

Харківський національний університет радіоелектроніки

Факультет Факультет комп'ютерної інженерії та управління

Кафедра електронних обчислювальних машин

Рівень вищої освіти перший (бакалаврський)

Спеціальність 123 «Комп'ютерна інженерія»
(код і повна назва)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма Комп'ютерна інженерія
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

“ _____ ” _____ 20__ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Щербак Аліні Сергіївні
(прізвище, ім'я, по батькові)

1. Тема роботи Програмно-апаратний комплекс для дистанційного керування мобільним роботом

затверджена наказом по університету від “ 26 ” травня 2025 р. № 424 Ст

2. Термін подання здобувачем роботи до екзаменаційної комісії 16 червня 2025 р.

3. Вхідні дані до роботи _____

1) аналіз сучасних мобільних роботів та методів керування;

2) вибір апаратних компонентів для реалізації проєкту;

3) розробка схеми підключення та живлення;

4) створення та тестування програмного забезпечення керування.

4. Перелік питань, що потрібно опрацювати у роботі _____

1) дослідження існуючих технічних рішень;

2) проєктування апаратної частини системи;

3) реалізація програмного забезпечення;

4) аналіз працездатності та ефективності розробленої моделі.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій _____

Слайд-презентація – 9 слайдів _____

6. Консультанти розділів роботи (заповнюється за наявності консультантів згідно з наказом, зазначеним у п.1)

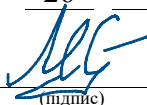
Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Строк / терміни виконання етапів роботи	Примітка
1	Аналіз проблеми та огляд існуючих рішень	27.05.25-30.05.25	
2	Вибір технології розробки та інструментальних засобів	31.05.25-02.06.25	
3	Розробка алгоритмічного забезпечення	03.06.25-05.06.25	
4	Розробка та відлагодження програмного забезпечення	06.06.25-09.06.25	
5	Оформлення матеріалів кваліфікаційної роботи	10.06.25-11.06.25	
6	Подання кваліфікаційної роботи керівникові та її попередній захист	12.06.25-13.06.25	
7	Подання кваліфікаційної роботи на рецензування	14.06.25-16.06.25	

Дата видачі завдання “ 26 ” травня 2025 р.

Здобувач _____


(підпис)

Керівник роботи _____

(підпис)

ст. викл. Ольга ЄРОШЕНКО _____

(посада, власне ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 68 с., 14 рис., 2 табл., 1 дод., 16 джерел.

ARDUINO, МІКРОКОНТРОЛЕР, МОБІЛЬНИЙ РОБОТ, BLUETOOTH, ДИСТАНЦІЙНЕ КЕРУВАННЯ, ДРАЙВЕР ДВИГУНА, АПАРАТНЕ ЗАБЕЗПЕЧЕННЯ, ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ

Метою даної кваліфікаційної роботи є розробка мобільної робототехнічної платформи з використанням мікроконтролерної плати Arduino Uno та реалізацією дистанційного керування через інтерфейс Bluetooth.

У ході виконання кваліфікаційної роботи було виконано проектування апаратної частини, включаючи схему підключення драйверів двигунів і Bluetooth-модуля, а також реалізовано алгоритми керування рухом, які дозволяють здійснювати маневрування у просторі відповідно до команд користувача. Проведено комплексне тестування системи з метою виявлення та усунення можливих недоліків, що забезпечило стабільність роботи платформи у різних умовах експлуатації. Результати роботи можуть бути використані для подальшого розвитку автономних мобільних роботів у сферах автоматизації, навчання та досліджень. Праця містить детальний опис технічних рішень, методів розробки, а також аналіз продуктивності створеної системи.

ABSTRACT

Bachelor's thesis: 68 pages, 14 figures, 2 tables, 1 appendices, 16 sources.

ARDUINO, MICROCONTROLLER, MOBILE ROBOT, BLUETOOTH,
REMOTE CONTROL, MOTOR DRIVER, HARDWARE, SOFTWARE

The major goal of this thesis is to develop a mobile robotic platform using the Arduino Uno microcontroller board and implementing remote control via the Bluetooth interface.

In order to complete the qualification work, the hardware was designed, including the connection diagram for the motor drivers and the Bluetooth module, and motion control algorithms were implemented to allow manoeuvring in space in accordance with user commands. Comprehensive testing of the system was carried out to identify and eliminate possible deficiencies, which ensured the stability of the platform in various operating conditions. The results of the work can be used for the further development of autonomous mobile robots in the areas of automation, education and research. The paper contains a detailed description of technical solutions, development methods, and performance analysis of the created system.

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ	8
ВСТУП	10
1 АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ.....	12
1.1 Огляд сучасних мобільних роботів	12
1.2 Порівняння методів керування	15
1.3 Аналіз вибору мікроконтролерної плати для реалізації проєкту	18
1.3.1 Огляд сучасних мікроконтролерних платформ для мобільної робототехніки	18
1.3.2 Архітектура та технічні характеристики Arduino.....	20
1.3.3 Інтерфейси та можливості підключення периферії.....	21
1.3.4 Екосистема розробки та програмування.....	24
1.3.5 Обмеження та особливості Arduino для мобільної робототехніки	25
1.3.6 Обґрунтування остаточного вибору.....	26
1.4 Аналіз драйверів керування двигунами для мобільних платформ	27
1.5 Висновки до розділу	30
2 РОЗРОБКА КОНСТРУКЦІЇ ТА АПАРАТНОЇ ЧАСТИНИ.....	31
2.1 Вибір платформи та компонентів.....	31
2.2. Схема підключення елементів	39
2.3. Розробка системи живлення комплексу	41
2.4.1 Вибір матеріалу основи та монтаж компонентів	42
2.4.2 Переваги обраного конструктивного рішення.....	43
2.5. Висновки до розділу	44
3 ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ КЕРУВАННЯ.....	46
3.1 Алгоритм ручного керування через Bluetooth	46
3.2 Реалізація програмного коду.....	51
3.3 Тестування системи	55

3.3.1	Перевірка апаратної частини	55
3.3.2	Функціональне тестування програмного забезпечення	56
3.3.3	Висновки за результатами тестування.....	57
3.4	Висновки до розділу	57
	ВИСНОВКИ.....	60
	ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	62
	ДОДАТОК А Графічний матеріал кваліфікаційної роботи.....	64

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

- АЦП – аналого-цифровий перетворювач
- ШИМ – широтно-імпульсна модуляція
- AREF – опорна напруга для АЦП (англ., Analog Reference)
- COM – послідовний порт зв'язку (англ., Communication port)
- EEPROM – електрично стирається програмована пам'ять тільки для читання (англ., Electrically Erasable Programmable Read-Only Memory)
- GND – загальний нульовий провід (англ., Ground)
- GPIO – загального призначення вхід/вихід (англ., General Purpose Input/Output)
- GPS – система глобального позиціонування (англ., Global Positioning System)
- I2C – інтерфейс міжінтегральної взаємодії (англ., Inter-Integrated Circuit)
- IDE – інтегроване середовище розробки (англ., Integrated Development Environment)
- IN – вхідний сигнал (англ., Input)
- IoT – інтернет речей (англ., Internet of Things)
- MISO – вхід даних від підключеного пристрою (англ., Master In Slave Out)
- MOSI – вихід даних до підключеного пристрою (англ., Master Out Slave In)
- PWM – широтно-імпульсна модуляція (англ., Pulse Width Modulation)
- ROS – операційна система роботів (англ., Robot Operating System)
- RTX – прийом і передача даних (англ., Receive and Transmit)
- RXD – прийом даних (англ., Receive Data)
- SCK – тактовий сигнал SPI (англ., Serial Clock)
- SCL – лінія тактування інтерфейсу I²C (англ., Serial Clock Line)

SDA – лінія даних інтерфейсу I²C (англ., Serial Data Line)

SPI – послідовний периферійний інтерфейс (англ., Serial Peripheral Interface)

TXD – передача даних (англ., Transmit Data)

TWI – інтерфейс двопровідної передачі (англ., Two Wire Interface)

UART – універсальний асинхронний приймач-передавач (англ., Universal Asynchronous Receiver-Transmitter)

UGV – безпілотний наземний транспортний засіб означає (англ., unmanned ground vehicle)

USB – універсальна послідовна шина (англ., Universal Serial Bus)

VCC – напруга живлення (англ., Voltage at the Common Collector)

ВСТУП

Сьогоднішній світ важко уявити без технічного прогресу, який щодня вдосконалюється та впливає на різні сфери життя людини, від побуту до складних промислових систем. Одним із напрямків, який активно розвивається останніми роками є робототехніка та створення автономних, або напівавтономних систем. Ці системи все частіше використовуються не лише в промисловості, але й у навчальних закладах, наукових лабораторіях і навіть у повсякденному житті. Особливої популярності набули мобільні роботизовані платформи, які здатні виконувати прості або складні задачі, залежно від конструкції та встановленого програмного забезпечення. Такі платформи є чудовою основою для навчання, оскільки дозволяють студентам практично ознайомитися з основами електроніки, програмування, управління моторами, сенсорними модулями та бездротовими засобами зв'язку.

Однією з актуальних задач, яку можна вирішити на базовому рівні з допомогою доступних технологій, є створення дистанційно керованого мобільного пристрою. Такі пристрої можуть керуватися через різні протоколи такі як радіоканал, Wi-Fi або Bluetooth. З огляду на простоту реалізації, надійність і доступність, Bluetooth є одним з найзручніших варіантів зв'язку на короткі відстані.

Метою даної кваліфікаційної роботи було спроектувати, зібрати та запрограмувати мобільну модель, якою можна керувати через Bluetooth-з'єднання за допомогою смартфона. Реалізація цього проекту дозволяє не тільки закріпити знання з мікроконтролерної техніки та програмування, але й вивчити на практиці основи автоматизованого управління, апаратного конструювання та розробки власного програмного забезпечення. В якості апаратної основи для реалізації було обрано мікроконтролер Arduino Uno, який є одним із найпопулярніших рішень у світі саморобної та навчальної електроніки. Цей контролер підтримує велику кількість різноманітних

модулів, серед яких і модуль HC-06, що забезпечує бездротове з'єднання за технологією Bluetooth. Для реалізації руху моделі було використано драйвер двигуна L298N та чотири колісні електродвигуни.

Для керування платформою було створено мобільний додаток, який надає користувачеві можливість задавати напрямок руху моделі за допомогою зручного графічного інтерфейсу. Інтерфейс є інтуїтивно зрозумілим, що дозволяє швидко зорієнтуватися у керуванні навіть без попередньої підготовки. З'єднання між мобільним пристроєм та моделлю здійснюється через бездротову технологію Bluetooth, яка забезпечує стабільний зв'язок на близькій відстані. У програмній частині моделі реалізована базова логіка керування, відповідно до якої сигнали, які надходять з мобільного додатку, приймаються мікроконтролером і інтерпретуються як команди на виконання певних дій. Наприклад, при натисканні відповідної кнопки на екрані телефону модель починає рухатися вперед, назад, здійснює поворот вліво або вправо. Крім того, присутня команда зупинки, яка миттєво блокує рух. Безпеці керування була приділена особлива увага, у випадку, якщо зв'язок між мобільним пристроєм і моделлю зникне зв'язок, або протягом певного часу не надійдуть жодні команди, модель автоматично перейде у стан зупинки. Такий підхід дозволяє зменшити ризики, пов'язані з можливими помилками в управлінні або втручанням сторонніх факторів.

Дана кваліфікаційна робота охоплює повний процес створення мобільної керованої моделі: від вибору елементної бази та побудови схеми, до розробки керуючого коду та створення програмного забезпечення для управління. Створений пристрій може бути використаний як навчальний інструмент або стати основою для подальших досліджень та покращення. В тому числі, можна розширити функціонал, додавши датчики для виявлення перешкод, реалізувавши автономний режим або використавши альтернативні способи зв'язку, такі як Wi-Fi чи радіоканал.

1 АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ

1.1 Огляд сучасних мобільних роботів

Мобільні роботи з колісною основою використовують не тільки в науково-дослідницькій діяльності, але й для забезпечення логістики транспортування товарів, рятувальної практики, у військовій сфері для ризикованого завдання, розвідки або знешкодження вибухонебезпечних предметів. Вони мають важливу роль в області сучасної робототехніки завдяки своїй ефективності та універсальності, особливу увагу привертають моделі, що мають можливість ручного керування – за допомогою радіопульта, мобільний застосунок, або через комп'ютерне програмне забезпечення. Це особливо важливо у непередбачуваних ситуаціях, коли робот знаходиться у змінному середовищі, ручне керування дозволяє оператору приймати рішення в реальному часі, корегувати роботу системи та контролювати виконання завдань.

Яскравим прикладом сучасного мобільного робота є Husky UGV канадської компанії Clearpath Robotics. Чотириколісний всюдихід використовується для проведення наукових, навчальних і прикладних досліджень. Платформа має міцну сталеву раму, корпус захищений від пилу та вологи, що надає змогу використовувати робота в складних умовах – у пилових середовищах, так і в холодному кліматі. Наприклад, Husky UGV використовується для спостереження за змінами чисельності імператорських пінгвінів на території Антарктиди, де важливою умовою є безшумність руху та надійна робота при низьких температурах.



Рисунок 1.1 – Husky UGV A300

Платформа розрахована на широкий спектр корисних навантажень: GPS системи, стереокамери, використання лідарів, ультразвукові сенсори, а також комп'ютерними платформами, такими як NVIDIA Jetson, Intel NUC або Raspberry Pi. Робот може рухатися як автономно, так і за командами оператора. Ручний режим включає роботу з геймпадом або керування з віддаленого комп'ютера, що дозволяє оператору повністю керувати рухом робота. Автономний режим базується на використанні Robot Operating System. За допомогою цієї системи можна реалізувати складні алгоритми навігації, обробку сенсорних даних, картографування, тощо.

Ще одним прикладом мобільної роботизованої платформи є робот розроблений канадської компанії Clearpath Robotics разом з Open Robotics – TurtleBot 4. Платформа заснована на операційній системі Robot Operating System, що є стандартом у сфері академічної та прикладної робототехніки.



Рисунок 1.2 – TurtleBot 4

TurtleBot 4 обладнаний диференціальним приводом, який дає змогу рухатися вперед, назад, а також обертатися на місці. Колісна основа забезпечує стійкість і маневреність на різних поверхнях, що дає змогу використовувати його в закритих приміщеннях, лабораторіях або навіть офісах. Робот має розширений режим ручного керування: оператор може надавати команди траєкторії руху, швидкість, або напрямок через ноутбук за допомогою ROS-команд, або спеціальних інтерфейсів, використати джойстик, мобільний застосунок, що взаємодіє з ROS через бездротовий зв'язок. Платформа обладнана стереокамерою, LIDAR-сенсором, ультразвуковими датчиками, та іншими пристроями для автономного режиму, та може задіяти такі алгоритми як Simultaneous Localization and Mapping, планування маршруту, уникнення перешкод та машинне бачення.

Незважаючи на те, що TurtleBot 4 спрямований насамперед на сферу освіти, він має серйозний функціонал і часто використовується як прототипна основа для наукових досліджень. Багато університетів і наукових центрів використовують його для відпрацювання і тестування сучасних методів у галузі робототехніки, штучного інтелекту та автономних систем.

Таким чином, сучасні мобільні роботи з ручним керуванням показують досить широкий спектр використання: від наукових досліджень і навчання до виконання спеціалізованих завдань у реальних умовах.

1.2 Порівняння методів керування

При розробці роботизованих систем важливим етапом є реалізація ефективного, енергозберігаючого стабільного способу керування. Від способу зв'язку між користувацьким інтерфейсом та апаратним забезпеченням залежить зручність в роботі, надійність системи, швидкість реагування на команди, масштабованість, а також рівень технічної складності реалізації. Відповідно до поставлених завдань навчального характеру, для демонстрації, промислового рішення або дослідницьких експериментів можуть бути обрані різні канали зв'язку, які мають свої особливості.

Найпопулярнішими методами дистанційного керування мобільними платформами є використання радіоканалу, Bluetooth і Wi-Fi. Кожна з технологій має свої переваги та недоліки, а також сферу доцільного застосування.

Використання радіоканалу є класичним і широко застосовуваним методом бездротового керування пристроями. Технології передачі даних у діапазонах 27МГц, 433 МГц, 868 МГц а також 2.4ГГц успішно використовуються в побутових радіокерованих моделях, автомобілях, дронах та іншій техніці. Перевагою радіоканалу є його мінімальна затримка сигналу, оскільки використовуються низькорівневі протоколи обміну. Це дає можливість оператору отримувати моментальний відгук на дії, що особливо важливо для динамічних систем, де навіть незначна затримка може призвести до втрати управління. Більше того, радіоканали можуть забезпечувати високу надійність в умовах з мінімальними перешкодами, зокрема на відкритих просторах або в приміщеннях зі стабільним електромагнітним фоном. Недоліком використання радіоканалу є обмеженість функцій керуючого пристрою та неможливість масштабованості без додаткової електроніки. Також у більшості випадків керуючий пристрій лише надає сигнали, та не приймає відповідь, що ускладнює контроль та функціональність.

Іншою поширеною технологією бездротового зв'язку є Bluetooth. Він був спеціально розроблений для забезпечення надійного швидкого з'єднання на невеликих відстанях до 10 метрів, враховуючи потреби низького енергоспоживання, доступності та простоти конфігурації.

У даному дипломному проекті використовується технологія Bluetooth для забезпечення зв'язку між користувачем та платформою, застосовано модуль HC-06 – популярне рішення для інтеграції Bluetooth-з'єднання в мікроконтролерні системи. Для обміну даними між керуючим пристроєм та платформою модуль використовує протокол UART. У проекті команди надходять із застосунку у вигляді цифр, які перетворюються на конкретні рухові дії в Arduino: рух уперед або назад, повороти, зупинка тощо.

Bluetooth працює в універсальному ISM діапазоні 2.4ГГц, що забезпечує сумісність з більшістю сучасних пристроїв. Хоча й використання Bluetooth передбачає застосування певного стеку протоколів, реалізація з допомогою модуля HC-06 спрощує цей процес, оскільки він працює як звичайний послідовний порт, запобігаючи використанню додаткових бібліотек та драйверного обслуговування.

У рамках дипломного проекту важливою деталлю є інтерактивна взаємодія між користувачем та платформою у реальному часі, що забезпечує Bluetooth-з'єднання. Завдяки цьому оператор може візуально оцінити роботу системи, її відповіді на команди, та корегувати за необхідності. Такий режим роботи допомагає не тільки ефективно тестувати платформу, але й засвоїти принципи побудови систем керування, обробки сигналів.

Отже, можна визначити, що використання Bluetooth в розробці мобільної платформи на базі Arduino є обґрунтованим вибором, зважаючи на простоту реалізації, енергоефективність, й масштабованість у разі розширення функціоналу мобільного додатку.

Технологія Wi-Fi набуває все більше популярності в області робототехніки завдяки високій гнучкості, масштабованості, пропускній здатності, яку вона забезпечує. Wi-Fi це стандарт бездротових локальних

мереж на стандартах IEEE 802.11. Він дозволяє реалізувати не тільки віддалене керування, а й інтерактивні та автономні функції, що забезпечує багатоканальний обмін інформацією між роботом і оператором.

Wi-Fi істотно збільшує функціональні межі пристроїв завдяки великій швидкості передавання даних, від кількох Мбіт/с до сотень Мбіт/с залежно від стандарту. Наприклад, застосування бездротових технологій зв'язку надає можливості виводити телеметричну інформацію, передавати відеопотік або одночасно керувати кількома підсистемами. Цінним що сучасні протоколи бездротового передавання даних разом з мережевими протоколами дозволяють інтегрувати роботизовані системи в наявні комп'ютерні мережі та інфраструктуру інтернету. Це дозволяє віддалено здійснювати моніторинг і управління роботами майже з будь-якої точки світу за наявності необхідного мережевого з'єднання, не дивлячись на обмеження фізичної відстані, що є необхідною умовою для роботи. Для реалізації такого з'єднання треба налаштувати зовнішній сервер, або створити веб-інтерфейсу управління, де мікроконтролер з Wi-Fi може працювати як веб-сервер та надавати інтерфейс для управління через браузер.

Wi-Fi підтримує клієнтський режим Station mode та режим точки доступу Access Point mode, що дозволяє реалізувати пряме підключення, коли робот створює власну точку доступу, до якої підключаються керуючий пристрій, або підключитися через маршрутизатор, де робот інтегрується в існуючу локальну мережу як клієнт. Останній варіант зручніший для реалізації віддаленого керування через Інтернет, або локальну мережу з декількома пристроями.

В той же час, використання Wi-Fi має деякі технічні виклики. Зовнішні перешкоди, завантаженість частотного діапазону, особливо 2.4ГГц, апаратні характеристики маршрутизатору, усе це впливає на якість зв'язку. Також, порівняно з використанням Bluetooth або радіоканалу, Wi-Fi потребує більше енергоспоживання, оскільки працює на вищій потужності для забезпечення великого радіусу дії, надає високі швидкості, та використовує більш складні

протоколи зв'язку, Мікросхеми Wi-Fi зазвичай мають ускладнену архітектуру і більше компонентів. Наприклад, мікроконтролер ESP8266 має середнє енергоспоживання при передачі 160–250 мА, на відміну від Bluetooth модуля HC-06 який має середнє енергоспоживання 30–50 мА при активному з'єднанні.

Підсумовуючи, можна сказати, що Wi-Fi є досить ефективним способом бездротового управління, який забезпечує масштабованість, гнучкість і широкі можливості зворотного зв'язку. Однак його застосування вимагає уважного аналізу умов використання, наявності живлення та програмних ресурсів, особливо у випадках створення автономних мобільних пристроїв, обмежених в енергетичних і апаратних ресурсах.

1.3 Аналіз вибору мікроконтролерної плати для реалізації проєкту

1.3.1 Огляд сучасних мікроконтролерних платформ для мобільної робототехніки

Ринок мікроконтролерних рішень охоплює три основні напрямки: одноплатні комп'ютери типу Raspberry Pi, спеціалізовані мікроконтролери на кшталт STM32 і ESP32, а також платформи швидкого прототипування серед яких найбільшу популярність здобув Arduino. Кожен напрямок характеризується специфічними технічними особливостями та цільовими застосуваннями.

Одноплатні комп'ютери Raspberry Pi функціонують під управлінням повноцінних дистрибутивів Linux, забезпечуючи значну обчислювальну потужність для реалізації алгоритмів комп'ютерного зору та роботи з мережевими протоколами. Проте застосування таких рішень для базових завдань керування мобільними платформами призводить до невиправданого збільшення енергоспоживання та ускладнення архітектури системи.

Мікроконтролери ESP32 та STM32 забезпечують високу продуктивність при обмеженому енергоспоживанні. ESP32 інтегрує модулі бездротового зв'язку Wi-Fi та Bluetooth, що визначає його застосування в IoT-системах. Лінійка STM32 охоплює широкий діапазон продуктивності та периферійних можливостей. Водночас розробка на цих платформах потребує глибокого розуміння архітектури мікроконтролерів та використання спеціалізованих середовищ програмування.

Платформа Arduino зайняла провідні позиції в сфері освітніх та прототипних проектів завдяки мінімальному порогу входу, розвиненій екосистемі бібліотек, активній спільноті розробників та економічній доступності. Комбінація цих характеристик визначає Arduino як оптимальне рішення для дипломного проектування та створення прототипів мобільних роботів.

Для реалізації проекту було розглянуто кілька популярних мікроконтролерних платформ, серед яких Arduino Uno R3[1], Arduino Nano[2], Arduino Mega 2560[3]. Основні характеристики порівняно в таблиці 1.1. Головними критеріями вибору були: доступність плати, простота програмування і підключення, об'єм пам'яті, кількість доступних інтерфейсів для зв'язку, споживання енергії, габарити, а також наявність технічної документації та підтримки спільноти.

Після детального аналізу було прийнято рішення задіяти в проекті Arduino Uno R3 з мікроконтролером ATmega328P[4]. Основними перевагами цієї плати є її простота, стабільність у роботі, поширеність, а також велика кількість прикладів, бібліотек і документації.

Таблиця 1.1 – Порівняння платформ

Платформа	Мікроконтролер	Пам'ять Flash	RAM	UART / I2C / SPI	Переваги	Недоліки
Arduino Uno R3	ATmega328	32КБ	2КБ	I2C/TWI SPI PWM	Простота, документація, підтримка	Обмежена пам'ять, 1 UART
Arduino Nano	ATmega328	32КБ	2КБ	UART, I2C, SPI	Компактна, ті ж можливості, що Uno	Менше роз'ємів для підключень
Arduino Mega 2560	ATmega2560	256КБ	8КБ	4×UART, I2C, SPI	Багато портів, більше пам'яті	Габарити, вища ціна

1.3.2 Архітектура та технічні характеристики Arduino

Конструкція Arduino Uno R3 базується на 8-розрядному мікроконтролері ATmega328P компанії Microchip, попередньо Atmel. Даний кристал входить до серії AVR і реалізує Гарвардську архітектуру із фізично розділеними сегментами пам'яті для програмного коду та операційних даних.

Технічна специфікація ATmega328P включає наступні параметри:

- обчислювальне ядро: 8-розрядний RISC-процесор із набором з 131 команди;
- робоча частота: максимально 20 МГц, у конфігурації Arduino Uno встановлено 16 МГц;
- програмна пам'ять: 32 КБ Flash-типу, 0.5 КБ зарезервовано під завантажувач;
- оперативна пам'ять: 2 КБ SRAM для роботи зі змінними в процесі виконання;
- енергонезалежна пам'ять: 1 КБ EEPROM для збереження константних параметрів;
- порти введення-виведення: 23 лінії, Arduino Uno експонує 14

цифрових;

- аналогові канали: 8 входів 10-розрядного АЦП (доступні 6 у Arduino Uno);

- лічильники-таймери: 3 модулі, один 16-розрядний, два 8-розрядних.

Тактування системи реалізовано через зовнішній кварцовий генератор 16 МГц, що гарантує стабільність частотних характеристик усіх периферійних вузлів. Альтернативно можливе використання вбудованого RC-осцилятора, проте із зниженою точністю синхронізації.

Після детального аналізу було прийнято рішення задіяти в проекті Arduino Uno R3 з мікроконтролером ATmega328P[4]. Основними перевагами цієї плати є її простота, стабільність у роботі, поширеність, а також велика кількість прикладів, бібліотек і документації.

1.3.3 Інтерфейси та можливості підключення периферії

Arduino Uno R3 базується на мікроконтролері ATmega328P, який працює на тактовій частоті 16 МГц. Плата має 32 Кб Flash-пам'яті, 2 Кб SRAM і 1 Кб EEPROM, це дає можливість програмний код і деякі постійні дані, необхідні під час роботи. Незважаючи на невеликий об'єм пам'яті, її цілком достатньо для виконання типових завдань керування пристроєм, збору та передачі даних, а також нескладної обробки сигналів.

Архітектура плати інтегрує декілька стандартизованих інтерфейсів передачі даних.

Розглянемо цифрові інтерфейси передачі даних. Інтерфейс UART реалізує асинхронну послідовну передачу даних між мікроконтролером та зовнішніми пристроями. Застосовується для встановлення зв'язку з персональним комп'ютером через USB-з'єднання та підключення модулів бездротової передачі даних, зокрема Bluetooth HC-06. Arduino Uno містить один апаратний UART-канал, розташований на контактах 0 – прийом та 1 – передача. Швидкість обміну конфігурується програмно в діапазоні від 300 до

115200 біт/с.

Протокол I2C забезпечує синхронну двопровідну комунікацію для інтеграції сенсорних модулів та допоміжних пристроїв. Реалізація використовує лінії SDA та SCL на контактах A4 та A5 відповідно. Топологія шини дозволяє підключення до 127 периферійних пристроїв, що обумовлює застосування для інтеграції множинних датчиків: акселерометрів, гіроскопів, сенсорів температури та вологості.

Інтерфейс SPI призначений для високошвидкісного синхронного обміну даними з периферійними пристроями, такими як карти пам'яті SD, графічні дисплеї та драйвери приводів. Реалізація задіює чотири сигнальні лінії: MISO, MOSI, SCK та SS на контактах 10-13. Пропускна здатність інтерфейсу перевищує показники I2C.

Односпрямований інтерфейс 1-Wire забезпечує комунікацію з датчиками температури типу DS18B20 через єдиний цифровий контакт. Протокол дозволяє підключати декілька пристроїв паралельно до однієї лінії, кожен з унікальною адресою. Особливістю є можливість живлення датчиків безпосередньо від лінії даних, що спрощує монтаж у мобільних системах.

Розглянемо аналогові та цифрові входи-виходи. Функція PWM доступна на 6 цифрових контактах 3, 5, 6, 9, 10, 11 та призначена для регулювання потужності навантаження через зміну коефіцієнта заповнення імпульсної послідовності. Він застосовується для управління швидкістю електродвигунів, регулювання яскравості світлодіодів та генерації псевдо-аналогових сигналів. Базова частота ШІМ становить приблизно 490 Гц для більшості каналів.

Цифрові порти GPIO представляють основний інтерфейс для підключення різноманітної периферії. Arduino Uno надає 14 цифрових контактів 0-13, кожен з яких може працювати як вхід або вихід. Максимальний струм навантаження складає 20 мА на контакт, загальний струм всіх контактів не повинен перевищувати 200 мА. Для входів доступні три режими: INPUT високий опір, INPUT_PULLUP з внутрішнім

підтягуючим резистором 20-50 кОм та OUTPUT. Внутрішні підтягуючі резистори особливо корисні при підключенні кнопок та перемикачів, оскільки усувають необхідність зовнішніх резисторів.

Аналогові входи реалізовані через 10-розрядний АЦП з 6 каналами на контактах A0-A5. Діапазон вимірювання за замовчуванням становить 0-5В з роздільною здатністю 1024 кроки (близько 4.9 мВ на крок). Можливість зміни опорної напруги через контакт AREF дозволяє підвищити точність вимірювань для сигналів меншої амплітуди. Швидкість перетворення складає приблизно 9600 зразків за секунду, що достатньо для більшості сенсорних застосувань у мобільній робототехніці.

Плата Arduino Uno R3 обладнана гнучкою системою живлення з кількома варіантами підключення. USB-порт забезпечує живлення 5В з максимальним струмом 500 мА, достатнім для роботи самої плати та невеликих периферійних пристроїв. Для більш потужних навантажень використовується роз'єм зовнішнього живлення, що приймає напругу в діапазоні 6-20В, рекомендовано 7-12В.

Вбудований лінійний стабілізатор напруги на мікросхемі перетворює зовнішню напругу у стабільні 5В для живлення мікроконтролера та периферії. Максимальний вихідний струм стабілізатора становить приблизно 800 мА, проте при високих вхідних напругах необхідно враховувати тепловиділення. Додатково доступна лінія 3.3В з струмом до 50 мА для живлення низьковольтних модулів.

Автоматичне переключення між джерелами живлення забезпечується діодною схемою, тобто при підключенні зовнішнього адаптера USB-живлення автоматично відключається. Світлодіодний індикатор PWR сигналізує про наявність живлення.

Для виконання завдань керування пристроями, збору та передачі даних, нескладної обробки сигналів достатньо й невеликого об'єму пам'яті. Ця версія плати підтримує найпоширеніші комунікаційні інтерфейси: UART, I2C (TWI), SPI, це дозволяє підключати широкий спектр зовнішніх модулів:

Bluetooth, датчики, двигуни, дисплеї тощо. Плата має 14 цифрових входів-виходів, 6 з яких підтримують ШІМ-модуляцію, та 6 аналогових входів, USB-порт для живлення і програмування, яке можливе завдяки мікросхемі CH340G, що створює віртуальний COM-порт, кнопку Reset, кварцовий резонатор на 16 МГц, світлодіодні індикатори живлення і TX, RX для швидкої оцінки стану плати під час роботи. Arduino Uno R3 підтримує живлення і від USB і від зовнішнього джерела 6-20 В, рекомендується 7-12 В.

1.3.4 Екосистема розробки та програмування

Середовище Arduino IDE служить основним інструментом для створення програмного забезпечення плат Arduino. Архітектура IDE базується на Processing IDE та використовує модифікований варіант мови C++, адаптований для роботи з мікроконтролерними системами.

Функціональні особливості Arduino IDE охоплюють різні аспекти, та включають в себе спрощений синтаксис, тобто автоматичне підключення базових бібліотек та відсутність потреби в явному описі функції `main()`, програмна структура має обов'язкові функції `setup()` для початкової конфігурації та `loop()` для циклічного виконання основного алгоритму. Система бібліотек має вбудований доступ до репозиторію готових програмних модулів для різноманітних датчиків та периферійних пристроїв. Автоматизована компіляція перетворює вихідний C++ код у виконуваний hex-файл з подальшим автоматичним завантаженням у мікроконтролер. Монітор послідовного порту є інтегрованим засобом налагодження через UART-інтерфейс.

Процес розробки спрощується завдяки наявності завантажувача – спеціалізованого програмного модуля, розміщеного у Flash-пам'яті мікроконтролера, який забезпечує можливість завантаження користувацького коду через USB-інтерфейс без застосування зовнішніх програматорів. Завантажувач займає 0.5 КБ Flash-пам'яті, однак істотно спрощує цикл

розробки.

Розвинена спільнота розробників сформувала комплексну базу ресурсів, що включає бібліотеки для поширених сенсорних та комунікаційних модулів, зразки програмного коду для стандартних технічних завдань, тематичні форуми та технічну документацію, освітні матеріали та практичні керівництва.

Додатковою перевагою платі є її надійність і низька вартість, вона має доступну ціну навіть для студентських проектів. У вільному доступі є як оригінальні версії, так і велика кількість якісних копій, які мають схожу архітектуру та працюють аналогічно, але при цьому коштують дешевше. Завдяки цьому можна реалізовувати проекти з невеликим бюджетом, але без втрати якості та функціональності.

Простота програмування також була важливим фактором у виборі. Arduino Uno програмується за допомогою Arduino IDE, яка є безкоштовною, має інтуїтивно зрозумілий інтерфейс і доступна для всіх основних операційних систем. Плата швидко визначається комп'ютером як COM-порт, це спрощує завантаження програми одним натисканням кнопки. У разі помилки або необхідності скинути програму, це можна зробити за допомогою кнопки RESET.

1.3.5 Обмеження та особливості Arduino для мобільної робототехніки

Плата має певні недоліки: невеликий об'єм оперативної пам'яті, лише один порт UART, відносно невеликий об'єм флеш-пам'яті.

Обмеження пам'яті SRAM 2 КБ лімітує складність алгоритмів та обсяг змінних для одночасного використання. Тож для мобільних робототехнічних систем це вимагає оптимізації програмного коду та ретельного планування використання пам'яті.

Flash 32 КБ забезпечує достатній обсяг для більшості завдань управління, проте може становити обмежуючий фактор для складних

навігаційних алгоритмів або систем обробки сенсорної інформації.

EEPROM 1 КБ дозволяє зберігати конфігураційні параметри та калібровочні коефіцієнти між циклами живлення.

Також є певні обмеження інтерфейсів, такі як: єдиний UART, який ускладнює одночасну роботу з декількома послідовними пристроями, наприклад Bluetooth-модулем та GPS-приймачем, та обмежена кількість переривань: лише 2 зовнішні переривання (контакти 2 та 3), що може лімітувати швидкість реакції на зміни стану сенсорів.

Продуктивність 8-розрядної архітектури з частотою 16 МГц достатня для базових завдань управління, проте може бути недостатньою для складних математичних обчислень або обробки зображень у реальному часі.

Енергоспоживання в діапазоні 20-50 мА в активному режим: може бути критичним параметром для автономних мобільних платформ, особливо при тривалій експлуатації.

Але в рамках дипломного проекту, плата повністю відповідає вимогам до продуктивності та кількості портів вводу-виводу, які необхідні для забезпечення віддаленого керування мобільним пристроєм через Bluetooth.

1.3.6 Обґрунтування остаточного вибору

Arduino Uno R3 була обрана як найбільш відповідна платформа для дипломного проекту, враховуючи усі вище перерахованих характеристики, як доступність, сумісність з Bluetooth модулем HC-06, підтримку всіх необхідних інтерфейсів, обрана плата є оптимальною платформою для реалізації дипломного проекту.

Чинники, які визначили вибір Arduino Uno R3:

а) технічна відповідність – плата забезпечує необхідний функціонал для реалізації системи управління мобільною платформою – достатню кількість цифрових та аналогових портів, підтримку ШІМ для керування електроприводами, UART для організації Bluetooth-зв'язку;

б) економічні міркування – низька вартість плати та сумісних компонентів забезпечує доступність проекту для студентських розробок без компромісів щодо якості та надійності системи;

в) швидкість реалізації – наявність розвиненої бібліотечної бази та зразків програмного коду скорочує терміни розробки та зменшує ймовірність програмних помилок;

г) перспективи розширення – архітектура проекту дозволяє подальше нарощування функціональності через додавання сенсорних модулів, альтернативних інтерфейсів зв'язку або міграцію на більш потужні плати екосистеми Arduino;

д) освітня складова – практична робота з Arduino забезпечує здобуття компетенцій у програмуванні мікроконтролерних систем, розуміння принципів вбудованих систем та досвід інтеграції периферійних пристроїв.

1.4 Аналіз драйверів керування двигунами для мобільних платформ

Для керування рухом мобільної платформи в цьому проекті застосовується драйвер двох двигунів на базі чіпа L298. Це один із найпопулярніших драйверів двигунів, який дає змогу керувати напрямком і швидкістю обертання двох колекторних двигунів постійного струму. Основною перевагою драйвера L298N[5] є його здатність працювати з широким діапазоном вхідних напруг для живлення мотора – від 5 до 35 В, та підтримка піквої сили тока 3А, що робить його сумісним з більшістю моторів, які використовуються у робототехніці та мобільних платформах. На випадок перевищення допустимої напруги або току, драйвер захистить систему та автоматично вимкне мотори.

Драйвер підтримує максимальну напругу, яка сягає 46 В. Він може працювати з робочим струмом до 2 А, що є задовільним для багатьох моторів. Піковий струм, який він може витримати, досягає 3 А. Драйвер може функціонувати в температурному діапазоні від -20°C до +135°C, що дає

змогу використовувати його в різних умовах довкілля без ризику перегрівання. Максимальна потужність, яку може споживати драйвер, становить 25 Вт, це досить велика потужність для малих мобільних платформ. Модуль має компактні розміри – 43,5 x 43,2 x 29,4 мм і важить лише 26 г, це робить його зручним для інтеграції в різні проекти.

При підключенні драйверу L298N треба враховувати функціональне призначення його виводів. Контакт Vss відповідає за живлення силової частини електродвигунів. Логічна частина драйвера, яка забезпечує керування мікросхемою, під'єднується до стабілізованого джерела напруги 5 В через відповідний вивід +5V. Контакт GND є загальним елементом схеми під час підключення драйвера, він з'єднується з нульовим потенціалом або мінусовою клемою джерела живлення.

Цифрові входи – IN1, IN2, IN3, IN4 відповідають за передачу керуючих сигналів до драйвера. Вони слугують для визначення напрямку обертання кожного з двигунів шляхом подання відповідних рівнів логічного сигналу від мікроконтролера.

Вихідні сигнали подаються на два ряди виводів: OUT1 і OUT2 – для підключення першого двигуна, а і OUT3 і OUT4 – для другого. Ці виходи підключаються до моторів і дають змогу керувати їхнім рухом у різних напрямках і з різною швидкістю, відповідно до заданої логіки роботи.

Значною перевагою драйверу L298N є його універсальність: він підходить для роботи як з колекторними двигунами, так і з кроковими двигунами. Також його можна використовувати в умовах з високими вимогами до надійності, адже він здатен працювати за високого току та температур. Драйвер легкий у використанні та зручний для інтеграції з мікроконтролерами завдяки наявності всіх необхідних виходів для під'єднання двигунів і простим входам для керування.

Під час пошуку оптимального драйверу керування було проведено аналіз серед найпопулярніших аналогів мікросхем: L298N, L293D[6], DRV8833[7] і TB6612FNG[8]. У таблиці 1.2 наведено основні характеристики

кожного з них, що включають кількість каналів, максимальний струм, напругу живлення двигунів, ефективність, переваги та недоліки.

Таблиця 1.2 – Порівняння

Драйвер	Канали	Макс. струм (на канал)	Напруга живлення двигунів	Ефективність	Переваги	Недоліки
L298N	2	2A (3A піковий)	5-46В	Низька	Простий, доступний, з регулятором	Високе тепловиділення, втрати енергії
L293D	2	0.6A (1.2A піковий)	4.5–36В	Низька	Дешевий, вбудовані діоди захисту	Малий струм, перегрів при навантаженні
DRV8833	2	до 1.5A (2A піковий)	2.7В–10.8В	Висока	Компактний, краща енергоефективність	Менша сумісність з великими двигунами
TB6612FNG	2	до 1.2A (3.2A піковий)	2.5-13.5В	Висока	Висока ефективність, мінімум втрат	Трохи дорожчий

Після проведеного аналізу було обрано драйвер L298N, який, хоч і має нижчу енергоефективність порівняно з іншими аналогами, але відрізняється кількома важливими перевагами: широким діапазоном робочої напруги, високою надійністю, доступністю та простотою у використанні. Це робить його доречним вибором для мобільних платформ у навчальних або демонстраційних проектах, де енергоспоживання не є критичним параметром.

1.5 Висновки до розділу

У розділі було проаналізовано мобільні платформи, які використовуються в галузі сучасної робототехніки. Було розглянуто існуючі методи керування роботами, їх конструктивні особливості та вимоги. На основі порівняння основних типів зв'язку, було визначено, що для реалізації даного проекту найбільш раціонально використовувати бездротову технологію Bluetooth, яка забезпечує достатню функціональність та простоту реалізації.

Провівши аналіз найпопулярніших мікроконтролерних платформ для керування мобільним роботом, вибір Arduino Uno R3 був обґрунтований як найкращий варіант, зважаючи на його доступність, простоту програмування, достатню кількість пам'яті.

Також було досліджено можливі варіанти драйверів для керування двигунами. З-поміж них був обраний L298N як надійний і перевірений пристрій, сумісний з обраною платформою мікроконтролера і такий, що відповідає технічним вимогам проекту.

Результати аналізу дозволили сформулювати технічні вимоги до апаратної частини системи та аргументувати вибір її основних компонентів, чим забезпечується надійне підґрунтя для подальшої розробки мобільного робота.

2 РОЗРОБКА КОНСТРУКЦІЇ ТА АПАРАТНОЇ ЧАСТИНИ

2.1 Вибір платформи та компонентів

Розробка дистанційно керованого мобільного робота вимагає ретельного підбору апаратних компонентів. Вибір здійснювався за такими критеріями: наявність компонентів на ринку, їх сумісність, функціональність, енергоспоживання, надійність, простота інтеграції, наявність технічної підтримки та документації. У цьому підрозділі описано кожен з основних компонентів, використаних у проєкті, та доведено доцільність їхнього вибору.

Для реалізації платформи мобільного робота була виготовлена площадка з фанери, вирізана вручну за попереднім кресленням. Це дало змогу індивідуально адаптувати конструкцію до особливостей проєкту. Чотириколісна конфігурація була обрана за простоті реалізації, зручності в управлінні та стійкості при русі. Платформа передбачає встановлення чотирьох двигунів з редукторами та чотирьох коліс, що дозволяє здійснювати рух уперед, назад, повороти вправо та ліво.

Фанерна основа забезпечує достатню жорсткість і легкість, дозволяючи надійно закріпити плату мікроконтролера, драйвер двигуна, акумулятори та Bluetooth модуль. Отвори для розміщення елементів були зроблені відповідно до розміщення елементів, що дозволило досягти компактності та простоти збірки.

Плата Arduino Uno R3 (рисунок 2.1) є центральним елементом системи керування. Серед кількох варіантів, що розглядалися, таких як Arduino Nano, Arduino Mega 2560 та ESP32, було обрано саме цю плату через її численні переваги. Arduino Uno відрізняється легкістю у використанні, значною кількістю доступних прикладів коду та бібліотек, що значно полегшує розробку та налагодження системи.

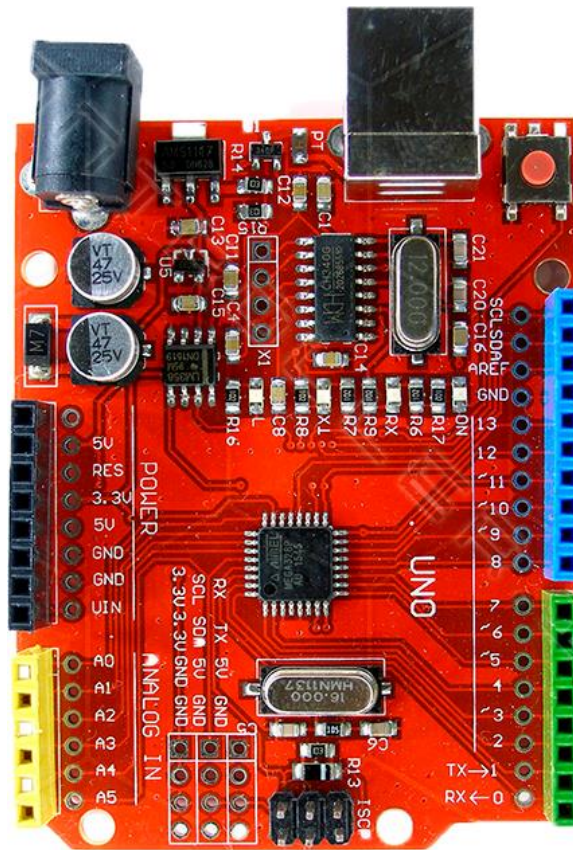


Рисунок 2.1 – Плата Arduino Uno R3

Ця платформа забезпечена достатньою кількістю цифрових і аналогових входів і виходів, потрібних для підключення драйвера двигуна, модуля Bluetooth та інших компонентів проекту. Іншою важливою перевагою є підтримка живлення від зовнішнього джерела з напругою від 7 до 12 В, це дозволяє безпосередньо використовувати акумуляторні батареї. Завдяки наявності великої кількості технічної документації, працювати з цією платою зручно навіть новачкам. Плата побудована на базі чипа ATmega328P. Він працює на тактовій частоті 16 МГц, має 14 цифрових портів, шість з яких підтримують ШІМ, шість аналогових входів, 2 КБ оперативної пам'яті, 32 КБ флеш-пам'яті для зберігання коду та 1 КБ EEPROM.

У цьому проекті було використано драйвер L298N для керування двигунами (рисунок 2.2). Це досить поширений модуль, який дозволяє керувати напрямком і швидкістю обертання двох двигунів постійного струму. Він підходить для роботи з напругою від 5 до 35 вольт і витримує

струм до 2 ампер на кожному каналі. Для зміни швидкості обертання використовується широко-імпульсна модуляція, тому рух можна досить точно налаштувати в залежності від потреб проекту.

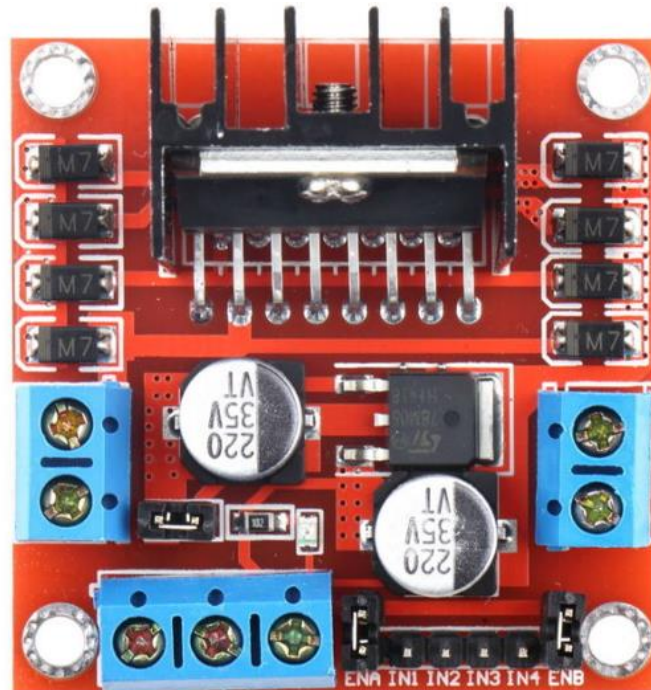


Рисунок 2.2 – драйвер L298N

Цей драйвер має вбудований стабілізатор на 5 вольт, але в моєму випадку він не використовується, оскільки плата Arduino живиться безпосередньо від двох літій-іонних акумулятори на 7,4 В. Такий підхід дозволяє уникнути непотрібних перетворень напруги і спрощує електричну схему. Підключення L298N до Arduino Uno є простим, а сам драйвер зарекомендував себе як надійний компонент, особливо завдяки наявності захисту від перегріву. В рамках цієї розробки модуль забезпечує надійне керування двома двигунами.

У конструкції мобільної платформи використовується чотири компактних мотор-редуктори постійного струму із двостороннім виходом осі (рисунок 2.3). Такий тип моторів забезпечує достатню силу для руху моделі, має просте кріплення до основи та не потребує складного обслуговування.

Завдяки наявності редуктора з передаточним числом 1:48, мотор надає підвищений крутний момент, за допомогою чого конструкцію можна впевнено пересувати навіть за наявності додаткового навантаження від встановленої електроніки.



Рисунок 2.3 – Двигун з редуктором

Номінальна напруга живлення складає близько 6–8 В, що добре узгоджується з загальною схемою живлення пристрою. При цьому струм споживання одного моторчика при напрузі 3.6 В становить приблизно 250 мА. Швидкість обертання без навантаження – близько 170 обертів за хвилину, що цілком достатньо для плавного та керованого руху платформи. Завдяки двосторонньому виходу осі, монтаж коліс або додаткових елементів не викликає труднощів. Самі мотори мають невеликі габарити – близько 64×20×20 мм та масу приблизно 26 грамів, тому добре підходять для мобільних робототехнічних конструкцій, де важлива компактність і зручність розміщення компонентів. У моєму випадку використання чотирьох таких моторів забезпечує не лише стійкість платформи, а й хорошу маневреність.

Така конфігурація дозволяє точно контролювати напрямок і швидкість руху, зокрема виконувати розвороти на місці, змінювати траєкторію, а також забезпечує впевнений рух на різних поверхнях.

Для реалізації бездротового управління в проекті використано модуль HC-06 (рисунок 2.4) – компактний і доступний Bluetooth компонент, що працює в ролі веденого пристрою. Його основна функція – передача даних між смартфоном і мікроконтролером Arduino через стандартний послідовний інтерфейс UART. Це дозволяє легко надсилати команди з телефону й отримувати зворотний зв'язок без використання дротів.

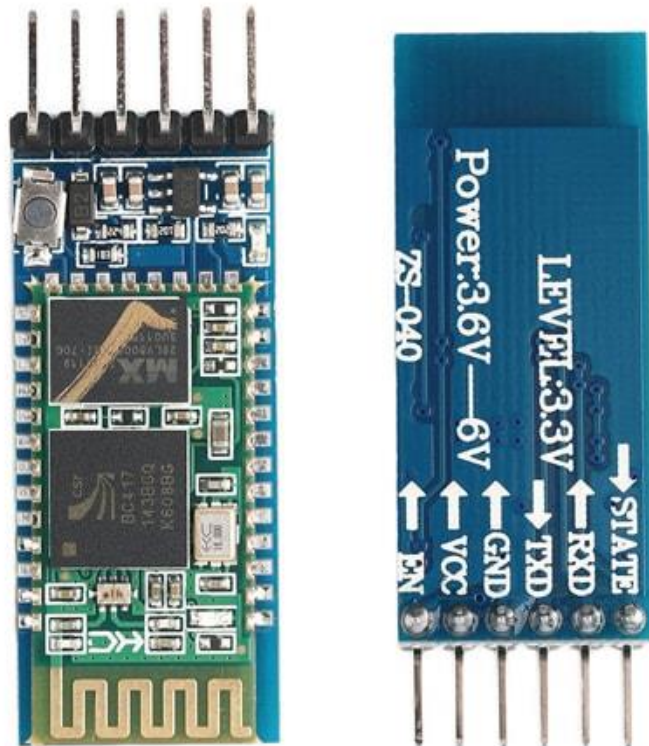


Рисунок 2.4 – Двигун з редуктором

Після живлення модуль автоматично починає працювати, індикатор починає блимати, вказуючи, що пристрій у режимі очікування з'єднання. Коли з'єднання встановлено, то світлодіод горить постійно. Підключення просте, лише кілька контактів – живлення VCC, земля GND, передача й

прийом даних – TXD, RXD. Є також додаткові піни, як EN вимикач модуля та STATE стан з'єднання, які можна за потреби використовувати.

Після підключення до телефону або ноутбука в системі з'являється віртуальний COM-порт, і вся інформація, що надсилається у цей порт, миттєво потрапляє на Arduino і навпаки. Якщо є потреба, параметри модуля можна змінити за допомогою AT-команд, наприклад: змінити назву пристрою, швидкість або пін-код доступу.

HC-06 є оптимальним вибором для дипломного проекту: він дешевий, надійно працює і не вимагає складного конфігурування, а головне ідеально поєднується з Arduino.

Для живлення моделі використовується перевірене рішення на основі двох акумуляторів формату 18650, а саме LGDBHG21865 ємністю 3000 мА·год кожен (рисунок 2.5). Ці енергоємні елементи здатні забезпечити стабільну подачу живлення на всі компоненти конструкції протягом тривалого часу, це особливо важливо при автономній роботі пристрою.



Рисунок 2.5 – Акумулятор LG LGDBHG21865

Акумулятори мають робочу напругу близько 3.6 В і добре витримують високі струмові навантаження, завдяки чому підходять навіть для приводів з підвищеним споживанням. Максимальний постійний струм розряду

становить до 20 А, завдяки чому можна уникнути падінь напруги під час пікових навантажень.

Акумулятори встановлюються у стандартний пластиковий тримач на два елементи (рисунок 2.6). Це зручно тим, що батарейки можна швидко замінити, або зарядити окремо без потреби щось відкручувати чи розбирати.



Рисунок 2.6 – Бокс для акумуляторів 18650 (2 слоти)

Електроживлення до всієї системи подається через простий перемикач з однієї клавіши (рисунок 2.7), який знаходиться на зовнішній частині корпусу. Завдяки цьому вимкнути чи ввімкнути модель - справа секундна, без мороки з дротами або батарейками. У такій конфігурації все працює стабільно протягом кількох годин, навіть якщо навантаження не з найменших. За потреби можна додати ще й стабілізатор напруги, щоб захистити всю схему від можливих скачків у живленні.



Рисунок 2.7 – Перемикач



Рисунок 2.8 – Джампери

Для з'єднання окремих компонентів я скористалась готовими гнучкими проводами з конекторами з обох боків (рисунок 2.8), вони дозволяють швидко під'єднувати елементи без пайки. Водночас з модулем Bluetooth з'єднання виконано шляхом безпосереднього припаювання проводів до контактів, що забезпечує більш надійний контакт і зменшує ймовірність випадкового від'єднання під час роботи.

Після ретельного аналізу технічних вимог проекту та доступних ресурсів було сформовано остаточний набір необхідних елементів. Усі обрані компоненти добре поєднуються між собою як за електричними характеристиками, так і за фізичними параметрами. Цей підхід дозволив забезпечити стабільну роботу моделі, спростив процес її збирання та подальшого обслуговування. Компоненти підбиралися з урахуванням доступності, надійності та зручності інтеграції, завдяки чому вдалося уникнути складних технічних рішень і зосередитися безпосередньо на функціональності пристрою.

2.2. Схема підключення елементів

Весь набір електронних компонентів було з'єднано згідно з індивідуальною схемою, яка забезпечує повноцінну взаємодію між керуючим модулем, виконавчими пристроями та передавальним модулем. Особливу увагу під час побудови системи було приділено надійності з'єднань і зручності подальшого обслуговування чи модифікацій.

Модуль бездротового зв'язку було інтегровано в систему так, щоб забезпечити передачу команд до платформи без фізичного з'єднання з комп'ютером. Вивід RTX з'єднано з цифровим контактом 13 на мікроконтролері, TXD – з 12-м піном. Земля (GND) модуля підключена одночасно і до спільного мінуса на драйвері моторів, і до GND на платі управління, щоб уникнути потенційної різниці напруг. Живлення модуля подається на його вивід VCC, який з'єднаний із виходом 5V на Arduino.

Драйвер керування моторами L298N відповідає за подачу сигналів на виконавчі пристрої – колісні двигуни. До його виходів Output A приєднано два двигуни з лівого боку конструкції, а на виходи Output B два правих. Щоб забезпечити незалежне керування кожним з моторів, було задіяно чотири цифрові лінії керування з мікроконтролера: IN1 поєднано з 2 піном, IN2 з 3 піном, IN3 з 4 піном, і нарешті, IN4 з 5 піном. Така структура дозволяє точно задавати напрямок обертання кожної пари двигунів окремо. Щодо живлення драйвера: на контакт +12V power подається напруга безпосередньо з акумуляторного блоку, який складається з двох акумуляторів формату 18650, з'єднаних у тримачі з двома слотами. Така конфігурація забезпечує сумарну напругу близько 7.4 В, чого достатньо для стабільної роботи обох пар двигунів. Ця ж напруга поступає і на вхід живлення мікроконтролера через вивід UIN, він дозволяє Arduino самостійно стабілізувати напругу до робочого рівня. Контакт power GND з'єднано як із загальною землею акумуляторного блоку, так і з модулем Bluetooth, щоб забезпечити єдиний потенціал у всій схемі.

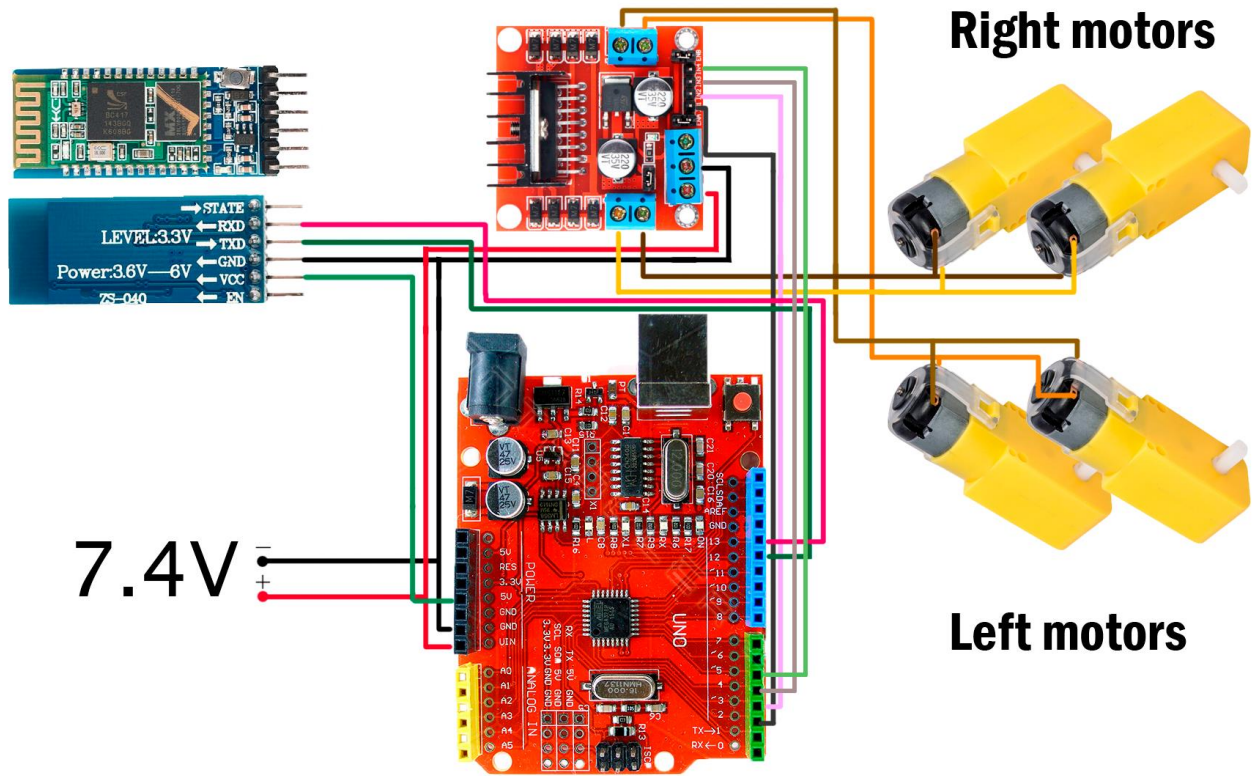


Рисунок 2.9 – Схема підключення

Окремо варто зазначити, що для зручності вмикання та вимикання живлення було встановлено простий клавішний перемикач, що розмикає або замикає ланцюг живлення між батарейним відсіком і рештою системи. Це дає змогу швидко відключати платформу без необхідності виймати акумулятори. Загальна структура з'єднань показана на рисунку 2.9, де візуально відображено логіку підключення всіх основних компонентів системи.

З'єднання між основними модулями виконано за допомогою стандартних монтажних проводів із роз'ємами, де обидва кінці мають штирькові наконечники. Такі дроти забезпечують зручне підключення до макетної плати або модулів із відповідними гніздами. Для більш надійного контакту, деякі з'єднання, зокрема між Bluetooth-модулем і іншими компонентами, були зафіксовані методом пайки, це дозволило уникнути випадкових розривів під час руху пристрою або вібрацій.

Електрична частина моделі спроектована таким чином, щоб забезпечити надійний обмін сигналами та стабільне електроживлення кожного модуля без ускладнення структури самої схеми. Всі компоненти підключені відповідно до їхніх специфікацій, а схема легко масштабується або адаптується до інших типів двигунів або модулів управління.

2.3. Розробка системи живлення комплексу

При побудові апаратної частини пристрою особливу увагу було приділено системі електроживлення, оскільки її правильна реалізація безпосередньо впливає на стабільність функціонування всіх компонентів моделі. Враховуючи різноманітність елементів, що входять до складу комплексу, та їх різні вимоги до електричних параметрів, виникла необхідність створення такої схеми живлення, яка б забезпечувала належну роботу кожного модуля за будь-яких умов.

Джерелом живлення є два акумулятори 18650, які розміщені в стандартному пластиковому тримачі з двома відділами. У моєму випадку я обрала модель LGDBHG21865, яка являє собою літій-іонний акумулятор номінальною ємністю 3000 мАг. Цей вибір був зумовлений потребою у тривалому часі безперервної роботи акумулятора без частого підзаряджання. Крім того, формат 18650 є досить поширеним, тому за необхідності батареї можна легко замінити без використання спеціальних інструментів або складних маніпуляцій. З'єднання між батарейним блоком і силовими входами інших компонентів реалізовано з урахуванням максимальної простоти та надійності. Особливо важливо було передбачити можливість швидкого вимкнення системи. Для цього у коло живлення було додано звичайний клавiшний вимикач, який розташовано між позитивним контактом батарейного відсіку та рештою електричного кола. Таке рішення дозволяє легко знеструмити пристрій, достатньо одного натискання без необхідності виймати акумулятори або втручатись у проводку.

Система живлення в цілому демонструє стабільну роботу в поєднанні з усіма ключовими компонентами включаючи модуль бездротового зв'язку, драйвери двигунів і сам контролер. Надійне енергопостачання є фундаментом для коректного виконання команд а також для уникнення збоїв під час передавання даних або активного руху моделі.

Дякуючи ретельно продуманому підходу до вибору елементів живлення а також правильній організації схеми їх підключення, вдалося досягти високої надійності, простоти обслуговування та адаптивності системи до змін. Всі ці чинники разом формують основу для стабільної та ефективної роботи пристрою в реальних умовах експлуатації.

2.4 Конструктивне рішення та компонування мобільного робота

2.4.1 Вибір матеріалу основи та монтаж компонентів

Для реалізації мобільного робота обрано фанерну основу розмірами 130×85 мм з товщиною 3 мм. Використання подвійного шару фанери забезпечує достатню жорсткість конструкції при невеликій масі. Такий розмір дозволяє розмістити всі необхідні компоненти з урахуванням вимог до електромагнітної сумісності. Фанера як конструкційний матеріал має переваги перед пластиком через простоту обробки та можливість створення кріпильних отворів без спеціального інструменту. Діелектричні властивості деревини також сприяють зменшенню паразитних ємностей між сигнальними ланцюгами.

Компонування системи виконано з урахуванням функціонального призначення кожного блоку. В лівій частині платформи розташовано акумуляторний блок з двох Li-ion елементів напругою 7.4В. Така локалізація джерела живлення дозволяє мінімізувати довжину силових провідників до драйвера двигунів.

Центральну частину займає драйвер L298N, що обґрунтовано його

роллю як основного розподільчого вузла для керування чотирма двигунами. Розташування драйвера посередині забезпечує рівномірну довжину з'єднань до всіх моторів, що важливо для синхронності їх роботи.

Bluetooth-модуль HC-06 встановлено під драйвером двигунів у центральній частині для покращення якості радіозв'язку. Така позиція забезпечує мінімальне екранування металевими частинами та рівномірне покриття сигналу в усіх напрямках.

Arduino Uno розміщено в правій частині платформи з орієнтацією портів підключення до центру. Це забезпечує зручність монтажу з'єднувальних проводів та доступ до роз'єму програмування.

Сигнальні з'єднання між Arduino та драйвером L298N виконані багатожильним дротом для забезпечення гнучкості при монтажі. Довжина проводів мінімізована до 8-10 см для зменшення індуктивних наведень. Кольорове кодування проводів відповідає стандарту: червоний – плюс живлення, чорний – земля, інші кольори - сигнальні ланцюги.

Arduino Uno надійно закріплено на платформі за допомогою високоміцного двостороннього монтажного скотчу. У той же спосіб зафіксовано драйвер L298N, Bluetooth-модуль HC-06 між Arduino та драйвером, а також акумуляторний блок, який встановлено в спеціально виготовлений фанерний тримач.

Використання високоміцного монтажного скотча забезпечує надійне кріплення всіх компонентів при збереженні можливості демонтажу для обслуговування чи модернізації системи.

2.4.2 Переваги обраного конструктивного рішення

Використання фанерної основи замість друкованої плати має кілька переваг для прототипування. Вартість виготовлення знижується в 5-7 разів порівняно з замовленням РСВ, що важливо для навчальних проектів.

Модульність конструкції дозволяє легко модифікувати систему та

додавати нові компоненти без перепроєктування всієї плати. Заміна несправних елементів також спрощується через роз'ємні з'єднання.

Механічна міцність фанерної конструкції достатня для навантажень мобільного робота. Товщина 12 мм подвійного шару забезпечує стійкість до згину навіть при динамічних навантаженнях від роботи двигунів. Габарити 130×85 мм оптимальні для балансу між компактністю та зручністю монтажу. Цей розмір дозволяє розмістити всі компоненти без накладення та забезпечує достатню площу для майбутніх модифікацій системи.

2.5. Висновки до розділу

У цьому розділі був детально розглянутий кожен елемент апаратної частини системи, а також пояснено, чому саме ці компоненти були обрані. Основна мета полягала в тому, щоб побудувати стабільну, функціональну й просту в реалізації платформу для дистанційного керування мобільною моделлю.

Під час вибору контролера важливо було знайти таке рішення, яке легко програмується, підтримує підключення додаткових модулів і не створює проблем з сумісністю. Саме тому було обрано Arduino Uno R3. Ця плата доволі популярна, має все необхідне для швидкого старту, а ще до неї існує багато прикладів і бібліотек, що суттєво спрощує роботу.

Для організації бездротового керування було обрано модуль HC-06, який працює через Bluetooth. Він легко підключається до Arduino через послідовний порт UART і добре працює з мобільними додатками. Таке рішення дозволило реалізувати просте та надійне керування моделлю зі смартфона. Щоб передати сигнали від контролера на колеса, було використано драйвер двигунів L298N. Він дозволяє керувати двома моторами незалежно, завдяки йому можна змінювати напрямок обертання. Це ідеально підходить для того, щоб модель могла рухатися вперед, назад і повертати.

Щодо самих двигунів, було використано чотири мотори постійного

струму по два з кожного боку. Така схема дає змогу моделі здійснювати поворот на місці або виконувати складні маневри. Під час складання були здійснені спроби зробити всі з'єднання надійними, але в той же час залишити можливість швидкого доступу для налагодження. Більшість з'єднань виконана за допомогою джамперів, деякі припаяні, особливо в місцях, де контакт має бути стабільним, наприклад, у Bluetooth модуля. Також залишено вільні порти на Arduino, щоб у майбутньому була можливість додати нові функції, наприклад, датчики або індикатори. Це робить систему гнучкою й відкриває простір для подальших вдосконалень. Загальний вигляд зібраної системи з усіма встановленими компонентами представлено на рисунку 2.10, який демонструє компактність та організованість монтажу всіх елементів.

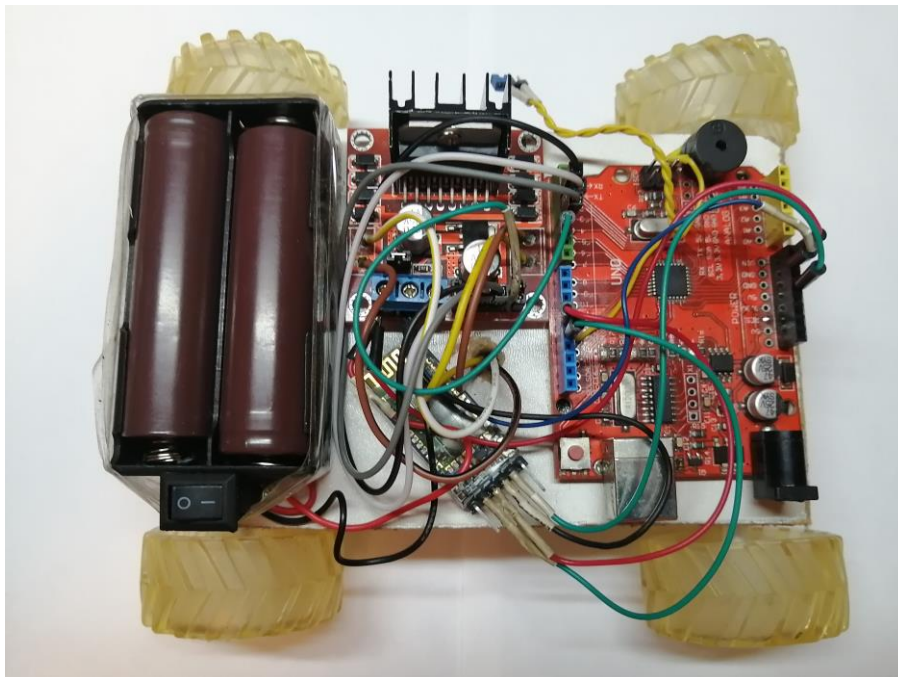


Рисунок 2.10 – Результат роботи

У підсумку, усі обрані компоненти добре між собою працюють, не створюють конфліктів, і їх достатньо для виконання основних функцій. Така апаратна частина стала надійною основою для всього проєкту та дозволила перейти до етапу тестування і вдосконалення моделі.

3 ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ КЕРУВАННЯ

3.1 Алгоритм ручного керування через Bluetooth

Для реалізації дистанційного керування мобільною платформою було розроблено програмне забезпечення на базі фреймворку Qt Creator версії 6.0. Вибір даного інструменту обумовлений його потужними можливостями для роботи з Bluetooth з'єднаннями, зручним інтерфейсом розробки та кросплатформеністю. Програма складається з декількох взаємопов'язаних модулів, кожен з яких виконує специфічні функції у процесі встановлення з'єднання та керування пристроєм.

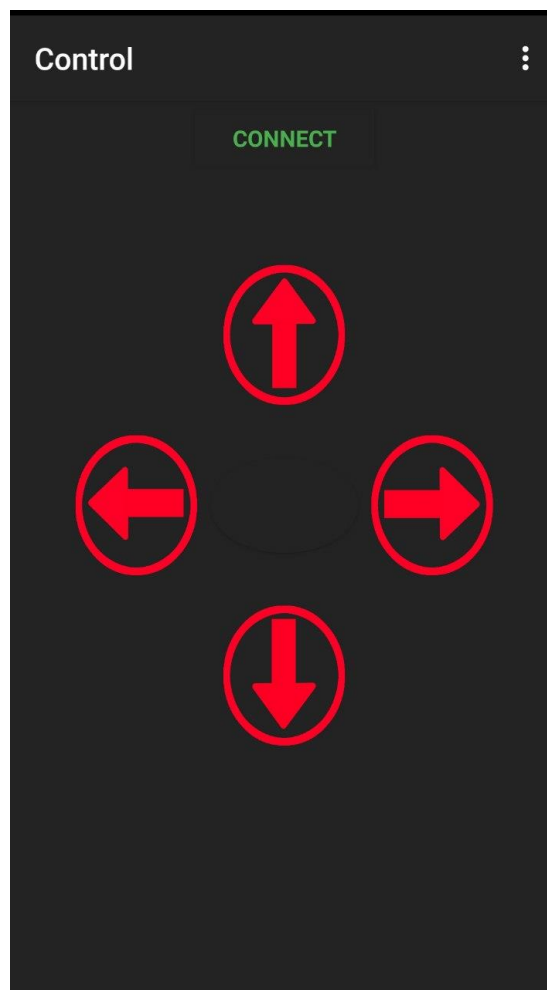


Рисунок 3.1 – Інтерфейс керування

Архітектура програми побудована за принципом розділення відповідальності між компонентами. Головне вікно програми MainWindow містить інтерфейс керування з чотирма основними кнопками для руху моделі: вперед, назад, ліворуч та праворуч. На рисунку 3.1 показано інтерфейс програми з елементами керування зробленими для максимальної зручності користувача та інтуїтивного сприйняття.

Структура програми включає два основні класи. MainWindow – головний клас програми, що відповідає за координацію всіх процесів, він ініціалізує користувацький інтерфейс та встановлює зв'язки між елементами, обробляє натискання кнопок керування з миттєвою реакцією системи, встановлює та підтримує серійне з'єднання через Bluetooth протокол, відправляє команди керування на мобільну платформу у реальному часі, відображає поточний статус системи через рядок повідомлень.

DeviceWindow – допоміжний клас для автоматизованого сканування та вибору Bluetooth пристроїв, він автоматично виявляє всі доступні Bluetooth пристрої у радіусі дії, відображає список знайдених пристроїв у зручному форматі з назвою та адресою, відповідає за фільтрацію та сортування пристроїв для полегшення пошуку потрібного модуля, передачу адреси обраного пристрою до головного вікна для встановлення з'єднання.

Алгоритм роботи програми структурований та можна розділити на два логічно відокремлені етапи: встановлення стабільного з'єднання та безпосереднє керування мобільною платформою.

Перейдемо до детального опису етапу встановлення з'єднання. При запуску програми користувач бачить головне вікно з кнопкою "CONNECT", розташованою у верхній частині інтерфейсу. Натискання цієї кнопки ініціює процес пошуку та підключення до Bluetooth пристроїв. Система автоматично викликає відкриття спеціалізованого вікна сканування пристроїв, яке використовує вбудований модуль QBluetoothDeviceDiscoveryAgent для здійснення пошуку.

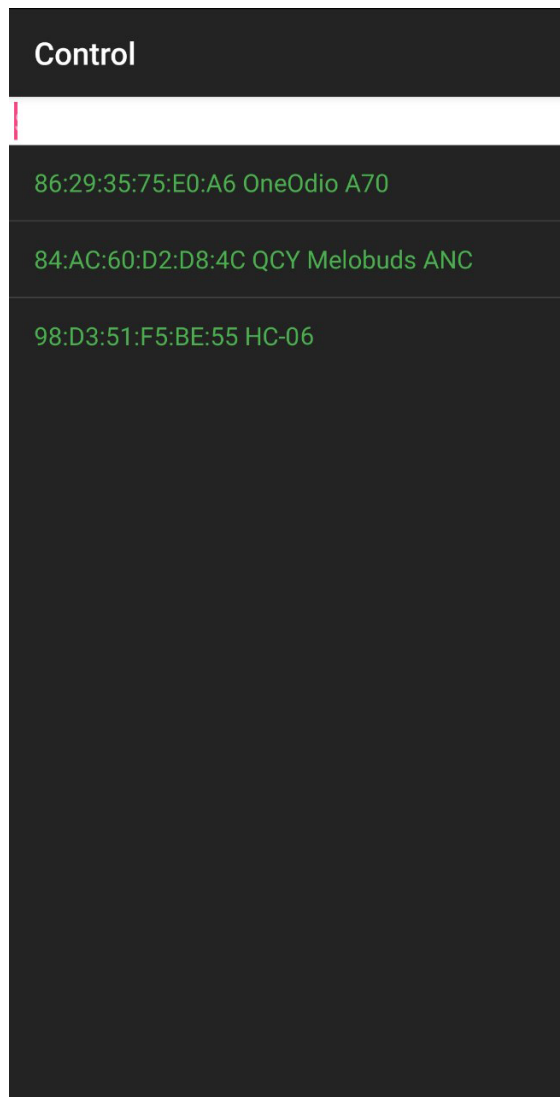


Рисунок 3.2 – Інтерфейс пошуку пристрою

Процес сканування відбувається у фоновому режимі без блокування основного інтерфейсу. Модуль автоматично розпочинає пошук всіх доступних Bluetooth пристроїв у радіусі дії, зазвичай до 10 метрів для класу 2 пристроїв. Під час сканування система аналізує сигнали від усіх активних Bluetooth модулів та формує список доступних підключень.

Знайдені пристрої поступово з'являються у списку із детальним зазначенням їх назви та унікальної MAC-адреси у шістнадцятковому форматі (рисунок 3.2). Список оновлюється динамічно у міру знаходження нових пристроїв, що дозволяє користувачу бачити актуальну інформацію про доступні з'єднання.

Після завершення сканування користувач може обрати потрібний

пристрою зі списку. У нашому випадку це HC-06 модуль, який відображається з відповідною назвою та MAC-адресою. При виборі пристрою програма отримує його унікальну адресу та автоматично намагається встановити серійне з'єднання через QSerialPort клас.

Параметри з'єднання налаштовуються відповідно до специфікації Bluetooth модуля. Швидкість передачі даних встановлюється на стандартному рівні 9600 біт за секунду, що забезпечує стабільну роботу та відповідає заводським налаштуванням HC-06 модуля на стороні Arduino контролера.

Процедура встановлення з'єднання реалізована у методі onDeviceSelected() (лістинг 3.1). У разі успішного встановлення з'єднання система відображає повідомлення про успішне підключення у рядку стану програми. Якщо з'єднання не вдалося встановити через технічні причини, наприклад, пристрій зайнятий іншим додатком або вийшов з радіусу дії, користувач отримує відповідне попереджувальне повідомлення з описом проблеми.

Лістинг 3.1 – З'єднання з пристроєм

```
void MainWindow::onDeviceSelected(const QString &deviceAddress)
{
    serial->setPortName(deviceAddress);
    serial->setBaudRate(QSerialPort::Baud9600);

    if(serial->open(QIODevice::WriteOnly)) {
        ui->statusbar->showMessage("Connected to "+
deviceAddress);
    }
    else {
        QMessageBox::warning(this, "Error", "Fail connection"+
deviceAddress);
    }
}
```

Після успішного встановлення стабільного Bluetooth з'єднання користувач отримує повний доступ до керування мобільною платформою за допомогою чотирьох основних кнопок, розташованих у центральній частині

інтерфейсу. Кожна кнопка має характерний візуальний дизайн у вигляді стрілки відповідного напрямку та червоного кольору для чіткого розуміння функціоналу.

Система команд побудована на принципі відправки унікальних символічних кодів, що забезпечує простоту реалізації на стороні Arduino та мінімізує ймовірність помилок у передачі. Кожна кнопка відправляє на мікроконтролер строго визначену команду у вигляді одиночного символу:

- символ "1" – команда руху вперед, активація обох двигунів у прямому напрямку;
- символ "2" – команда руху назад, реверс обох двигунів;
- символ "3" – команда повороту ліворуч, лівий двигун стоп, правий – вперед;
- символ "4" – команда повороту праворуч, правий двигун стоп, лівий – вперед.

Обробка натискань кнопок (лістинг 3.2) реалізована через механізм слотів Qt framework, що забезпечує практично миттєву реакцію системи на дії користувача без затримок у інтерфейсі. При натисканні будь-якої кнопки керування відповідний слот негайно активується та виконує відправку команди через встановлене серійне з'єднання.

Лістинг 3.2 – Обробка натискань кнопок

```
void MainWindow::on_forward_clicked() {
    serial->write("1");
    ui->statusbar->showMessage("clicked forward");
}
void MainWindow::on_back_clicked() {
    serial->write("2");
}
void MainWindow::on_left_clicked() {
    serial->write("3");
}
void MainWindow::on_right_clicked() {
    serial->write("4");
}
```

Програма використовує односторонню модель передачі даних, де

комп'ютер виступає виключно у ролі передавача команд, а Arduino контролер функціонує як приймач та виконавець отриманих інструкцій. Такий підхід значно спрощує логіку роботи системи, зменшує навантаження на канал зв'язку та мінімізує ймовірність виникнення конфліктів або колізій у процесі передачі даних.

Розроблений користувацький інтерфейс забезпечує максимально інтуїтивне керування завдяки продуманому логічному розташуванню елементів управління та наявності постійного візуального зворотного зв'язку. Рядок стану, розташований у нижній частині головного вікна, безперервно відображає актуальну інформацію про поточний статус підключення, виконані команди та можливі помилки системи.

Додатково система включає механізми обробки помилок та виключних ситуацій. У разі втрати з'єднання з Bluetooth модулем програма автоматично блокує кнопки керування та інформує користувача про необхідність повторного підключення. Це запобігає відправці команд у "нікуди" та забезпечує коректну роботу системи у всіх можливих сценаріях використання.

3.2 Реалізація програмного коду

У даному підрозділі розглядається реалізація програмного забезпечення для дистанційного керування мобільною моделлю через Bluetooth-модуль. Програмний код був написаний у середовищі розробки Arduino IDE, оскільки воно є зручним для роботи з мікроконтролерами Arduino, підтримує швидке завантаження скетчів, має вбудований серійний монітор для відлагодження, а також добре документоване середовище. У даному проєкті використано мікроконтролер Arduino Uno.

Програмний код розділений на кілька логічних блоків. У першій частині відбувається визначення змінних і підключення необхідних бібліотек. Зокрема, підключається бібліотека SoftwareSerial, яка дозволяє

створити додатковий програмний послідовний порт для зв'язку з Bluetooth-модулем. У моєму випадку використовувалися піни 12 і 13 для програмного UART-з'єднання (лістинг 3.3).

Лістинг 3.3 – Ініціалізація Bluetooth зв'язку за допомогою SoftwareSerial

```
#include <SoftwareSerial.h>
SoftwareSerial mySerial(12, 13); // RX, TX
```

Крім цього, були визначені піни для підключення моторів (лістинг 3.4), де M1 і M2 відповідають за напрямок обертання моторів, а E1 і E2 за швидкість, яка регулюється за допомогою широтно-імпульсної модуляції.

Лістинг 3.4 – Ініціалізація пінів моторів

```
int M1 = 2;
int E1 = 3;
int M2 = 4;
int E2 = 5;
```

Також оголошено змінну `speed`, яка задає максимальну швидкість моторів. Встановлено таймер `commandTimeout`, що дозволяє автоматично зупинити робота, якщо протягом певного часу не надходить команда.

Далі в функції `setup()` ініціалізується серійний порт, програмний серійний порт для Bluetooth та встановлюються режими роботи пінів (лістинг 3.5).

Лістинг 3.5 – Ініціалізація пінів моторів

```
void setup() {
  Serial.begin(9600);
  mySerial.begin(9600);
  for (int i = 2; i <= 8; i++) {
    pinMode(i, OUTPUT);
  }
}
```

```
}

```

Функція `loop()` є основною і постійно виконується. Вона перевіряє, чи надходить команда через Bluetooth. Якщо команда отримана, то вона передається у функцію `handleCommand()`, яка розпізнає символ і виконує відповідну дію. Якщо протягом певного часу команда не надходить, робот зупиняється.

Реакція на команди реалізована у функції `handleCommand()` за допомогою оператора `switch` (лістинг 3.6). Команди мають такий вигляд:

- '1' – рух уперед;
- '2' – назад;
- '3' – поворот ліворуч;
- '4' – поворот праворуч;
- '5' – повна зупинка.

Кожна команда передається з мобільного телефону за допомогою спеціального застосунку, який я створила самостійно в середовищі Thinkable. Додаток має кнопки, які відправляють відповідні символи через Bluetooth при натисканні. Таким чином, користувач має змогу вручну керувати машинкою в реальному часі.

Лістинг 3.6 – Обробка команд

```
void handleCommand(char command) {
  switch (command) {
    case '1':
      moveMotors(HIGH, HIGH, speed, speed);
      break;
    case '2':
      moveMotors(LOW, LOW, speed, speed);
      break;
    case '3':
      moveMotors(HIGH, LOW, speed, speed);
      break;
    case '4':
      moveMotors(LOW, HIGH, speed, speed);

```

```

        break;
    case '5':
        stop();
        break;
    }
}

```

Функція `moveMotors()` виконує керування напрямком і швидкістю обертання кожного з двох моторів, відповідно до параметрів (лістинг 3.7).

Лістинг 3.7 – Функція керування моторами

```

void moveMotors(int motor1Dir, int motor2Dir, uint8_t speed1,
uint8_t speed2) {
    digitalWrite(M1, motor1Dir);
    analogWrite(E1, speed1);
    digitalWrite(M2, motor2Dir);
    analogWrite(E2, speed2);
    digitalWrite(Led, LOW);
}

```

Для реалізації безпеки було додано функцію `stop()` (лістинг 3.8), яка зупиняє обидва мотори й одночасно вмикає світлодіод, що сигналізує про зупинку.

Лістинг 3.8 – Функція керування моторами

```

void stop() {
    moveMotors(LOW, LOW, 0, 0);
    digitalWrite(Led, HIGH);
}

```

Загалом, програмний код працює стабільно, забезпечує всі необхідні функції для ручного дистанційного керування, і дозволяє оперативно зупинити модель у разі втрати з'єднання. Створений застосунок для керування Android надсилає прості текстові команди по Bluetooth, а мікроконтролер виконує відповідну реакцію. Така схема дозволяє легко розширювати функціональність, наприклад, додати режим автономного руху

або зворотний зв'язок у майбутньому.

Таким чином, програмне забезпечення відіграє ключову роль у функціонуванні всього комплексу та демонструє ефективну інтеграцію апаратної і програмної частин у рамках курсового проєкту.

3.3 Тестування системи

У даному розділі описано, як розроблена система керування мобільною моделлю була протестована за допомогою Bluetooth-модуля HC-06, Arduino Uno та Qt-додатку. Головна ціллю тестів було переконатися, що всі частини працюють так, як задумано, знайти можливі помилки, якщо вони є, і впевнитися у відповідності вимогам функціоналу.

Під час перевірки системи спочатку було протестовано кожен компонент окремо, чи правильно працює обладнання та чи реагує на сигнали. Потім було звернуто увагу на саму програму: чи правильно вона передає команди і виконує свої функції. Коли окремі частини показали себе добре, було з'єднано все разом і перевірили, як система працює в цілому. І вже наприкінці було проаналізовано чи зручно користуватись розробленим керуванням, чи стабільно все працює в реальних умовах.

3.3.1 Перевірка апаратної частини

На цьому етапі було протестовано кожен компонент по черзі: сам Arduino Uno, Bluetooth-модуль HC-06, драйвер L298N та двигуни.

Щоб переконатися, що сама плата Arduino Uno працює справно, було завантажено простий тестовий скетч для блимання світлодіодом, що вже присутній на платі та підключений до піну 13, та виведення тексту в серійному моніторі. Це дозволило впевнитися, що мікроконтролер реагує на прошивку, не зависає, живлення стабільне, а підключення до комп'ютера через USB працює без перешкод. Також перевірено, що Arduino правильно

обробляє сигнали від модуля Bluetooth та передає їх далі на драйвер моторів.

Основною метою перевірки Bluetooth модулю HC-06 було впевнитися, що він передає і приймає дані як треба. Його було під'єднано до Arduino згідно схеми. Через монітор порту в Arduino IDE відправлено команди з комп'ютера. Перевірено, чи мікроконтролер правильно їх отримує. У результаті модуль HC-06 без проблем приймав команди. Світлодіод на модулі показував, що Bluetooth-з'єднання активне.

Для перевірки драйвера L298N до нього підключили один мотор і надіслали сигнали з Arduino, щоб змусити його рухатися вперед і назад. Використовувався простий скетч з PWM-сигналами. В результаті драйвер правильно змінює напрямок обертання двигуна і контролює швидкість, як і очікувалося.

3.3.2 Функціональне тестування програмного забезпечення

На цьому етапі тестування головна увага приділялася перевірці того, як працюють основні частини програми: інтерфейс, з'єднання через Bluetooth і передача команд на Arduino.

Спочатку була перевірка Bluetooth-з'єднання. Вікно пошуку успішно знайшло всі доступні пристрої, і програма дозволила обрати потрібний. Після вибору, як і передбачалося, адреса пристрою зберігалася, і відразу ж з'являлося повідомлення про те, що виконується спроба з'єднання. У більшості випадків система коректно підключалась, і на екрані з'являлось повідомлення, що пристрій підключено, із зазначенням його адреси. Якщо щось йшло не за планом, з'являлось вікно з відповідним повідомленням про помилку. Все це було реалізовано через QSerialPort і працювало стабільно.

Далі були протестовані кнопки керування, відповідальні за рух моделі. Натискання кожної кнопки запускало певну дію: на Arduino надсилалася команда для початку руху в заданому напрямку. Послідовний монітор показав, що всі команди надходили належним чином і плата реагує на них

правильно, модель рухалася вперед, назад, повертала вліво або вправо, залежно від того, яка кнопка була натиснута. Загалом, під час функціонального тестування вдалося переконатися у правильності роботи програмної частини: всі елементи між собою взаємодіють, і користувач отримує очікувану реакцію від системи.

3.3.3 Висновки за результатами тестування

У результаті всіх проведених перевірок можна зробити висновок, що система позитивної сторони і добре працює в умовах, максимально наближених до реального використання. Під час тестування було перевірено як окремі апаратні компоненти, так і програмне забезпечення. Окремо було перевірено, як працює з'єднання з Bluetooth модулем, як система реагує на дії користувача, та чи доходять команди до Arduino і наскільки правильно на них реагує сама модель.

Проект був протестований в різних ситуаціях, і система належним чином виконувала свої завдання. Було забезпечено стабільне розпізнавання доступних Bluetooth-пристроїв в радіусі дії модуля HC-06, а процес встановлення з'єднання відбувався без значних затримок. Передача команд з керуючої програми на Arduino здійснювалася в режимі реального часу без помітних затримок, що підтверджується реакцією моделі на кожну дію користувача.

Під час тестування користувацький інтерфейс продемонстрував чітку структуру та інтуїтивно зрозумілу взаємодію: кнопки управління стабільно спрацьовували при кожному натисканні, а повідомлення про стан з'єднання та помилки відображалися своєчасно і відповідали реальному стану системи. Завдяки цьому було забезпечено комфортну роботу користувача, без потреби в додаткових інструкціях чи роз'ясненнях.

3.4 Висновки до розділу

Третій розділ даної дипломної роботи був присвячений практичній частині проекту, зокрема, впровадженню та тестуванню розробленої системи. Варто зазначити, що саме цей етап став своєрідним підсумком попередніх розділів, оскільки теоретичні знання, які були розглянуті раніше, тут трансформувались у конкретні дії та рішення, що мали на меті створити працездатний і функціональний продукт. Під час реалізації довелося неодноразово повертатися до попередніх етапів, переосмислювати окремі деталі, вносити корективи у схему або змінювати логіку програми для досягнення стабільної роботи системи.

Система будувалася поступово, починаючи з підключення та налаштування апаратного забезпечення. На цьому етапі особливу увагу було приділено вибору необхідних компонентів. Далі розпочалася робота над програмною частиною, яка включала в себе написання коду для роботи з мікроконтролером Arduino та створення програми для керування пристроєм з використанням середовища Qt. Розробка додатку виявилася цікавою, але подекуди складною. Працюючи з Bluetooth в Qt, ми зіткнулися з деякими труднощами, пов'язаними з сумісністю і коректною роботою модуля HC-06 в різних умовах. Також важливо було правильно організувати обмін даними між головним вікном та вікном вибору пристрою. Для цього ми використали сигнали та слоти, які є одним з ключових механізмів в Qt. Здається, що сама логіка підключення не надто складна, але важливо було коректно реалізувати зв'язок між обраним пристроєм та ініціалізацією послідовного порту. У деяких випадках програма не відкривала порт, що вимагало додаткової перевірки доступності пристрою та обробки можливих помилок.

Що стосується тестування, воно було проведено в умовах, наближених до реальних, включаючи перевірку реакції системи на різні команди, стабільність Bluetooth з'єднання, затримки відповіді, а також поведінку в разі втрати зв'язку або відсутності доступних пристроїв. Результати тестування показали, що система може функціонувати без будь-яких серйозних збоїв.

Таким чином, можна сказати, що розроблена система повністю виконала поставлені перед нею завдання. Під час роботи над проектом мені вдалося не тільки реалізувати запланований функціонал, але й покращити свої практичні навички роботи з мікроконтролерами, протоколами бездротового зв'язку та програмування графічного інтерфейсу користувача.

ВИСНОВКИ

У ході виконання кваліфікаційної роботи було досягнуто поставленої мети, а саме розробити мобільну модель із дистанційним керуванням за допомогою Bluetooth модулю, яка побудована на базі мікроконтролера Arduino Uno. В процесі створення працездатної системи був отриманий практичний досвід в проектуванні, монтажі, програмуванні та тестуванні подібних рішень.

На етапі теоретичного дослідження було проведено огляд існуючих мобільних роботів, які сьогодні використовуються в різних галузях, від промисловості до освітніх цілей. Було визначено, що схеми керування на основі мікроконтролерів є найбільш ефективними для простих мобільних платформ завдяки їх гнучкості, доступності та широкому розповсюдженню. Платформа Arduino виявилась особливо зручною, вона має багато вже готових бібліотек, гарну документацію, а також активно підтримується спільнотою розробників. Саме тому було прийнято рішення використати цю платформу як основу для створення апаратної частини системи.

Другий розділ роботи був присвячений вибору апаратної складової. Всі елементи були ретельно підібрані, зокрема плата керування, модуль Bluetooth HC-06, драйвер двигунів L298N, двигуни постійного струму та інші допоміжні компоненти. Була складена електрична схема, в якій особливу увагу приділено правильному підключенню та живленню кожного елемента. З урахуванням обмежень по напрузі та струму, також була розроблена схема живлення, яка забезпечує стабільну роботу системи без перевантажень.

У третьому розділі розглядалася програмна частина, яка виявилася не менш важливою, ніж апаратна. Саме правильна реалізація коду дала можливість точно й надійно керувати моделлю. Застосунок, написаний у середовищі Qt, дозволяє користувачу обирати пристрій для з'єднання, встановлювати Bluetooth зв'язок та надсилати команди руху. Було

реалізовано інтуїтивно зрозумілий інтерфейс, який виконує поставлені функції. Також було організовано просту, але надійну логіку обробки подій, що дозволило забезпечити стабільність з'єднання. Особливу увагу було приділено обміну даними між модулями та обробці ситуацій, коли з'єднання втрачено або пристрій недоступний.

У процесі тестування вдалося переконатися, що система працює у відповідності до очікувань. Під час практичної перевірки модель реагувала на команди з додатку швидко і без помітних затримок. Також перевірено стабільність Bluetooth з'єднання. Тестувалася не тільки працездатність моделі в цілому, але й кожного окремого модуля, що допомогло виявити деякі недоліки в початковій реалізації. Після внесення змін система стала більш стійкою до збоїв.

Отже, результати кваліфікаційної роботи підтвердили, що навіть із відносно простих компонентів можна створити систему, яка виконує свої функції повністю і стабільно. При цьому, важливо не тільки правильно підібрати апаратні елементи, а й грамотно реалізувати програмну частину, продумати логіку, обробку помилок та тестування в різних умовах. Також варто зазначити, що робота над цим проектом дозволила поглибити знання з програмування, схемотехніки, роботи з мікроконтролерами та бездротовими технологіями, що є надзвичайно корисним як з практичної точки зору, так і з точки зору подальшої професійної діяльності. Виконання даного дипломного проєкту дозволило вирішити поставлену задачу та відкрило перспективи для подальшого розвитку. У майбутньому до системи можна буде додати нові функції, наприклад, датчики перешкод, GPS-навігацію. Це зробить систему ще більш функціональною та пристосованою для використання в реальних умовах.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Arduino Uno Rev3 datasheet. URL: <https://docs.arduino.cc/resources/datasheets/A000066-datasheet.pdf> (дата звернення: 15.05.2025).
2. Arduino Mega 2560 datasheet. URL: <https://docs.arduino.cc/resources/datasheets/A000005-datasheet.pdf> (дата звернення: 15.05.2025).
3. Arduino Nano datasheet. URL: <https://docs.arduino.cc/resources/datasheets/A000067-datasheet.pdf> (дата звернення: 15.05.2025)
4. Microchip ATmega328PB datasheet. URL: <https://datasheet4u.com/pdf-down/A/T/m/ATmega328PB-Microchip.pdf> (дата звернення: 15.05.2025).
5. Full Bridge Motor Driver Dual L298N datasheet. URL: https://cdn.sparkfun.com/assets/7/1/d/6/c/Full-Bridge_Motor_Driver_Dual_-_L298N.pdf (дата звернення: 15.05.2025).
6. Texas Instruments L293 Dual H-Bridge Motor Driver. URL: <https://www.ti.com/lit/ds/symlink/l293.pdf> (дата звернення: 15.05.2025).
7. Texas Instruments DRV8833 Dual Low-Voltage H-Bridge Motor Driver. URL: <https://www.ti.com/lit/ds/symlink/drv8833.pdf> (дата звернення: 15.05.2025).
8. Toshiba TB6612FNG Dual Motor. URL: <https://cdn-shop.adafruit.com/product-files/1944/TB6612FNG+datasheet.pdf> (дата звернення: 15.05.2025).
9. Blum J. Exploring Arduino: Tools and Techniques for Engineering Wizardry. 2nd ed. John Wiley and Sons Ltd, 2019. 512 p.
10. Fedorchenko V., Yeroshenko O., Shmatko O., Kolomiitsev O., Omarov M. Password hashing methods and algorithms on the .NET platform. Advanced Information Systems. 2024. Vol. 8(4). P. 82–92. doi:

<https://doi.org/10.20998/2522-9052.2024.4.11>

11. Tanenbaum A. S., Bos H. Modern Operating Systems. 4th ed. Pearson, 2014. 1136 p.

12. Husky A300 Data Sheet.
URL: https://gomarketing.ottomotors.com/l/92812/2024-10-14/95nk19/92812/1728925412aC5a34ww/Husky_A300_DataSheet_Oct24.pdf (date of access: 15.05.2025).

13. Vijayalakshmi S., Archana M. Robotic Car Using Arduino with Bluetooth Controller. International Journal of Innovative in Science and Engineering. 2019. Vol. 1, no. 1. P. 8.

14. Meteab W. K., ALRikabi H. Th. S., Al Sultani S. A. H., Aljazaery I. A. Controlling and Monitoring a Robot-Car Based on Smart Phone Applications. International Conference on Sustainable Engineering and Technology. 2021. Vol. 1094, no. 1.

15. Monk S. Programming Arduino: Getting Started with Sketches, Second Edition (Tab). 2nd ed. McGraw Hill TAB, 2016. 192 p.

16. McComb G. Robot Builder's Bonanza. 5th ed. McGraw Hill TAB, 2018. 704 p.