирішенням стало остаточне переключення на режим точки доступу (Access

Point), що дозволяє уникати зовнішніх чинників та забезпечує стабільний зв'язок на короткій дистанції у межах локальної мережі.

Загалом, усі виявлені технічні труднощі не були критичними, однак їхнє системне опрацювання дозволило досягти високої надійності функціонування пристрою. Аналіз типових помилок дав змогу сформулювати чіткі рекомендації щодо збирання та налаштування системи, які можуть бути використані не лише в межах цього проєкту, а й для інших подібних платформ.

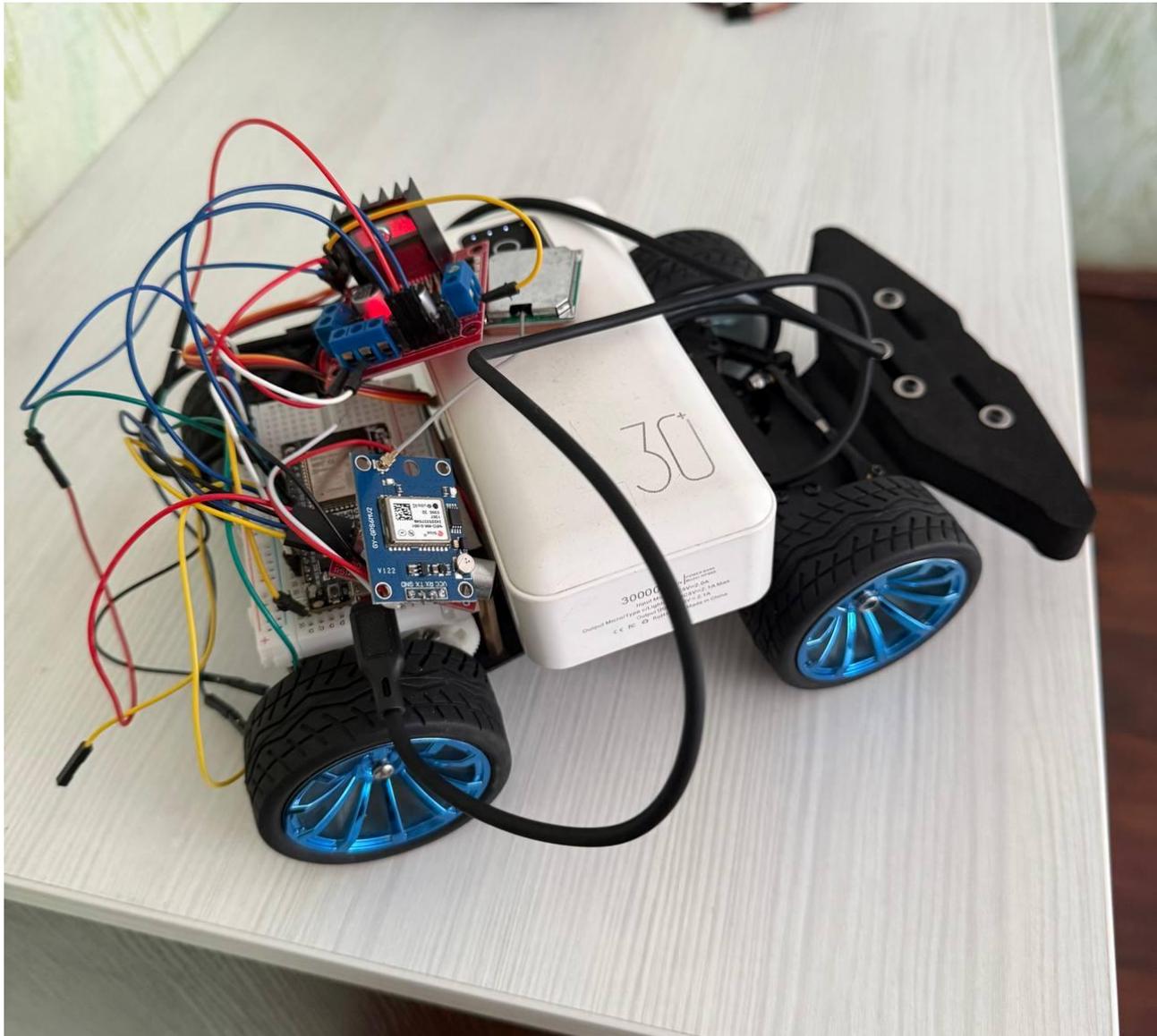


Рисунок 2.8. Зібраний пристрій

Фінальна збірка мобільної платформи виконана з використанням доступних компонентів та макетного з'єднання, що дозволило зосередитись на

тестуванні функціональності, а не на зовнішньому оформленні. Незважаючи на базовий вигляд та відкрите компонування, система повністю демонструє заявлену логіку взаємодії між модулями й відповідає вимогам проєкту. Конструкція залишається придатною для подальших доопрацювань, зокрема для впорядкування проводки та створення корпусу.

РОЗДІЛ 3

ТЕСТУВАННЯ І ОЦІНКА РЕЗУЛЬТАТІВ РОБОТИ СИСТЕМИ

3.1. Умови проведення випробувань

Комплексне тестування є невід’ємною складовою процесу розробки будь-якої інтерактивної технічної системи. Воно дозволяє не лише перевірити відповідність пристрою функціональним вимогам, а й оцінити його працездатність у наближених до реальних умовах експлуатації. У випадку мобільної платформи, орієнтованої на виявлення акустичних аномалій і передачу координат, тестування охоплювало як апаратну, так і програмну частини, а також взаємодію з користувачем.

Перший етап випробувань відбувався в контрольованому лабораторному середовищі. Це дало змогу верифікувати базову функціональність пристрою — правильність підключення всіх модулів, адекватну реакцію на вхідні команди та коректну обробку сигналів. Зокрема, було перевірено запуск ESP32, створення точки доступу Wi-Fi, підключення користувацького пристрою, доступ до веб-інтерфейсу та виконання команд руху. Особливу увагу приділяли реакції на сценарії з утриманням кнопок, перемикання напрямків, а також перевірки стабільності PWM-сигналів для серводвигуна.

Наступним етапом стали випробування на відкритому просторі. Тестування проводилося за різних погодних і акустичних умов: у денний час на вулиці з нерівним рельєфом, а також у вечірніх умовах зі зменшеним рівнем освітлення. Перевірялися стабільність зв’язку між ESP32 і пристроєм користувача, дальність впевненого з’єднання, швидкість обробки HTTP-запитів і чутливість до зміни рівня шуму. Було зафіксовано, що при прямій видимості між дроном і оператором система демонструє стабільну роботу в радіусі 10–15 метрів.

Умови тестування були спеціально ускладнені для перевірки стійкості системи до зовнішніх впливів. Зокрема, в полігонних умовах створювалися

навмисні перешкоди: рух перед дроном сторонніх об'єктів, зміна рівня освітлення, механічні удари в корпус платформи, а також різкі звуки — наприклад, хлопки в долоні, удари по столу, голосні команди. Ці фактори дозволили оцінити ефективність реакції системи на зміну акустичного середовища, зокрема момент виявлення аномалій та ініціацію зупинки руху.

Живлення системи під час усіх етапів тестування здійснювалося від автономного джерела — павербанку з ємністю 30000 мА. Це дозволяло уникнути залежності від стаціонарних мереж і забезпечити тривалу безперервну роботу. Перевагою такого підходу була можливість переміщення пристрою у реальних умовах — на подвір'ї, у полі, на доріжках або в складських приміщеннях — без потреби у повторному перезавантаженні чи перепідключенні.

У цілому, умови тестування охопили широкий спектр сценаріїв: як контрольоване середовище з прогнозованими параметрами, так і динамічні зміни, наближені до потенційних умов реального використання. Це дозволило об'єктивно оцінити рівень стабільності роботи системи, її чутливість до зовнішніх факторів, швидкість реакції на події та зручність взаємодії з інтерфейсом керування. Отримані результати стали базою для подальшого вдосконалення як апаратної, так і програмної частин пристрою.

3.2. Перевірка функціональності керування

Ключовою особливістю створеної системи є її здатність до надійного та інтуїтивного дистанційного керування, реалізованого через веб-інтерфейс. Це дозволяє оператору взаємодіяти з пристроєм безпосередньо через браузер із будь-якого пристрою, що підтримує Wi-Fi. Функціональність керування тестувалася з акцентом на швидкість реакції системи, точність виконання команд і стійкість до аварійних ситуацій.

Після підключення користувача до точки доступу, створеної ESP32, у браузері відкривається інтерфейс керування, що містить кнопки для основних

напрямків руху — вперед, назад, вліво, вправо, а також зупинка. Вся взаємодія відбувається через передачу HTTP GET-запитів, які обробляються вбудованим сервером на ESP32. Під час тестування була підтверджена висока чутливість інтерфейсу: кожна дія користувача миттєво викликала відповідну зміну стану виконавчих механізмів дрона.

У режимі тривалого натискання системою реалізується механізм повторної передачі команд із заданим інтервалом (близько 500 мс), що дозволяє підтримувати обраний напрямок руху до моменту відпускання кнопки. Це нагадує принцип роботи традиційних джойстиків або пультів дистанційного керування. Водночас система не допускала надмірного дублювання запитів і уникала перевантаження, демонструючи стабільність навіть при швидких багаторазових натисканнях.

У рамках випробувань також були змодельовані аварійні ситуації. Зокрема, перевірялися сценарії втрати з'єднання під час активного руху, примусового перезавантаження ESP32, а також повного зникнення живлення. У всіх випадках система проявила себе надійно: після зникнення сигналу пристрій автоматично зупиняв двигун, запобігаючи мимовільному переміщенню платформи. Це стало можливим завдяки логіці безпечної зупинки, яка реалізована у коді — при відсутності нових команд упродовж певного інтервалу рух блокується, а серводвигун повертається у нейтральне положення.

Окрему увагу приділяли часам відгуку на команду — від моменту натискання до початку руху. Навіть при роботі у зоні з потенційними перешкодами (наприклад, бетонні стіни або побутові Wi-Fi мережі) час відгуку залишався в межах прийнятних — від 80 до 150 мс, що не створювало дискомфорту під час взаємодії. Це свідчить про якісну реалізацію внутрішнього веб-сервера, правильне балансування задач у мікроконтролері та оптимальне використання ресурсів ESP32.

Таким чином, тестування функціональності керування підтвердило, що система здатна працювати стабільно, без збоїв та із передбачуваною поведінкою

навіть в умовах підвищеного навантаження. Застосована схема взаємодії через локальний веб-інтерфейс продемонструвала високу ефективність і зручність у використанні, що є важливою перевагою в умовах, де недоступна традиційна інфраструктура зв'язку або інтернет-покриття.

3.3. Робота модуля GPS

Одним із ключових функціональних елементів мобільної платформи є модуль глобального позиціонування (GPS), який забезпечує визначення географічних координат у режимі реального часу. У межах проекту використовувався модуль NEO-6M — надійне та економічно доцільне рішення, сумісне з мікроконтролером ESP32 та середовищем Arduino IDE. Основна мета тестування полягала у перевірці здатності пристрою визначати місцезнаходження з прийнятною точністю та частотою оновлення.

Першочергово модуль GPS тестувався на відкритій місцевості з прямою видимістю до неба. В таких умовах час першого захоплення сигналу (так званий “time to first fix”, TTFF) становив приблизно 20–30 секунд, що відповідає заявленим характеристикам виробника. Після початкової фіксації координат модуль переходив у стабільний режим роботи, з оновленням даних з інтервалом близько однієї секунди. У потоці даних, що надходив через UART, містилися відомості про широту, довготу, кількість видимих супутників, точність, швидкість та інші супутні параметри.

З технічної точки зору, обробка даних координат здійснювалася за допомогою бібліотеки TinyGPS++, яка дозволяє легко витягати ключову інформацію з NMEA-повідомлень. У тестовому режимі ці дані виводилися у серійну консоль Arduino IDE, що давало змогу в режимі реального часу відстежувати положення дрона. Такий підхід є достатнім для прототипу, однак у перспективі передбачається інтеграція цих координат у веб-інтерфейс або їх буферизація з подальшим переглядом користувачем.

Під час випробувань була зафіксована середня точність позиціонування в межах 2–3 метрів при наявності стабільного сигналу. Це відповідає вимогам більшості побутових або охоронних застосувань, де надвисока точність не є критичною. Однак варто зазначити, що точність суттєво погіршувалася при наближенні до висотних будівель, дерев, або в умовах ущільненої забудови — так званий “міський каньйон”. У таких випадках кількість доступних супутників знижувалась, що впливало як на стабільність сигналу, так і на частоту оновлення координат.

У приміщенні робота модуля GPS ускладнювалася — сигнал втрачався повністю або координати оновлювалися з помітними затримками. Це є типовим обмеженням для таких типів модулів, які не мають підтримки додаткових протоколів (як-от A-GPS або підтримка зовнішніх антен), і не є критичним недоліком, якщо система переважно працює на відкритому просторі.

Узагальнюючи результати випробувань, можна констатувати, що обраний GPS-модуль повністю виконав свої функції у межах завдань, поставлених у цьому проєкті. Його інтеграція до системи забезпечила можливість не лише визначати місцезнаходження платформи, але й закладати логіку дій на основі просторової прив’язки. Така функціональність відкриває потенціал для реалізації маршрутного патрулювання, визначення геозон, ведення журналу подій за координатами тощо.

3.4. Реакція на акустичні аномалії

Інтеграція мікрофонного модуля у склад мобільної платформи дозволила реалізувати базову здатність пристрою реагувати на зміни акустичного фону в реальному часі. Така функціональність є важливою умовою для використання системи в охоронних, пошукових або моніторингових завданнях, де виявлення аномального звуку може свідчити про присутність сторонніх об’єктів або виникнення потенційної загрози.

У проєкті використано мікрофонний модуль типу KY-038, який передає аналоговий сигнал пропорційний рівню гучності середовища. Підключення до ESP32 реалізовано через один із аналогових пінів, що дозволяє у циклі зчитувати значення напруги та проводити його порівняння з наперед заданим порогом. У ході тестування був підібраний робочий поріг, який дозволяє відфільтрувати фоновий шум і зафіксувати лише значущі коливання.

Випробування проводилися в умовах змінного звукового середовища. Було змодельовано декілька типових акустичних сценаріїв: розмови на відстані 1–2 метри від платформи, різкий свист, клацання пальцями, удари по дерев'яній поверхні. У кожному з випадків система стабільно реєструвала зміну звуку, і при перевищенні порогового значення виконувалась запрограмована дія — зупинка дрона. Такий підхід дозволив протестувати роботу детектора на предмет хибних спрацьовувань та чутливості.

Усі зафіксовані події дублювалися у серійній консолі, що дозволяло розробнику відстежувати характер і частоту реакцій у реальному часі. Повідомлення містили інформацію про поточне значення сигналу та факт перевищення порогу. Це спростило процес калібрування та верифікації поведінки модуля в різних умовах — як у приміщенні, так і на відкритому повітрі.

Варто зазначити, що поточна реалізація використовує простий пороговий метод обробки, який не враховує частотні характеристики сигналу. Це означає, що система реагує на будь-який гучний звук без розпізнавання його джерела чи природи. Однак така реалізація цілком достатня для базового реагування, наприклад, у разі несанкціонованого проникнення або раптової події у полі зору пристрою.

У перспективі планується впровадження більш складних алгоритмів аналізу звукового сигналу. Зокрема, застосування спектрального аналізу (швидке перетворення Фур'є) дозволить класифікувати джерела шуму за частотними ознаками — наприклад, відрізнити людський голос від механічного

шуму або пострілу. Крім того, використання масиву мікрофонів (microphone array) відкриє можливість визначати напрямок джерела звуку, що дозволить платформі автономно орієнтуватись на об'єкт, що викликав аномалію.

Таким чином, навіть у своїй поточній реалізації система показала здатність реагувати на зміну звукового середовища з прийнятною точністю і стабільністю. Це підтверджує ефективність обраного технічного рішення для використання в умовах, де акустичне виявлення є важливим фактором для реагування або прийняття рішень платформою.

3.5. Сценарії використання та поведінка системи

Для оцінки прикладної цінності розробленої системи було змодельовано низку ситуацій, які максимально наближають поведінку мобільної платформи до умов її реального застосування. Сценарії тестування охоплювали типові випадки, що можуть виникнути під час охоронної діяльності, моніторингу об'єктів або локального патрулювання території. Такий підхід дозволив не лише перевірити технічну готовність системи, а й проаналізувати ефективність логіки взаємодії сенсорів, зв'язку та програмного забезпечення.

Один із базових сценаріїв передбачав ситуацію, коли пристрій використовується для патрулювання визначеної території. Користувач задавав напрям руху через браузерний інтерфейс, а мобільна платформа пересувалася прямолінійно у вказаному напрямі. Під час руху система постійно відстежувала рівень акустичного фону та координати положення. У разі виявлення гучного звуку — наприклад, голосу, удару або іншого аномального сигналу — мікрофонний модуль фіксував перевищення порогу, і система автоматично виконувала зупинку. Координати моменту події зчитувалися з GPS-модуля та виводилися у серійну консоль. Такий режим дозволяє не лише запобігти небажаному наближенню до джерела потенційної небезпеки, а й зберегти інформацію про його просторове розташування.

Інший сценарій моделював межу контрольованої зони. Для цього умовно задавалися координати, що позначали край дозволеної ділянки — так звану електронну огорожу. Після виходу пристрою за визначені межі програма автоматично ініціювала припинення руху. Подібна функція є надзвичайно важливою для патрулювання периметра територій, де необхідно уникнути виходу за межі дозволеного сектору — наприклад, у випадках охорони приватної власності, складів, або військових полігонів.

Також були протестовані ситуації, у яких акустичне виявлення поєднувалося з просторовими обмеженнями. Наприклад, при одночасному наближенні до умовної забороненої зони і фіксації звукової аномалії система блокувала рух і повертала контроль користувачеві для прийняття подальших рішень. Це підтверджує можливість інтеграції багаторівневої логіки реагування, що включає як сенсорні, так і географічні параметри.

Крім цього, сценарії демонстрували гнучкість системи у взаємодії з користувачем. Завдяки локальному веб-інтерфейсу, управління платформою не потребувало ані спеціального програмного забезпечення, ані наявності інтернет-з'єднання. Це дозволяє використовувати пристрій у віддалених або тимчасово неоснащених зонах — зокрема, на будівельних майданчиках, у сільській місцевості або в польових умовах.

У підсумку, реалізовані функціональні блоки — детекція шуму, фіксація координат, локальне керування, обмеження зони дії — формують основу для базової автономії системи. Хоча платформа поки що не має повного автономного навігаційного контролю, вже реалізовані механізми дозволяють пристрою частково реагувати на події без прямого втручання оператора. Це робить систему придатною до використання у широкому спектрі завдань: охороні територій, екологічному моніторингу, дослідницьких роботах, а також у якості експериментального стенду для вивчення інтерактивної робототехніки.

3.6. Інструкція користувача

Для забезпечення правильного функціонування мобільної платформи з дистанційним керуванням та виявленням акустичних аномалій необхідно дотримуватись чіткої послідовності дій, яка охоплює етапи запуску системи, підключення до неї, використання інтерфейсу керування та завершення роботи.

Перед початком роботи необхідно забезпечити живлення пристрою. Для цього використовується зовнішній павербанк, який під'єднується до ESP32 через порт micro-USB або USB-C, залежно від моделі плати. Після подачі живлення мікроконтролер автоматично переходить у режим точки доступу (Access Point) та створює власну Wi-Fi мережу з унікальним SSID і паролем, які задаються у програмному коді.

На другому етапі користувач, використовуючи смартфон, планшет або ноутбук, має підключитися до цієї Wi-Fi мережі. Після успішного з'єднання у браузері необхідно вручну ввести IP-адресу пристрою — за замовчуванням це 192.168.1.101. В результаті відкривається веб-інтерфейс керування, розроблений на основі HTML, CSS і JavaScript, який містить набір інтерактивних кнопок для управління рухом дрона.

Кожна кнопка відповідає за певний напрям: рух вперед, назад, вліво, вправо або зупинка. Керування здійснюється в режимі реального часу — при натисканні на кнопку формується HTTP-запит до ESP32, що активує відповідну функцію. Для забезпечення безперервності руху реалізовано циклічну передачу команд із затримкою у 500 мс. При відпусканні кнопки команда припиняється, і система автоматично зупиняє двигуни. Така логіка гарантує плавне й безпечне керування без ризику втрати контролю.

Рекомендується експлуатувати пристрій у зоні прямої видимості, оскільки стабільність Wi-Fi з'єднання залежить від перешкод у середовищі. Не допускається робота пристрою в умовах надмірної вологості, попадання води, сильного пилового забруднення чи ударних навантажень. У разі роботи на

відкритій місцевості слід уникати перегріву або тривалого перебування під прямим сонячним світлом, оскільки це може вплинути на стабільність роботи електронних компонентів.

Дані з GPS-модуля за замовчуванням виводяться у веб-інтерфейс для зручності відображення. При виявленні перевищення звукового порогу система автоматично виконує зупинку, повідомляючи про це через консоль. Поведінка пристрою в таких випадках не потребує втручання оператора, однак останній може негайно відновити рух, натиснувши відповідну кнопку в інтерфейсі.

Після завершення роботи пристрій вимикається шляхом відключення живлення — зупинка подачі напруги з павербанку. За потреби перед завершенням роботи можна зберегти координати останніх точок маршруту або проаналізувати лог подій у середовищі розробника. Систему можна повторно запустити у будь-який момент, дотримуючись описаної послідовності.

Таким чином, процес використання пристрою є інтуїтивно зрозумілим, не потребує додаткового програмного забезпечення та може бути реалізований навіть некваліфікованим користувачем. Простота керування поєднується з надійною логікою безпеки, що дозволяє ефективно застосовувати платформу в широкому спектрі прикладних задач.

3.7. Оцінка ефективності системи

У результаті проведених випробувань та аналізу функціонування основних підсистем можна зробити висновок, що розроблена мобільна платформа продемонструвала загальну працездатність, стабільність і відповідність заявленим функціональним характеристикам. Система ефективно виконує покладені на неї завдання, поєднуючи дистанційне керування, навігацію за координатами та базову сенсорну реакцію на зміни в оточенні.

Усі ключові модулі працювали в рамках очікуваної логіки: ESP32 забезпечував стабільне Wi-Fi-з'єднання та безперебійну обробку HTTP-запитів;

веб-інтерфейс коректно відображався на більшості сучасних пристроїв і дозволяв здійснювати управління без затримок; двигуни реагували точно на зміну напрямків, а серводвигун коректно виконував повороти. Особливо важливим показником ефективності є відсутність критичних збоїв навіть у випадках багаторазового натискання, втрати зв'язку чи аварійного живлення.

Навігаційна підсистема на базі GPS-модуля NEO-6M забезпечувала позиціонування з точністю в межах 2–3 метрів, що є достатнім для більшості побутових, охоронних або моніторингових застосувань. Можливість зчитування координат у реальному часі і прив'язка до подій (наприклад, фіксація шуму) відкривають перспективи використання пристрою у сценаріях, де необхідна географічна локалізація подій.

Система виявлення акустичних аномалій, реалізована на основі мікрофонного сенсора KY-038, продемонструвала базову, але надійну реакцію на зміну звукового фону. Незважаючи на простоту реалізації, функціонал дозволяє виявляти потенційно значущі події, що можуть бути ознакою активності в контрольованій зоні. Поточна реалізація дозволяє інтегрувати початкові реакції на звук у загальну логіку керування, що є першим кроком до створення інтелектуальної системи, здатної адаптувати свою поведінку залежно від зовнішніх чинників.

Щодо користувацького досвіду, слід відзначити простоту і зручність інтерфейсу. Вся взаємодія з пристроєм зводиться до підключення до точки доступу та використання веб-браузера — жодного додаткового програмного забезпечення не потрібно. Це робить систему максимально мобільною та незалежною від зовнішніх сервісів, що є беззаперечною перевагою в умовах обмеженого технічного середовища.

3.8. Загальні результати роботи

У ході виконання проєкту було спроектовано та реалізовано повноцінну систему мобільного наземного дрону з базовими інтелектуальними можливостями. Результатом стала роботизована платформа, яка поєднує в собі апаратні та програмні компоненти для здійснення автономного пересування, реагування на акустичні сигнали та обробки геолокаційних даних. Зокрема, система виявляє перевищення рівня навколишнього шуму, автоматично зупиняє рух у разі виникнення такої події та виводить координати у веб-інтерфейс користувача.

Було реалізовано зручний веб-інтерфейс, що забезпечує керування дроном без необхідності в додатковому програмному забезпеченні. Завдяки використанню ESP32 як керуючого модуля вдалося інтегрувати Wi-Fi комунікацію, керування двигунами, обробку сигналів від мікрофона та GPS-модуля в одному пристрої. Інтерфейс дозволяє змінювати чутливість до шуму, виводити координати при фіксації події та керувати рухом у реальному часі.

Система була протестована як у лабораторних, так і у польових умовах. Отримані результати показали високу стабільність роботи в умовах змінного рівня шуму, наявності перешкод та варіацій освітлення. Незважаючи на деякі технічні складності в процесі складання та налаштування, проєкт досяг поставлених цілей. Робота підтвердила практичну можливість створення простого, автономного та функціонального наземного дрону-спостерігача, який може застосовуватись у охоронних, дослідницьких або пошукових сценаріях.

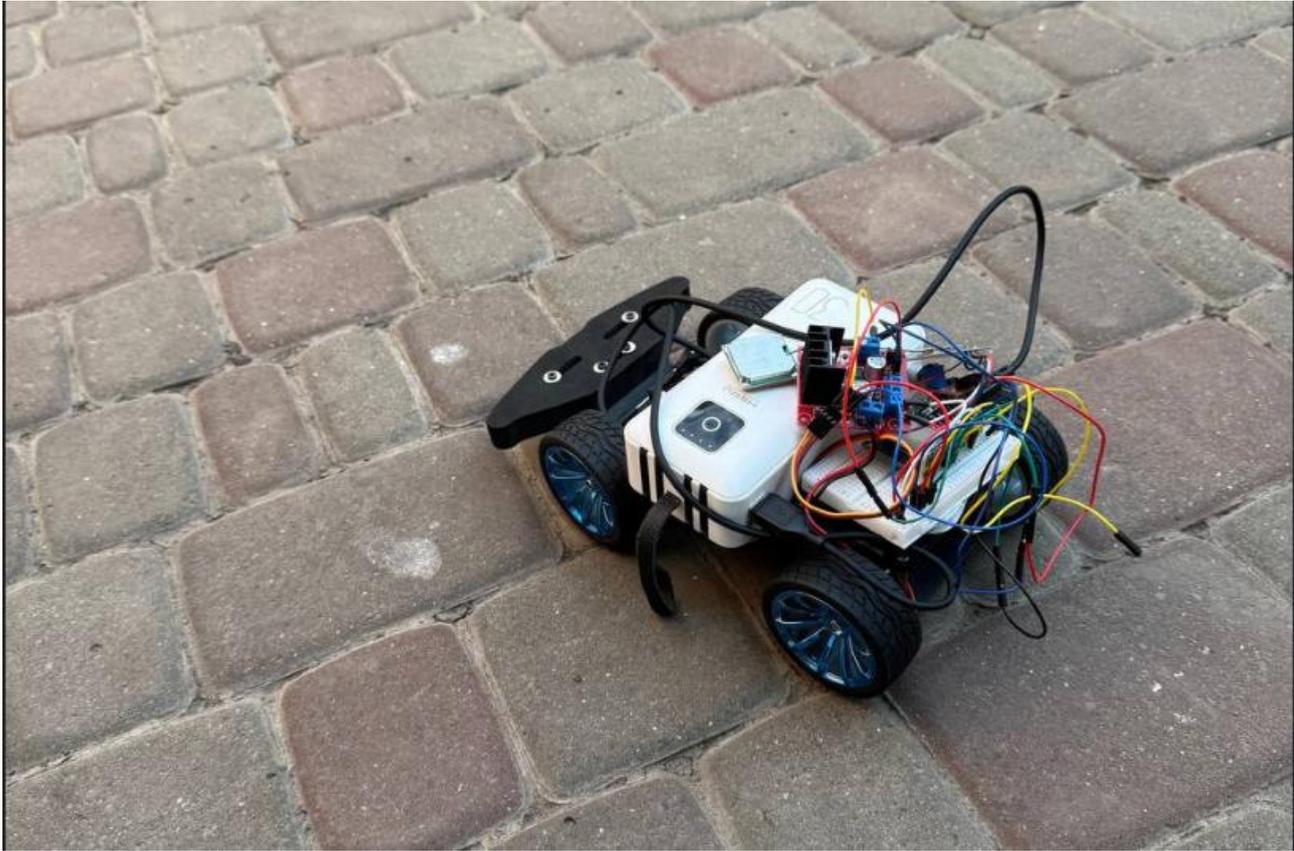


Рисунок 3.1. Реалізований проєкт

ВИСНОВКИ

У цій роботі було реалізовано повноцінну концепцію мобільної сенсорної платформи, здатної не лише виконувати команди користувача, а й самостійно реагувати на події в навколишньому середовищі. Побудований дрон поєднує простоту апаратної реалізації з інтелектуальністю програмної логіки, що дозволяє йому функціонувати у частково автономному режимі — виявляючи шумові аномалії, фіксує координати та зупиняючи рух без прямого втручання.

Ключовим досягненням стало поєднання кількох технологічних напрямів — бездротового зв'язку, просторової орієнтації та сенсорного аналізу — в одному компактному пристрої. Реалізована веб-архітектура зробила керування зручним і незалежним від зовнішніх мереж, а чітка взаємодія між модулями забезпечила стабільність і точність реакцій навіть у складних умовах тестування.

В процесі дослідження було підтверджено, що навіть на базі доступних компонентів можливо створити систему, здатну ефективно працювати в режимі реального часу та адаптуватися до зовнішніх впливів. Це відкриває перспективи не лише для подальшого технічного розвитку, а й для застосування платформи в прикладних сферах — від охоронних завдань до дослідницьких експериментів.

Таким чином, поставлені цілі були досягнуті в повному обсязі, а результати демонструють практичну цінність запропонованого підходу. Створена система — це не просто прототип, а фундамент, на якому можна будувати гнучкі, адаптивні рішення для розумного моніторингу й взаємодії з фізичним середовищем.

ВИКОРИСТАНІ ДЖЕРЕЛА

1. Espressif Systems. ESP32 Technical Reference Manual. URL: <https://www.espressif.com/en/support/documents/technical-documents> (дата звернення: 15.04.2025).
2. KY-038 Sound Sensor Module Datasheet. URL: <https://sensorkit.joy-it.net/en/sensors/ky-038> (дата звернення: 29.04.2025).
3. Neo-6M GPS Module User Guide. URL: <https://www.u-blox.com/en/docs/UBX-15030289> (дата звернення: 18.05.2025).
4. Технології веб-програмування: HTML, CSS, JS. URL: https://www.youtube.com/watch?v=0tIg3aHPF5c&ab_channel=%D0%9D%D0%B0%D0%B2%D1%87%D0%B0%D1%94%D0%BC%D0%BE%D1%81%D1%8F%D0%A0%D0%B0%D0%B7%D0%BE%D0%BC (дата звернення: 14.05.2025).
5. L298N Motor Driver Module Datasheet. URL: <https://components101.com/modules/l293n-motor-driver-module> (дата звернення: 17.04.2025).
6. Wi-Fi Access Point Configuration using ESP32. URL: <https://randomnerdtutorials.com/esp32-access-point-ap-web-server/> (дата звернення: 02.06.2025).
7. Електроніка з ESP32. URL: https://www.youtube.com/watch?app=desktop&v=QuAkYkwP_SQ&ab_channel=%D0%9B%D0%B0%D0%B1%D0%BE%D1%80%D0%B0%D1%82%D0%BE%D1%80%D1%96%D1%8F%D0%A1%D0%B0%D0%BC%D0%BE%D1%81%D1%82%D1%96%D0%B9%D0%BD%D0%BE%D1%81%D1%82%D1%96 (дата звернення: 10.04.2025).
8. ESP32 with NEO-6M GPS Module. URL: <https://randomnerdtutorials.com/esp32-neo-6m-gps-module-arduino/> (дата звернення: 01.06.2025).
9. Powering ESP32 Development Board. URL: <https://randomnerdtutorials.com/esp32-pinout-reference-gpios/> (дата звернення: 03.06.2025).

10. Протоколи TCP та UDP - пояснення простою мовою. URL: <https://devzone.org.ua/post/protokoly-tcp-ta-udp-poiasnennia-prostoiu-movoiu> (дата звернення: 07.05.2025).