Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Автоматизації, комп'ютерно-інтегрованих технологій та систем
(повна назва)

Кафедра Комп'ютерно-інтегрованих технологій, автоматизації, робототехніки та безпекової інженерії
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти перший (бакалаврський)
КТ 1. Програмно-апаратний комплекс для сортування поштових відправлень.
ПТ 1. Розроблення апаратного забезпечення.
(тема)

Виконав:
здобувач 4 року навчання,
групи АКТАКІТ-22-2
Владислав АРТЮХ
(власне ім'я, прізвище)

Спеціальність 151 Автоматизація та комп'ютерно-інтегровані технології
(код і повна назва спеціальності)

Тип програми освітньо-професійна

Освітня програма Автоматизація та комп'ютерно-інтегровані технології
(повна назва освітньої програми)

Керівник доц. Дмитро НІКІТІН
(посада, власне ім'я, прізвище)

Допускається до захисту

Завідувач кафедри _____ Ігор НЕВЛЮДОВ
(підпис) (власне ім'я, прізвище)

2026 р

Я, Артюх Владислав Сергійович, як здобувач вищої освіти ХНУРЕ, розумію і підтримую політику закладу із академічної доброчесності. Я не надавав і не одержував недозволену допомогу під час підготовки кваліфікаційної роботи. Я не використовував штучний інтелект для підготовки кваліфікаційної роботи. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

"13" липня 2026 р.  Владислав АРТЮХ

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Факультет Автоматизації, комп'ютерно-інтегрованих технологій та систем
Кафедра Комп'ютерно-інегрованих технологій, автоматизації, робототехніки та безпекової інженерії
Рівень вищої освіти перший (бакалаврський)
Спеціальність 151 Автоматизація та комп'ютерно-інтегровані технології
(код і повна назва)
Тип програми освітньо-професійна
Освітня програма Автоматизація та комп'ютерно-інтегровані технології
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри КІТАРБІ _____
(підпис)
« ___»_____2026 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

Здобувачеві Артюху Владиславу Серійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи КТ 1. Програмно-апаратний комплекс для сортування поштових відправлень. ПТ 1. Розроблення апаратного забезпечення.
Затверджена наказом по університету від «04» 06 2026 р. № 472 Ст
2. Термін подання здобувачем роботи до екзаменаційної комісії 2026 р.
3. Вихідні дані до роботи 1. Мобільна роботизована платформа для автоматизованого сортування поштових відправлень.
2. Raspberry Pi 5, мікроконтролери STM32H750VGT6 і STM32F103, драйвери CW-5045, крокові двигуни, ультразвукові датчики та LCD-дисплей.
3. Інтерфейси SPI, UART і GPIO; акумуляторна система живлення з BMS та DC/DC-перетворювачами; модульна конструкція з елементами FDM-друку.
4. Перелік питань, що потрібно опрацювати в роботі 4.1 Вступ;
4.2 Аналіз сучасних систем автоматизованого сортування, мобільних роботизованих платформ та елементної бази;
4.3 Розроблення архітектури, апаратного забезпечення та системи керування програмно-апаратного комплексу;
4.4 Математичне моделювання руху та розроблення програмного забезпечення комплексу;
4.5 Розроблення конструкції мобільної роботизованої платформи та експериментальна перевірка її працездатності;
4.6 Висновки.

5. Перелік графічного матеріалу із зазначенням обов'язкових креслеників, схем, плакатів, комп'ютерних ілюстрацій: демонстраційний матеріал у форматі PPT, А4 – 10 сторінок.

6. Консультанти розділів роботи (п.6 включається до завдання 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	Термін виконання етапів роботи	Приміт-ка
1	Аналіз літературних джерел	27.04.2026 – 10.05.2026	виконано
2	Аналіз систем сортування та мобільних платформ	11.05.2026 – 24.05.2026	виконано
3	Проєктування апаратного забезпечення комплексу	25.05.2026 – 07.06.2026	виконано
4	Розроблення та виготовлення мобільної платформи	08.06.2026 – 21.06.2026	виконано
5	Монтаж, налагодження та випробування комплексу	22.06.2026 – 30.06.2026	виконано
6	Оформлення пояснювальної записки	01.07.2026 – 03.07.2026	виконано
7	Подання роботи на перевірку Інтернет-сервісом StrikePlagiarism	04.07.2026	
8	Подання роботи на рецензію	06.07.2026	
9	Подання роботи на підпис зав. кафедри	09.07.2026	
11	Подання кваліфікаційної роботи в ЕК	12.07.2026	

Дата видачі завдання 27.04.2026 р.

Здобувач _____ Владислав АРТЮХ
(підпис) (посада, власне ім'я, прізвище)

Керівник роботи _____ доц. Дмитро НІКІТІН
(підпис) (власне ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка: 114 с., 25 табл., 17 рис., 2 дод, 44 джерела.

ПРОГРАМНО-АПАРАТНИЙ КОМПЛЕКС, МОБІЛЬНА РОБОТИЗОВАНА ПЛАТФОРМА, СОРТУВАННЯ ПОШТОВИХ ВІДПРАВЛЕНЬ, RASPBERRY PI 5, КРОКОВИЙ ДВИГУН, CW-5045, SPI, УЛЬТРАЗВУКОВИЙ ДАТЧИК, FDM-ДРУК.

Об’єктом дослідження є процес автоматизованого сортування поштових відправлень із використанням мобільних роботизованих платформ.

Предметом дослідження є апаратне забезпечення програмно-апаратного комплексу для сортування поштових відправлень, зокрема структура взаємодії модулів, сенсорна підсистема, система керування та виконавчі механізми.

Метою роботи є розроблення апаратного забезпечення комплексу, що забезпечує узгоджену роботу мобільної платформи, Raspberry Pi 5, контролера STM32H750VGT6, локальних модулів STM32F103, датчиків, драйверів крокових двигунів і засобів обміну даними.

У роботі проаналізовано сучасні системи автоматизованого сортування та мобільні роботизовані платформи, обґрунтовано вибір багаторівневої модульної архітектури, елементної бази, інтерфейсів зв’язку й підсистеми живлення.

Розроблено структурну організацію комплексу, схему взаємодії Raspberry Pi 5 зі STM32, підсистему ультразвукового контролю простору, керування виконавчими механізмами через драйвери CW-5045 та конструкцію мобільної роботизованої платформи з елементами FDM-друку.

Практичне значення роботи полягає у створенні масштабованої апаратної основи для автоматизації транспортування та сортування поштових відправлень у логістичних системах із можливістю подальшої модернізації.

ABSTRACT

Explanatory note: 114 pages, 25 tables, 17 figures, 2 appendices, 44 references.

HARDWARE-SOFTWARE COMPLEX, MOBILE ROBOTIC PLATFORM, PARCEL SORTING, RASPBERRY PI 5, STEPPER MOTOR, CW-5045, SPI, ULTRASONIC SENSOR, FDM PRINTING.

The object of the research is the process of automated parcel sorting using mobile robotic platforms.

The subject of the research is the hardware of a hardware-software complex for parcel sorting, including the interaction structure of hardware modules, the sensor subsystem, the control system and the actuating mechanisms.

The aim of the qualification work is to develop the hardware of the complex that ensures coordinated operation of the mobile platform, Raspberry Pi 5, STM32H750VGT6 central controller, STM32F103 local modules, sensors, stepper motor drivers and data exchange interfaces.

The work analyzes modern automated parcel sorting systems and mobile robotic platforms, substantiates the choice of a multi-level modular architecture, component base, communication interfaces and power supply subsystem.

The structural organization of the complex, the interaction scheme between Raspberry Pi 5 and STM32 microcontrollers, the ultrasonic space monitoring subsystem, CW-5045-based actuator control and the mobile robotic platform design with FDM-printed elements have been developed.

The practical value of the work lies in creating a scalable hardware basis for automating parcel transportation and sorting processes in logistics systems with the possibility of further modernization.

ЗМІСТ

С.

Перелік скорочень 9
Вступ 10
1 Аналіз сучасних систем автоматизованого сортування поштових відправлень та обґрунтування напрямку розробки 13
1.1 Аналіз сучасного стану автоматизації логістичних процесів 13
1.2 Аналіз існуючих систем сортування поштових відправлень 16
1.3 Аналіз мобільних роботизованих платформ для автоматизованого сортування поштових відправлень 20
1.4 Аналіз сучасної елементної бази мобільних роботизованих платформ .. 23
1.5 Постановка задачі бакалаврської роботи 27
2 Розроблення апаратного забезпечення програмно-апаратного комплексу для сортування поштових відправлень 29
2.1 Загальна архітектура програмно-апаратного комплексу 29
2.2 Розроблення структурної організації апаратного забезпечення програмно-апаратного комплексу 35
2.2.1 Розроблення центрального модуля керування 36
2.2.2 Розроблення локальної сенсорної підсистеми 40
2.2.3 Розроблення підсистеми керування виконавчими механізмами 43
2.2.4 Розроблення підсистеми електроживлення програмно-апаратного комплексу 47
2.2.5 Розроблення системи інформаційної взаємодії між апаратними модулями програмно-апаратного комплексу 51
2.3 Математичне моделювання та дослідження системи керування рухом мобільної роботизованої платформи 56
3 Розроблення програмного забезпечення програмно-апаратного комплексу для сортування поштових відправлень 61
3.1 Загальна архітектура програмного забезпечення 61

3.2 Програмне забезпечення центрального контролера STM32H750 63
3.2.1 Архітектура програмного забезпечення STM32H750 64
3.2.2 Ініціалізація апаратних ресурсів 65
3.2.3 Організація головного циклу програми 66
3.2.4 Організація обміну даними між Raspberry Pi та STM32H750 67
3.2.5 Алгоритм керування кроковими двигунами 68
3.2.6 Розроблення програмного забезпечення Raspberry Pi 5 73
3.2.7 Забезпечення надійності, діагностики та налагодження програмно-апаратного комплексу 75
3.3 Розроблення конструкції мобільної роботизованої платформи 79
4 Експериментальна перевірка працездатності програмно-апаратного.комплексу 85
4.1 Мета та порядок проведення експериментальної перевірки 85
4.2 Перевірка роботи центрального модуля керування 86
4.3 Перевірка інформаційного обміну між Raspberry Pi 5 та STM32H750VGT6 88
4.4 Перевірка локальної сенсорної підсистеми 91
4.5 Перевірка підсистеми керування виконавчими механізмами 92
4.6 Перевірка підсистеми електроживлення 94
4.7 Оцінювання результатів експериментальної перевірки 95
5 Охорона праці 98
5.1 Аналіз умов експлуатації програмно-апаратного комплексу 98
5.2 Небезпечні та шкідливі фактори під час експлуатації комплексу 99
5.3 Заходи безпечної експлуатації програмно-апаратного комплексу 101
5.4 Висновки до розділу 103
Висновки 105
Перелік джерел посилання 108
Додаток А Демонстраційні матеріали 113

ПЕРЕЛІК СКОРОЧЕНЬ

ПАК – програмно-апаратний комплекс;

ПЗ – програмне забезпечення;

ТЗ – технічне завдання;

AGV – Automated Guided Vehicle, автоматизований транспортний засіб;

AMR – Autonomous Mobile Robot, автономний мобільний робот;

BMS – Battery Management System, система керування акумуляторною батареєю;

CAD – Computer-Aided Design, система автоматизованого проєктування;

DC/DC – Direct Current/Direct Current Converter, перетворювач постійної напруги;

FDM – Fused Deposition Modeling, моделювання методом пошарового наплавлення;

GPIO – General-Purpose Input/Output, універсальні входи-виходи;

I²C – Inter-Integrated Circuit, послідовний інтерфейс обміну даними;

IMU – Inertial Measurement Unit, інерціальний вимірювальний модуль;

IoT – Internet of Things, Інтернет речей;

LCD – Liquid Crystal Display, рідкокристалічний дисплей;

PWM – Pulse-Width Modulation, широтно-імпульсна модуляція;

QR – Quick Response Code, двовимірний код швидкого зчитування;

SPI – Serial Peripheral Interface, послідовний периферійний інтерфейс;

UART – Universal Asynchronous Receiver-Transmitter, універсальний асинхронний приймач-передавач;

USB – Universal Serial Bus, універсальна послідовна шина.

ВСТУП

Сучасний розвиток логістичної галузі безпосередньо пов’язаний із постійним зростанням обсягів електронної комерції, підвищенням вимог до швидкості доставки товарів та необхідністю забезпечення високої точності обробки поштових відправлень. Логістичні підприємства щоденно обробляють значну кількість посилок, які відрізняються за габаритами, масою, напрямками доставки та умовами транспортування. За таких умов традиційні методи ручного сортування поступово втрачають ефективність, оскільки потребують значних трудових ресурсів, характеризуються обмеженою продуктивністю та залежать від людського фактора.

Одним із найбільш перспективних напрямів підвищення ефективності логістичних процесів є впровадження автоматизованих програмно-апаратних комплексів, які поєднують роботизовані платформи, сенсорні системи, засоби комп’ютерного зору, мікроконтролерні системи керування та сучасні інформаційні інтерфейси. Використання таких рішень дозволяє скоротити час виконання операцій сортування, підвищити точність переміщення вантажів, зменшити вплив людського фактора та забезпечити можливість масштабування логістичних систем відповідно до зростання навантаження.

Особливе місце серед засобів автоматизації займають мобільні роботизовані платформи, які можуть автономно переміщуватися у межах складських або сортувальних приміщень, транспортувати поштові відправлення між технологічними зонами та взаємодіяти з іншими елементами автоматизованої системи. На відміну від стаціонарних конвеєрних систем, мобільні платформи забезпечують більшу гнучкість технологічного процесу, оскільки їх маршрути можуть змінюватися відповідно до поточних умов роботи. Це є особливо важливим для логістичних центрів, у яких необхідно швидко адаптуватися до змін потоків відправлень, конфігурації робочих зон та режимів

роботи обладнання.

Разом із тим створення ефективної мобільної роботизованої платформи потребує комплексного підходу до розроблення апаратного забезпечення. Необхідно забезпечити узгоджену роботу системи керування, виконавчих механізмів, сенсорної підсистеми, системи електроживлення та засобів інформаційного обміну. Особливе значення має правильний розподіл функцій між обчислювальними модулями. Задачі високорівневого керування, обробки зображень та формування логіки сортування доцільно виконувати на одноплатному комп’ютері, тоді як задачі реального часу, пов’язані з керуванням двигунами, опитуванням датчиків та формуванням сигналів керування, повинні виконуватися мікроконтролерами.

Актуальність теми кваліфікаційної роботи обумовлена необхідністю розроблення апаратного забезпечення програмно-апаратного комплексу для сортування поштових відправлень, який поєднує мобільну роботизовану платформу, розподілену систему керування, сенсорну підсистему та сучасні цифрові інтерфейси взаємодії. Такий комплекс може бути використаний як основа для створення автоматизованих систем транспортування та сортування, а також як експериментальна платформа для подальшого розвитку роботизованих логістичних рішень.

Метою кваліфікаційної роботи є розроблення апаратного забезпечення програмно-апаратного комплексу для сортування поштових відправлень, що забезпечує ефективну взаємодію мобільної роботизованої платформи, центрального контролера, локальних сенсорних модулів, виконавчих механізмів та засобів інформаційного обміну.

Об’єктом дослідження є процес автоматизованого сортування поштових відправлень із використанням мобільних роботизованих платформ.

Предметом дослідження є апаратне забезпечення програмно-апаратного комплексу для автоматизованого сортування поштових відправлень, принципи

його побудови, структура взаємодії апаратних модулів, сенсорна підсистема, система керування та виконавчі механізми мобільної роботизованої платформи.

Для досягнення поставленої мети необхідно вирішити такі завдання:

– провести аналіз сучасних систем автоматизованого сортування поштових відправлень і мобільних роботизованих платформ;

– проаналізувати сучасну елементну базу, придатну для побудови програмно-апаратного комплексу;

– обґрунтувати вибір багаторівневої модульної архітектури комплексу;

– розробити структуру центрального модуля керування на базі STM32H750VGT6 та локальної сенсорної підсистеми на базі STM32F103;

– розробити підсистему керування виконавчими механізмами з використанням драйверів CW-5045;

– розробити підсистему електроживлення та структуру інформаційної взаємодії між апаратними модулями;

– розробити математичну модель руху мобільної роботизованої платформи, виконати розрахунок основних параметрів системи керування та оцінити її динамічні характеристики;

– розробити конструкцію мобільної роботизованої платформи, виготовити та скласти її експериментальний зразок;

– провести експериментальну перевірку працездатності запропонованих апаратних і програмних рішень;

– розробити заходи з охорони праці та безпечної експлуатації програмно-апаратного комплексу.

Для вирішення поставлених завдань передбачається застосувати методи системного аналізу, структурного та модульного проєктування, порівняльного аналізу технічних рішень, математичного моделювання систем керування, тривимірного моделювання, адитивного виготовлення деталей, а також експериментальної перевірки працездатності окремих вузлів програмно-апаратного комплексу.

Очікуване практичне значення роботи полягає у можливості створення апаратної основи програмно-апаратного комплексу, призначеного для автоматизації процесів транспортування та сортування поштових відправлень. Передбачається, що запропонована модульна структура забезпечить можливість подальшого розширення комплексу, підключення додаткових сенсорних і виконавчих модулів та його адаптації до різних умов експлуатації.

Очікуване практичне значення роботи полягає у можливості створення апаратної основи програмно-апаратного комплексу, призначеного для автоматизації процесів транспортування та сортування поштових відправлень. Передбачається, що запропонована модульна структура забезпечить можливість подальшого розширення комплексу, підключення додаткових сенсорних і виконавчих модулів та його адаптації до різних умов експлуатації.

Кваліфікаційну роботу оформлено згідно з ДСТУ [1-2], а також з використанням навчально-методичних матеріалів [3–5].

1 АНАЛІЗ СУЧАСНИХ СИСТЕМ АВТОМАТИЗОВАНОГО СОРТУВАННЯ ПОШТОВИХ ВІДПРАВЛЕНЬ ТА ОБҐРУНТУВАННЯ НАПРЯМКУ РОЗРОБКИ

1.1 Аналіз сучасного стану автоматизації логістичних процесів

Стрімкий розвиток електронної комерції, глобалізація ринку товарів та цифрова трансформація логістичної галузі призвели до істотного збільшення обсягів поштових відправлень. Щороку оператори поштового зв'язку та логістичні компанії обробляють мільйони посилок різних розмірів і категорій, що висуває підвищені вимоги до швидкості, точності та надійності виконання операцій із приймання, сортування, транспортування та доставки. За таких умов використання традиційних ручних методів обробки вантажів стає економічно менш ефективним, оскільки вони характеризуються значними витратами часу, високою ймовірністю виникнення помилок та обмеженою продуктивністю [6].

У сучасних умовах конкурентоспроможність логістичних підприємств значною мірою визначається рівнем автоматизації виробничих і складських процесів. Використання автоматизованих систем дозволяє суттєво скоротити час виконання логістичних операцій, мінімізувати вплив людського фактора, підвищити точність сортування поштових відправлень та забезпечити стабільну роботу підприємства навіть за умов постійного зростання навантаження. Саме тому автоматизація логістичних процесів сьогодні розглядається як один із ключових напрямів розвитку сучасної транспортно-складської інфраструктури.

Типовий цикл обробки поштових відправлень включає приймання посилок, їх ідентифікацію, транспортування між технологічними зонами [7], сортування відповідно до визначених критеріїв, тимчасове зберігання та подальше відвантаження. На кожному із зазначених етапів дедалі ширше застосовуються автоматизовані технічні засоби, які забезпечують високу

продуктивність та безперервність технологічного процесу. Узагальнену структуру сучасного логістичного процесу наведено на рисунку 1.1.



Рисунок 1.1 – Основні етапи автоматизації логістичних процесів

Основу сучасних автоматизованих логістичних комплексів становлять конвеєрні системи, роботизовані маніпулятори, автоматизовані транспортні засоби (AGV – Automated Guided Vehicle), автономні мобільні роботи (AMR – Autonomous Mobile Robot), системи машинного зору, обладнання автоматичної ідентифікації вантажів, а також централізовані системи керування технологічними процесами. Взаємодія зазначених компонентів забезпечує виконання більшості операцій без безпосередньої участі людини, що позитивно впливає на продуктивність, безпеку та якість виконання логістичних процесів [7].

Особливу роль у розвитку сучасних логістичних комплексів відіграють мобільні роботизовані платформи. На відміну від традиційних транспортних систем із фіксованими маршрутами, мобільні роботи здатні автономно переміщуватися виробничими та складськими приміщеннями, змінювати

маршрути відповідно до поточної ситуації та взаємодіяти з іншими елементами автоматизованої системи. Використання таких платформ дозволяє підвищити гнучкість виробництва, оптимізувати використання складських площ та значно скоротити час транспортування вантажів між окремими технологічними зонами.

Для узагальнення основних напрямів розвитку сучасної автоматизації логістичних процесів сформовано таблицю 1.1 [7].

Таблиця 1.1 – Основні напрями автоматизації логістичних процесів

Напрям автоматизації	Основне призначення	Основні переваги
Конвеєрні системи	Автоматичне транспортування вантажів	Висока продуктивність, безперервність роботи
Роботизовані маніпулятори	Сортування та переміщення відправлень	Висока точність, повторюваність операцій
AGV	Перевезення вантажів за визначеними маршрутами	Зменшення використання ручної праці
AMR	Автономне транспортування із побудовою маршруту	Висока гнучкість та адаптивність
Машинний зір	Ідентифікація та контроль поштових відправлень	Висока швидкість аналізу та мінімізація помилок
IoT та системи моніторингу	Контроль стану обладнання та логістичних процесів	Централізоване керування та збір даних

Аналіз сучасних тенденцій розвитку логістичної галузі свідчить, що найбільш перспективним напрямом є побудова інтелектуальних програмно-апаратних комплексів, які поєднують мобільні роботизовані платформи, автоматизовані системи керування, комп’ютерний зір, технології Інтернету

речей та засоби централізованого моніторингу. Використання таких рішень забезпечує високу масштабованість систем, можливість їх модернізації, ефективний розподіл завдань між окремими модулями та адаптацію до змін умов експлуатації [7].

Таким чином, сучасний стан розвитку логістичних процесів характеризується активним впровадженням роботизованих та комп'ютерно-інтегрованих технологій, спрямованих на підвищення продуктивності, точності й ефективності обробки поштових відправлень. Аналіз існуючих тенденцій підтверджує актуальність розроблення нових програмно-апаратних комплексів, здатних забезпечити автоматизоване сортування поштових відправлень із використанням мобільних роботизованих платформ, сучасних систем керування та модульної архітектури. Це є підґрунтям для подальшого аналізу існуючих систем сортування поштових відправлень, який розглянуто у наступному підрозділі [8].

1.2 Аналіз існуючих систем сортування поштових відправлень

Постійне зростання обсягів поштових відправлень, розвиток електронної комерції та підвищення вимог до швидкості доставки обумовили необхідність широкого впровадження автоматизованих систем сортування у сучасних логістичних центрах. Якщо раніше процес сортування значною мірою виконувався вручну, то сьогодні більшість великих логістичних операторів використовує високопродуктивні програмно-апаратні комплекси, які забезпечують автоматичну ідентифікацію, транспортування, сортування та контроль руху поштових відправлень. Використання таких систем дозволяє значно підвищити продуктивність роботи підприємств, мінімізувати кількість помилок, скоротити час обробки посилок та забезпечити стабільність логістичних процесів навіть за умов нерівномірного навантаження [9].

Сучасні системи автоматизованого сортування являють собою складні

програмно-апаратні комплекси, до складу яких входять транспортні системи, роботизовані маніпулятори, мобільні роботизовані платформи, засоби автоматичної ідентифікації, системи комп'ютерного зору, програмне забезпечення керування та централізовані інформаційні системи[10].

Взаємодія зазначених компонентів забезпечує безперервне виконання всіх етапів логістичного процесу – від моменту надходження поштового відправлення до його передачі у відповідний напрямок доставки [10].

Залежно від принципу роботи сучасні системи сортування можна умовно поділити на кілька основних груп: конвеєрні системи, роботизовані сортувальні комплекси, системи із використанням автоматизованих транспортних засобів (AGV), автономних мобільних роботів (AMR), а також комбіновані комплекси, що поєднують декілька технологій одночасно. Узагальнену класифікацію сучасних систем сортування наведено на рисунку 1.2 [10].



Рисунок 1.2 – Класифікація сучасних систем автоматизованого сортування поштових відправлень

Найбільш поширеним рішенням залишаються конвеєрні системи [10] сортування, які характеризуються високою продуктивністю та здатністю безперервно транспортувати значні обсяги поштових відправлень. Такі системи

широко використовуються у великих логістичних центрах та сортувальних терміналах, де маршрути транспортування є заздалегідь визначеними та практично не змінюються. Основними перевагами конвеєрних систем є висока пропускна здатність, надійність роботи та відносно просте технічне обслуговування. Разом із тим вони характеризуються недостатньою гнучкістю, значною вартістю модернізації та потребують суттєвих змін інфраструктури у разі зміни технологічного процесу [11].

Одним із найбільш перспективних напрямів розвитку логістичних комплексів є використання мобільних роботизованих платформ [12]. На відміну від конвеєрних систем вони здатні автономно змінювати маршрути руху, взаємодіяти між собою та адаптуватися до змін виробничого середовища. Саме такий підхід використовується у сучасних логістичних центрах провідних світових компаній, де мобільні роботи виконують транспортування вантажів між окремими технологічними зонами, що дозволяє значно підвищити гнучкість виробництва та ефективніше використовувати виробничі площі.

Одним із найбільш відомих прикладів впровадження мобільних роботизованих систем є логістичні центри компанії Amazon Robotics [13], у яких реалізовано концепцію Goods-to-Person. Особливістю даного підходу є транспортування стелажів із товарами безпосередньо до робочого місця оператора або роботизованого комплексу. Завдяки цьому суттєво скорочується час пошуку необхідних товарів, зменшується кількість зайвих переміщень персоналу та підвищується загальна продуктивність складського комплексу. Разом із тим подібні системи вимагають спеціалізованої інфраструктури, централізованої системи керування та значних фінансових витрат на впровадження.

Ще одним прикладом високого рівня автоматизації є технологічна платформа Ocado Smart Platform [14], у якій використовується велика кількість автономних роботизованих платформ, що одночасно переміщуються спеціальною координатною сіткою над складськими модулями. Така архітектура

забезпечує дуже високу продуктивність сортування та комплектування замовлень, однак характеризується складністю побудови, високими вимогами до точності позиціонування роботів та використанням спеціалізованої інфраструктури.

Сучасні логістичні оператори, такі як DHL [15], активно впроваджують автономні мобільні роботи, системи машинного зору, технології Інтернету речей та цифрові системи керування складськими процесами.

Основною перевагою такого підходу є можливість поступової модернізації існуючих логістичних центрів без необхідності повної реконструкції виробничої інфраструктури. Аналогічні тенденції спостерігаються і в Україні, де великі оператори поштового зв'язку, зокрема Нова Пошта [9] та Укрпошта [16], поступово збільшують рівень автоматизації сортувальних центрів шляхом впровадження сучасних конвеєрних систем, автоматичного сканування поштових відправлень та централізованого програмного керування логістичними процесами.

Проведений аналіз свідчить, що сучасні системи автоматизованого сортування досягли високого рівня технологічного розвитку та забезпечують ефективну обробку великих потоків поштових відправлень. Разом із тим більшість існуючих рішень орієнтована на використання у великих логістичних центрах, характеризується значною вартістю впровадження, високою складністю інтеграції та залежністю від спеціалізованої інфраструктури. Крім того, багато сучасних комплексів мають обмежені можливості швидкої модернізації та адаптації до нових технологічних вимог.

Таким чином, проведений аналіз підтверджує доцільність розроблення програмно-апаратного комплексу модульної побудови, який поєднуватиме сучасні засоби автоматизації, мобільну роботизовану платформу, розподілену систему керування та відкриту архітектуру взаємодії апаратних модулів. Такий підхід дозволить забезпечити можливість подальшого розвитку комплексу, його адаптацію до різних умов експлуатації та інтеграцію нових технічних рішень без

суттєвої зміни загальної структури системи.

1.3 Аналіз мобільних роботизованих платформ для автоматизованого сортування поштових відправлень

Сучасні логістичні комплекси дедалі активніше використовують мобільні роботизовані платформи як один із ключових елементів автоматизації процесів транспортування та сортування поштових відправлень.

На відміну від стаціонарних конвеєрних систем такі платформи забезпечують гнучке переміщення вантажів між окремими технологічними зонами, дозволяють оперативно змінювати маршрути руху відповідно до виробничої ситуації та легко інтегруються до складу сучасних програмно-апаратних комплексів. Використання мобільних роботів [17] сприяє підвищенню продуктивності логістичних центрів, ефективнішому використанню виробничих площ і скороченню часу виконання транспортних операцій.

Конструкція мобільної роботизованої платформи визначається умовами її експлуатації, характером технологічних операцій та вимогами до точності позиціонування [18]. У сучасній практиці найбільшого поширення набули платформи з диференціальним приводом, керуванням за принципом Акермана (Ackermann [19]), колесами типу Omni [20], колесами Mecanum [21], а також спеціалізовані конструкції для роботи у складних умовах експлуатації. Кожен із зазначених типів має власні конструктивні особливості, переваги та обмеження, що визначають доцільність його використання у конкретній галузі.

Для узагальнення основних типів мобільних роботизованих платформ [22] їх класифікацію наведено на рисунку 1.3.

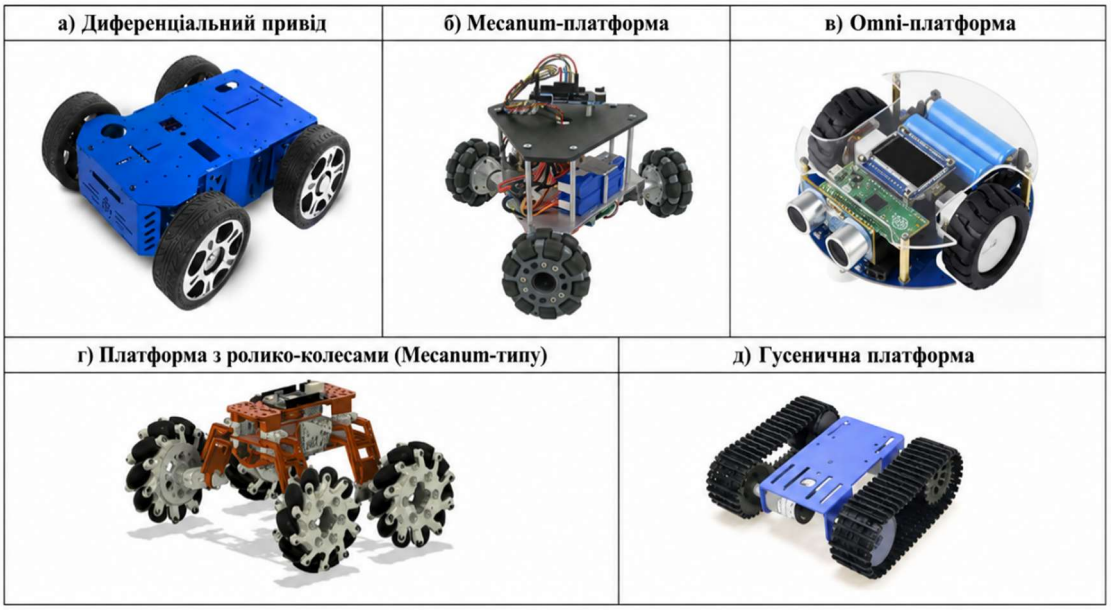


Рисунок 1.3 – Класифікація мобільних роботизованих платформ [22]

Найбільш поширеним рішенням у логістичних комплексах є платформи з диференціальним приводом. Така конструкція передбачає використання двох незалежно керованих ведучих коліс та одного або декількох опорних коліс. Зміна швидкості обертання ведучих коліс забезпечує рух платформи вперед, назад, а також її поворот навколо власної осі. Основними перевагами диференціального приводу є простота конструкції [23], висока надійність, невелика кількість механічних вузлів, низька вартість реалізації та достатня маневреність для роботи у складських приміщеннях. Саме тому даний тип приводу найчастіше використовується в автономних транспортних роботах логістичних центрів.

Платформи з керуванням за принципом Акермана [19] широко застосовуються у транспортних засобах та мобільних роботах, що працюють на великих відкритих територіях. Такі системи забезпечують плавний рух на високих швидкостях та характеризуються доброю курсовою стійкістю. Водночас значний радіус повороту обмежує можливість їх ефективного використання у вузьких проходах складських приміщень, тому у внутрішньоскладській логістиці вони використовуються значно рідше.

Колеса типу Omni [20] та Mecanum [21] забезпечують платформі

можливість всенапрямного руху без зміни орієнтації корпусу. Завдяки спеціальній конструкції роликів такі платформи можуть переміщуватися вперед, назад, убік або по діагоналі, що підвищує маневреність. Разом із тим такі системи є конструктивно складнішими, потребують точнішої синхронізації приводів, більш чутливі до якості покриття підлоги та мають нижчу енергоефективність порівняно з диференціальним приводом.

Окрему групу становлять спеціалізовані мобільні платформи для роботи на нерівній поверхні або складному рельєфі [22]. До них належать гусеничні та багатоколісні конструкції, які забезпечують підвищену прохідність і стійкість. Проте в автоматизованих сортувальних комплексах такі рішення застосовуються рідко, оскільки логістичні центри переважно мають рівне покриття та не потребують підвищеної прохідності.

Для порівняння основних характеристик сучасних мобільних платформ сформовано таблицю 1.3.

Таблиця 1.3 – Порівняльна характеристика мобільних роботизованих платформ [22]

Тип платформи	Маневреність	Конструктивна складність	Вартість реалізації	Основна сфера застосування
Диференціальний привід	Висока	Низька	Низька	Логістичні комплекси, AGV, AMR
Ackermann	Середня	Середня	Середня	Транспортні роботи, відкриті території
Omni	Дуже висока	Висока	Висока	Виробничі комплекси, лабораторні роботи
Mecanum	Дуже висока	Висока	Висока	Роботизовані платформи спеціального призначення
Гусенична	Низька в приміщеннях	Середня	Висока	Позашляхові роботи, спеціальна техніка

Проведений аналіз свідчить, що для задач автоматизованого сортування поштових відправлень найбільш доцільним є використання мобільної роботизованої платформи з диференціальним приводом. Таке рішення забезпечує оптимальне співвідношення між конструктивною простотою, маневреністю, надійністю та економічною ефективністю. Крім того, даний тип платформи добре поєднується із сучасними засобами навігації, системами комп'ютерного зору, лазерними далекомірами та іншими компонентами програмно-апаратних комплексів [17, 23].

Таким чином, аналіз сучасних мобільних роботизованих платформ показав, що різні типи конструкцій мають власні переваги та обмеження залежно від умов експлуатації. Для автоматизованого сортування поштових відправлень найбільш раціональним є застосування платформи з диференціальним приводом, яка забезпечує необхідну маневреність, простоту реалізації та можливість подальшої інтеграції із сучасними системами керування. Отримані результати аналізу є підґрунтям для вибору елементної бази та проектування апаратного забезпечення програмно-апаратного комплексу, що розглядається у наступному підрозділі.

1.4 Аналіз сучасної елементної бази мобільних роботизованих платформ

Ефективність сучасних мобільних роботизованих платформ значною мірою визначається правильним вибором елементної бази, яка забезпечує керування виконавчими механізмами, обробку інформації від датчиків, взаємодію між окремими модулями системи та виконання алгоритмів навігації. Розвиток мікроелектроніки, засобів обчислювальної техніки та цифрових систем керування сприяв появі великої кількості компонентів, що можуть використовуватися при побудові програмно-апаратних комплексів різного рівня складності. Вибір конкретних технічних рішень залежить від вимог до продуктивності системи, швидкодії, енергоефективності, масштабованості,

вартості та можливості подальшої модернізації [23].

Основу більшості сучасних мобільних роботизованих платформ становлять мікроконтролери, які виконують функції керування виконавчими механізмами, обробки сигналів від датчиків та реалізації алгоритмів реального часу. Найбільшого поширення набули мікроконтролери сімейств STM32, ESP32, AVR та PIC [24 - 27]. Серед них особливе місце займають STM32, побудовані на базі ядер ARM Cortex-M, оскільки вони характеризуються високою продуктивністю, розвиненою периферією, підтримкою сучасних інтерфейсів обміну даними та можливістю масштабування у промисловій автоматизації, робототехніці й комп’ютерно-інтегрованих системах керування.

Для виконання задач високорівневої обробки інформації дедалі частіше використовуються одноплатні комп'ютери, які забезпечують значно більшу обчислювальну потужність порівняно з мікроконтролерами. До найбільш поширених представників цього класу належать Raspberry Pi [6, 28-29], NVIDIA Jetson [30], Orange Pi [31] та інші подібні пристрої. Їх використання дозволяє реалізовувати алгоритми машинного зору, обробки зображень, штучного інтелекту, побудови карт навколишнього середовища та централізованого керування роботизованими комплексами. Практика побудови сучасних мобільних роботів свідчить, що найбільш ефективним є поєднання мікроконтролера, який виконує задачі реального часу, з одноплатним комп'ютером [6], який відповідає за високорівневу логіку роботи системи.

Не менш важливим елементом мобільних роботизованих платформ є сенсорна система. Для забезпечення автономної роботи використовуються енкодери, інерціальні вимірювальні модулі (IMU), ультразвукові та лазерні далекоміри, ToF-сенсори, камери технічного зору, датчики наближення та інші засоби збору інформації [22]. Комплексне використання декількох типів датчиків дозволяє підвищити точність визначення положення платформи, забезпечити виявлення перешкод та реалізувати алгоритми автономної навігації. Саме тому сучасні роботизовані комплекси дедалі частіше використовують

принцип мультисенсорної інтеграції, за якого інформація від різних джерел обробляється спільно для підвищення надійності роботи системи.

Для виконання задач високорівневої обробки інформації дедалі частіше використовуються одноплатні комп'ютери, які забезпечують значно більшу обчислювальну потужність порівняно з мікроконтролерами. До найбільш поширених представників цього класу належать Raspberry Pi [28-29], NVIDIA Jetson [30], Orange Pi [31] та інші подібні пристрої. Їх використання дозволяє реалізовувати алгоритми машинного зору, обробки зображень, штучного інтелекту, побудови карт навколишнього середовища та централізованого керування роботизованими платформами.

Не менш важливим елементом мобільних роботизованих платформ є сенсорна система. Для забезпечення автономної роботи використовуються енкодери, інерціальні вимірювальні модулі (IMU), ультразвукові та лазерні далекоміри, ToF-сенсори, камери технічного зору, датчики наближення та інші засоби збору інформації [22]. Комплексне використання декількох типів датчиків дозволяє підвищити точність визначення положення платформи, забезпечити виявлення перешкод та реалізувати алгоритми автономної навігації. Саме тому сучасні роботизовані комплекси дедалі частіше використовують принцип мультисенсорної інтеграції, за якого інформація від різних джерел обробляється спільно для підвищення надійності роботи системи.

Для передачі інформації між окремими модулями програмно-апаратного комплексу застосовуються різноманітні цифрові інтерфейси. Найбільш поширеними є UART, SPI, I²C, USB, Ethernet, CAN та RS-485 [32]. Кожен із зазначених інтерфейсів має власні особливості використання. Інтерфейси UART та I²C [32] широко застосовуються для підключення датчиків і допоміжних модулів, SPI забезпечує високошвидкісний обмін інформацією між електронними компонентами, тоді як CAN та Ethernet [32] використовуються у складних розподілених системах керування, де необхідно забезпечити надійну передачу даних між великою кількістю пристроїв.

Важливу роль у роботі мобільної роботизованої платформи відіграє система електроживлення, яка повинна забезпечувати стабільну роботу всіх електронних модулів незалежно від режиму навантаження. До складу сучасних систем живлення входять акумуляторні батареї, модулі перетворення напруги, системи захисту та контролю заряду. При проектуванні програмно-апаратних комплексів особлива увага приділяється енергоефективності, електромагнітній сумісності та забезпеченню безпечної роботи всіх електронних компонентів [33].

Для узагальнення основних характеристик сучасної елементної бази мобільних роботизованих платформ сформовано таблицю 1.4.

Таблиця 1.4 – Порівняльна характеристика основних компонентів програмно-апаратного комплексу [33]

Компонент	Основне призначення	Основні переваги	Особливості застосування
Мікроконтролер	Керування виконавчими механізмами	Висока швидкодія в режимі реального часу	Реалізація алгоритмів керування
Одноплатний комп’ютер	Високорівнева обробка даних	Велика обчислювальна потужність	Машинний зір, штучний інтелект
Датчики	Збір інформації про стан системи та середовище	Забезпечення автономної роботи	Навігація, позиціонування, безпекаф
Драйвери двигунів	Керування електроприводами	Точне регулювання параметрів руху	Робота з двигунами постійного струму, BLDC, кроковими двигунами
Інтерфейси зв’язку	Обмін інформацією між модулями	Надійність та швидкість передачі даних	Інтеграція компонентів комплексу
Система живлення	Забезпечення електроживлення	Стабільність та енергоефективність	Автономна робота мобільної платформи

Проведений аналіз сучасної елементної бази свідчить, що побудова ефективних мобільних роботизованих платформ потребує комплексного використання мікроконтролерів, високопродуктивних обчислювальних модулів, сучасних сенсорних систем та надійних інтерфейсів обміну даними. При цьому кожен компонент виконує власну функцію, а їх взаємодія визначає загальну ефективність роботи програмно-апаратного комплексу [22].

Таким чином, сучасний розвиток електронної компонентної бази дозволяє створювати високопродуктивні роботизовані системи з модульною архітектурою, що забезпечують високу швидкодію, масштабованість та можливість інтеграції нових функціональних модулів.

Отримані результати аналізу підтверджують доцільність використання комбінованої архітектури, у якій мікроконтролер забезпечує виконання задач реального часу, а одноплатний комп'ютер реалізує функції високорівневого керування, обробки інформації та взаємодії із зовнішніми інформаційними системами. Такий підхід є основою для формування технічних вимог до програмно-апаратного комплексу, що розробляється, які будуть сформульовані у наступному підрозділі.

1.5 Постановка задачі бакалаврской роботи

Проведений аналіз сучасного стану автоматизації логістичних процесів, існуючих систем сортування поштових відправлень, мобільних роботизованих платформ та сучасної елементної бази показав, що розвиток програмно-апаратних комплексів є одним із найбільш перспективних напрямів удосконалення логістичної інфраструктури. Використання роботизованих систем дозволяє значно підвищити продуктивність процесів сортування, скоротити час обробки поштових відправлень, мінімізувати вплив людського фактора та забезпечити високий рівень масштабованості логістичних комплексів.

Разом із тим проведений аналіз засвідчив, що більшість існуючих технічних рішень характеризуються значною складністю реалізації, високою вартістю впровадження, залежністю від спеціалізованої інфраструктури та обмеженими можливостями модернізації. Це обумовлює необхідність розроблення програмно-апаратного комплексу модульної побудови, який забезпечуватиме можливість інтеграції сучасних засобів автоматизації, гнучке керування окремими підсистемами та подальше розширення функціональних можливостей.

Результатом виконання бакалаврскої роботи має стати програмно-апаратний комплекс модульної архітектури, що забезпечує можливість автоматизованого сортування поштових відправлень із використанням мобільної роботизованої платформи та сучасної електронної компонентної бази. Запропоновані технічні рішення повинні забезпечувати можливість подальшого розвитку комплексу, інтеграції додаткових функціональних модулів і адаптації до різних умов експлуатації.

2 РОЗРОБЛЕННЯ АПАРАТНОГО ЗАБЕЗПЕЧЕННЯ ПРОГРАМНО-АПАРАТНОГО КОМПЛЕКСУ ДЛЯ СОРТУВАННЯ ПОШТОВИХ ВІДПРАВЛЕНЬ

2.1 Загальна архітектура програмно-апаратного комплексу

Проведений у першому розділі аналіз сучасних логістичних систем, мобільних роботизованих платформ та елементної бази показав, що ефективність автоматизованого сортування поштових відправлень значною мірою визначається правильно побудованою архітектурою програмно-апаратного комплексу. Саме структура взаємодії між обчислювальними модулями, засобами керування, сенсорною підсистемою та виконавчими механізмами визначає швидкодію системи, її масштабованість, надійність і можливість подальшої модернізації. У зв’язку з цим одним із ключових етапів розроблення є формування загальної архітектури комплексу, яка повинна забезпечувати ефективний розподіл функцій між окремими апаратними модулями та підтримувати їх узгоджену роботу [22].

Під час проектування програмно-апаратного комплексу були сформовані основні технічні вимоги, яким повинна відповідати система. До них належать модульна побудова, можливість масштабування, забезпечення роботи в режимі реального часу, підтримка сучасних інтерфейсів обміну даними, висока надійність, енергоефективність та можливість інтеграції додаткових функціональних модулів без суттєвої зміни загальної структури комплексу. Виконання зазначених вимог дозволяє використовувати систему як основу для створення сучасних автоматизованих комплексів сортування різної продуктивності та конфігурації.

Зазначені технічні вимоги є взаємопов’язаними: модульність забезпечує можливість подальшого розширення комплексу, масштабованість визначає його

роботу з різною кількістю апаратних модулів, а режим реального часу є необхідним для керування виконавчими механізмами. Надійність, відмовостійкість та енергоефективність визначають стабільність роботи мобільної роботизованої платформи під час тривалої експлуатації. Для узагальнення сформованих вимог до побудови програмно-апаратного комплексу та визначення їх функціонального призначення основні технічні вимоги систематизовано в таблиці 2.1.

Таблиця 2.1 – Основні технічні вимоги до програмно-апаратного комплексу

Вимога	Необхідність реалізації
Модульність	Незалежна розробка та модернізація окремих вузлів
Масштабованість	Можливість підключення нових модулів
Робота в режимі реального часу	Керування виконавчими механізмами
Висока продуктивність	Обробка інформації від великої кількості датчиків
Надійність	Безперервна робота комплексу
Відмовостійкість	Локалізація несправностей окремих модулів
Енергоефективність	Автономна робота системи

На підставі сформованих технічних вимог було визначено структуру програмно-апаратного комплексу та обґрунтовано необхідність використання багаторівневої модульної архітектури.

З урахуванням поставлених вимог у роботі запропоновано архітектуру комплексу, яка передбачає розподіл функцій між окремими рівнями керування. Верхній рівень призначений для виконання задач високорівневої обробки інформації, прийняття рішень та взаємодії з інформаційними системами. Середній рівень забезпечує координацію роботи апаратних модулів, обмін даними та формування команд керування. Нижній рівень відповідає за

виконання задач реального часу, зокрема опитування датчиків, керування виконавчими механізмами та контроль стану окремих вузлів комплексу [34].

Такий підхід дозволяє підвищити надійність роботи системи, спростити її подальшу модернізацію та забезпечити можливість підключення додаткових функціональних модулів без суттєвої зміни загальної структури комплексу. Загальну структуру програмно-апаратного комплексу наведено на рисунку 2.1 [25-29].

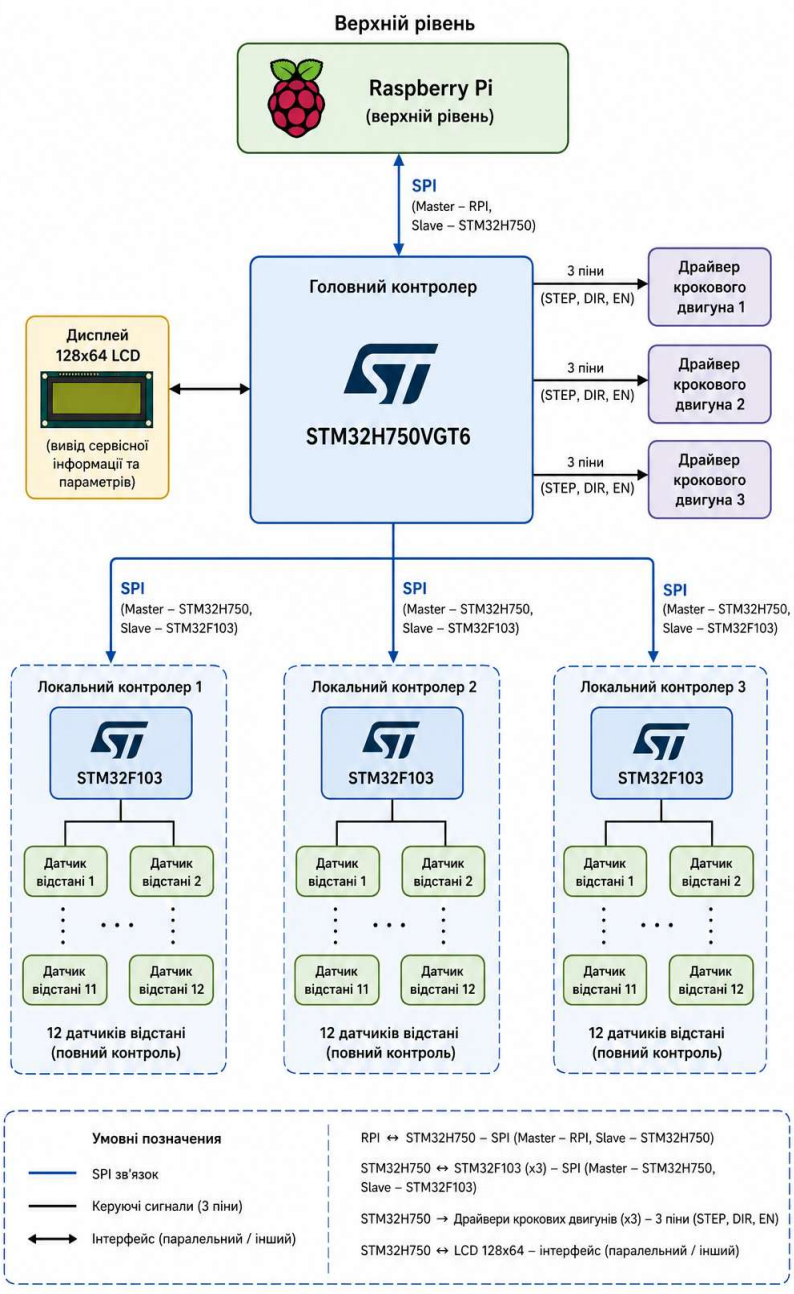


Рисунок 2.1 – Загальна структурна схема програмно-апаратного комплексу для сортування поштових відправлень

Аналіз сформованих вимог показує, що реалізація системи на базі одного обчислювального модуля є недоцільною, тому для забезпечення необхідної продуктивності запропоновано використання багаторівневої архітектури [22]. Функціональне призначення основних модулів наведено в таблиці 2.2.

Таблиця 2.2 – Функціональне призначення основних модулів комплексу

Модуль	Основні функції	Інтерфейс
Raspberry Pi 5	Виконання високорівневих алгоритмів керування та взаємодія з користувачем	SPI, Ethernet, USB
STM32H750VGT6	Центральне керування комплексом, координація роботи модулів	SPI, GPIO
STM32F103	Локальне опитування датчиків та попередня обробка інформації	SPI, GPIO
Драйвери двигунів	Керування кроковими двигунами	STEP, DIR, EN
LCD-дисплей	Відображення службової інформації	GPIO (паралельний інтерфейс)
Датчики відстані	Виявлення перешкод та контроль положення	GPIO

Верхній рівень архітектури представлений одноплатним комп'ютером Raspberry Pi 5, який виконує функції високорівневого керування комплексом. До його основних задач належать обробка інформації від системи технічного зору, виконання алгоритмів сортування, формування маршрутів руху мобільної платформи, взаємодія із серверною частиною системи та забезпечення користувацького інтерфейсу. Завдяки високій обчислювальній продуктивності Raspberry Pi [28-29] забезпечує виконання складних алгоритмів, які недоцільно реалізовувати безпосередньо на мікроконтролерах.

Центральним елементом системи керування є високопродуктивний мікроконтролер STM32H750VGT6 [25-27], який реалізує виконання задач реального часу. На нього покладаються функції керування приводами виконавчих механізмів, збір інформації від локальних контролерів, координація

роботи окремих модулів комплексу, формування керуючих сигналів для драйверів крокових двигунів, а також реалізація швидкісного обміну даними з верхнім рівнем системи. Вибір мікроконтролера сімейства STM32 [25-27] обумовлений високою продуктивністю ядра ARM Cortex-M7 [28], розвиненою периферією, підтримкою сучасних інтерфейсів зв'язку та широкими можливостями масштабування.

Для підвищення ефективності роботи сенсорної підсистеми використано розподілену структуру локальних контролерів на базі STM32F103 [25-27]. Кожний локальний контролер здійснює опитування групи датчиків відстані, виконує первинну обробку отриманої інформації та передає результати головному контролеру через цифровий інтерфейс SPI [32]. Такий підхід дозволяє рівномірно розподілити обчислювальне навантаження між окремими вузлами системи, скоротити час реакції на зміну стану датчиків та забезпечити можливість подальшого розширення сенсорної мережі без зміни принципів функціонування комплексу.

Керування виконавчими механізмами [22] реалізовано за допомогою окремих драйверів крокових двигунів, які отримують сигнали STEP, DIR та EN від головного мікроконтролера. Подібне рішення дозволяє ізолювати силову частину від логічних кіл керування, підвищити надійність роботи системи та забезпечити незалежне керування кожним виконавчим механізмом. Така архітектура є типовою для сучасних автоматизованих систем позиціонування та робототехнічних комплексів.

Для забезпечення стабільної роботи електронних модулів передбачено окрему підсистему електроживлення. До її складу входять акумуляторна батарея, система керування батареєю (BMS) [33], DC/DC-перетворювачі різних рівнів напруги та модуль розподілу живлення. Така структура дозволяє забезпечити необхідні рівні напруги для різних груп споживачів, реалізувати захист від перевантаження й короткого замикання, а також підвищити енергоефективність програмно-апаратного комплексу.

Обмін інформацією між модулями комплексу здійснюється за допомогою цифрових інтерфейсів. Для взаємодії Raspberry Pi з головним мікроконтролером передбачено використання SPI, UART або USB [32] залежно від режиму роботи системи. Зв’язок головного контролера з локальними модулями реалізовано через SPI, що забезпечує високу швидкість обміну, малі затримки та можливість одночасної роботи з кількома периферійними пристроями.

Важливою особливістю запропонованої архітектури є можливість її подальшого розвитку без зміни базових принципів побудови. За необхідності до комплексу можуть бути інтегровані додаткові сенсорні модулі, нові виконавчі механізми або інші функціональні вузли без зміни структури центральної системи керування. Це забезпечує універсальність запропонованого рішення та можливість його адаптації до різних виробничих умов.

Запропонована архітектура забезпечує чітке функціональне розмежування між рівнями системи, дозволяє незалежно модернізувати окремі модулі без зміни загальної структури комплексу та створює основу для подальшого розширення функціональних можливостей. Модульний принцип побудови також спрощує процес технічного обслуговування, локалізації несправностей та інтеграції нового обладнання, що є важливою перевагою під час експлуатації сучасних автоматизованих систем.

Таким чином, у результаті проектування було сформовано загальну архітектуру програмно-апаратного комплексу для сортування поштових відправлень, яка поєднує багаторівневу систему керування, розподілену сенсорну підсистему, сучасні засоби обчислювальної техніки та модульну структуру взаємодії окремих компонентів. Запропонована архітектура відповідає поставленим технічним вимогам і є основою для подальшого проектування окремих функціональних вузлів комплексу, що розглядаються у наступних підрозділах.

2.2 Розроблення структурної організації апаратного забезпечення програмно-апаратного комплексу

Після визначення загальної архітектури програмно-апаратного комплексу наступним етапом проектування є розроблення його структурної організації. На цьому етапі визначається склад апаратних модулів, їх функціональне призначення, принципи взаємодії між окремими вузлами, а також розподіл функцій керування між центральним контролером, локальними мікроконтролерами, сенсорною підсистемою та виконавчими механізмами [22].

При проектуванні апаратного забезпечення особливу увагу приділено забезпеченню модульності конструкції, можливості масштабування системи, простоті технічного обслуговування та високій швидкодії під час виконання задач реального часу. Для цього запропоновано розподілену структуру керування, у якій окремі апаратні модулі виконують спеціалізовані функції, взаємодіючи між собою за допомогою високошвидкісних цифрових інтерфейсів.

Такий підхід дозволяє зменшити навантаження на центральний контролер, скоротити час обробки інформації від сенсорної системи, забезпечити незалежне керування окремими виконавчими механізмами та підвищити загальну надійність роботи програмно-апаратного комплексу. Крім того, модульна структура значно спрощує подальшу модернізацію системи шляхом підключення нових функціональних вузлів без зміни принципів побудови всієї архітектури.

У даному підрозділі розглянуто структурну організацію апаратного забезпечення програмно-апаратного комплексу, виконано проектування основних апаратних модулів, визначено їх функціональне призначення та принципи взаємодії між окремими компонентами системи.

2.2.1 Розроблення центрального модуля керування

Після формування загальної архітектури наступним етапом є розроблення центрального модуля керування, який координує роботу апаратних компонентів системи. Центральний контролер обробляє інформацію від локальних модулів, формує керуючі сигнали для виконавчих механізмів та забезпечує інформаційну взаємодію між рівнями архітектури.

При виборі центрального контролера враховано необхідність роботи в режимі реального часу, обслуговування периферійних пристроїв, підтримки високошвидкісних інтерфейсів та масштабування комплексу. Центральний модуль повинен мати достатній запас продуктивності для реалізації алгоритмів керування й стабільної взаємодії з локальними контролерами.

Для реалізації зазначених вимог у роботі обрано високопродуктивний мікроконтролер STM32H750VGT6 [25-27] виробництва компанії STMicroelectronics [25-27]. Даний мікроконтролер побудований на базі ядра ARM Cortex-M7 [28] та поєднує високу обчислювальну продуктивність із широкими можливостями периферії, що робить його придатним для використання у сучасних робототехнічних та комп'ютерно-інтегрованих системах. Значний обсяг оперативної пам'яті, наявність декількох цифрових інтерфейсів SPI, UART, I²C, USB [23] , а також розвинена система таймерів дозволяють одночасно виконувати задачі керування виконавчими механізмами, обробки інформації від сенсорної підсистеми та взаємодії з верхнім рівнем керування. Як видно з таблиці 2.3, мікроконтролер STM32H750VGT6 [25-27] має достатній запас обчислювальної продуктивності та периферійних ресурсів для реалізації функцій центрального модуля керування програмно-апаратного комплексу.

У запропонованому програмно-апаратному комплексі центральний контролер виконує функції координатора всієї системи. Через інтерфейс SPI [32] він здійснює інформаційний обмін з одноплатним комп’ютером Raspberry Pi [28-29], який виконує високорівневі алгоритми керування.

Таблиця 2.3 – Основні характеристики мікроконтролера STM32H750VGT6

Параметр	Значення
Ядро	ARM Cortex-M7
Тактова частота	до 480 МГц
Flash-пам’ять	1 МБ
SRAM	1 МБ
SPI	до 6
UART	4
I²C	4
Таймери	20+
GPIO	понад 100

Також центральний контролер підтримує зв’язок із трьома локальними мікроконтролерами STM32F103 [25-27], що відповідають за збір та попередню обробку інформації від датчиків відстані. Такий розподіл функцій дозволяє відокремити задачі реального часу від ресурсоємних алгоритмів аналізу даних, що позитивно впливає на швидкодію та надійність комплексу.

Окремим важливим завданням центрального модуля є формування сигналів керування кроковими двигунами мобільної платформи. Для цього використовуються спеціалізовані виходи мікроконтролера, які генерують сигнали STEP, DIR та EN, що надходять до драйверів двигунів. Використання апаратних таймерів дозволяє формувати імпульсні послідовності з високою точністю, забезпечувати плавний розгін і гальмування двигунів, а також реалізовувати синхронне керування декількома виконавчими механізмами без додаткового навантаження на центральне процесорне ядро.

Для взаємодії з оператором до центрального контролера підключено графічний LCD-дисплей, який використовується для відображення службової інформації про стан комплексу, результати самодіагностики, параметри роботи та повідомлення про можливі аварійні ситуації. Наявність локального інтерфейсу відображення інформації значно спрощує процес налагодження,

технічного обслуговування та експлуатації комплексу без необхідності підключення зовнішнього комп'ютера.

Крім відображення поточного стану комплексу, локальний дисплей може використовуватися під час первинного налаштування та технічної перевірки центрального модуля. Через нього доцільно виводити інформацію про активність інтерфейсів зв’язку, стан підключених локальних контролерів, наявність помилок обміну даними та готовність виконавчих механізмів до роботи. Це дозволяє швидше виявляти несправності окремих вузлів і спрощує перевірку працездатності системи без використання додаткового діагностичного обладнання.

Для наочного подання складу периферійних модулів та їх взаємодії із зовнішніми пристроями структурну організацію центрального модуля наведено на рисунку 2.2.

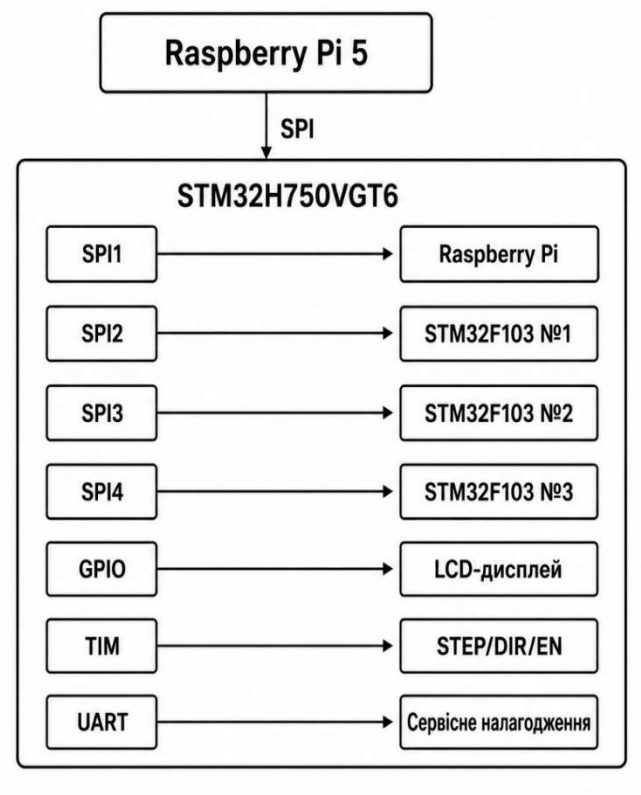


Рисунок 2.2 – Структурна організація центрального модуля керування

Як видно з рисунка 2.2, центральний контролер взаємодіє з усіма основними функціональними модулями комплексу, забезпечуючи обмін інформацією через інтерфейси SPI [32], керування виконавчими механізмами за допомогою таймерів та взаємодію з оператором через LCD-дисплей [22].

Центральний модуль у цій структурі виконує роль проміжної ланки між високорівневим обчислювальним модулем, локальними мікроконтролерами, виконавчими механізмами та засобами відображення інформації. Така організація дозволяє розподілити функції між окремими апаратними вузлами та зменшити навантаження на кожен із них.

Основні функції центрального модуля керування пов’язані з координацією роботи системи, обміном даними, формуванням керуючих сигналів для приводів, обробкою службової інформації та передаванням даних на дисплей оператора. Для узагальнення призначення цих функцій сформовано таблицю 2.4.

Таблиця 2.4 – Основні функції центрального модуля керування

Функція	Призначення
Координація роботи системи	Синхронізація всіх апаратних модулів
Керування приводами	Формування сигналів STEP, DIR, EN
Обмін із Raspberry Pi	Передача команд і службових даних
Обмін із STM32F103	Приймання інформації від локальних контролерів
Відображення інформації	Керування LCD-дисплеєм
Сервісне налагодження	Передача діагностичних даних через UART

Наведений розподіл функцій підтверджує, що центральний модуль забезпечує координацію роботи всіх підсистем програмно-апаратного комплексу.

Таким чином, у результаті проектування розроблено структуру центрального модуля керування програмно-апаратного комплексу. Використання мікроконтролера STM32H750VGT6 [25-27] забезпечує виконання задач реального часу, координацію роботи локальних контролерів, формування

керуючих сигналів та високошвидкісний інформаційний обмін між усіма рівнями системи. Запропоноване технічне рішення відповідає сформованим вимогам та створює основу для подальшого проектування інших функціональних модулів комплексу.

2.2.2 Розроблення локальної сенсорної підсистеми

Після розроблення центрального контролера наступним етапом проектування є розроблення локальної сенсорної підсистеми, яка забезпечує збір інформації про навколишнє середовище та передачу її до центрального модуля керування. Ефективність роботи мобільної роботизованої платформи значною мірою залежить від швидкості отримання інформації про наявність перешкод, взаємне розташування об'єктів та поточний стан робочої зони. Саме тому сенсорна підсистема повинна забезпечувати високу швидкодію, надійність та можливість одночасної роботи великої кількості датчиків.

Під час проектування комплексу було прийнято рішення використовувати розподілену структуру збору даних, у якій функції опитування датчиків виконують окремі локальні контролери. Такий підхід дозволяє зменшити навантаження на центральний контролер, скоротити час реакції системи на зміну стану навколишнього середовища та забезпечити паралельну обробку інформації від декількох груп сенсорів.

Для реалізації локальної сенсорної підсистеми використано три мікроконтролери STM32F103, кожен із яких обслуговує окрему групу датчиків відстані. Локальні контролери виконують циклічне опитування підключених сенсорів, здійснюють первинну перевірку достовірності отриманих даних, їх попередню обробку та формують інформаційні пакети для передачі до центрального контролера STM32H750VGT6 через інтерфейс SPI.

Вибір мікроконтролера STM32F103 обумовлений достатньою обчислювальною продуктивністю для виконання задач локального опитування датчиків, підтримкою швидкісного інтерфейсу SPI, великою кількістю

універсальних входів-виходів GPIO [32], широкою доступністю програмних засобів розроблення та високою надійністю. Використання окремих локальних контролерів дозволяє винести задачі первинної обробки інформації із центрального модуля, що позитивно впливає на швидкодію всієї системи.

Розподіл функцій між декількома локальними контролерами дозволяє виконувати обробку інформації паралельно, що значно підвищує продуктивність комплексу порівняно із системою, у якій усі датчики безпосередньо підключені до центрального контролера. Крім того, така архітектура забезпечує можливість подальшого розширення сенсорної системи шляхом підключення додаткових локальних модулів без зміни принципів функціонування всієї системи.

Важливою перевагою використання локальних контролерів є також підвищення відмовостійкості комплексу. У разі виникнення несправності одного із сенсорних модулів інші локальні контролери продовжують виконувати свої функції, що дозволяє локалізувати відмову та зменшити її вплив на роботу програмно-апаратного комплексу в цілому.

Для взаємодії локальних контролерів із центральним модулем використовується інтерфейс SPI [32], який забезпечує високу швидкість передачі інформації, мінімальні затримки та просту реалізацію багатовузлової структури. Такий підхід відповідає вимогам роботи системи в режимі реального часу та забезпечує своєчасне надходження інформації до центрального контролера для подальшого прийняття керуючих рішень.

Для наочного відображення структури локальної сенсорної підсистеми її організацію наведено на рисунку 2.3.

Як видно з рисунка 2.3, локальна сенсорна підсистема має розподілену структуру, у якій кожний локальний контролер виконує незалежне опитування власної групи датчиків та передає результати обробки до центрального контролера. Така організація забезпечує паралельну обробку інформації, підвищує продуктивність системи та спрощує її масштабування.

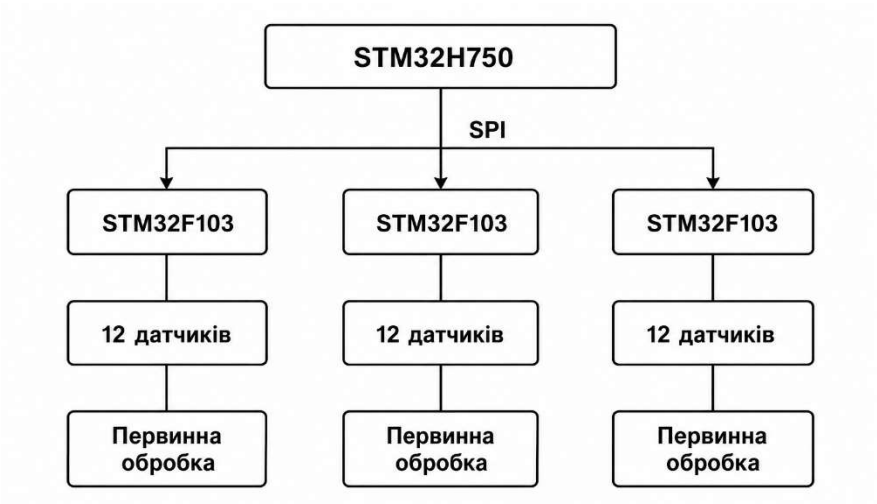


Рисунок 2.3 – Структурна організація локальної сенсорної підсистеми

Первинна обробка інформації включає перевірку коректності вимірювань, відкидання помилкових значень, усереднення результатів декількох вимірювань та формування інформаційного пакета для передачі центральному контролеру. Такий підхід дозволяє зменшити вплив випадкових похибок вимірювання та підвищити достовірність отриманих даних.

Розподіл датчиків відстані наведено у таблиці 2.5.

Таблиця 2.5 – Функціональне призначення локальних контролерів [26]

Контролер	Основні функції	Кількість датчиків
STM32F103 №1	Опитування першої групи датчиків	12
STM32F103 №2	Опитування другої групи датчиків	12
STM32F103 №3	Опитування третьої групи датчиків	12

Дані, наведені в таблиці 2.5, свідчать, що використання трьох локальних контролерів дозволяє рівномірно розподілити навантаження між окремими модулями сенсорної підсистеми та забезпечити незалежне опитування всіх датчиків відстані.

Кількість датчиків визначена необхідністю забезпечення безперервного контролю простору навколо мобільної роботизованої платформи. Розташування сенсорів дозволяє контролювати передню, задню та бокові зони, що забезпечує своєчасне виявлення перешкод під час руху та маневрування.

Основні характеристики мікроконтролера STM32F103 [26] наведено в таблиці 2.6.

Таблиця 2.6 – Основні характеристики STM32F103 [26]

Параметр	Значення
Ядро	ARM Cortex-M3
Частота	72 МГц
Flash	128 КБ
SRAM	20 КБ
SPI	2
UART	3
GPIO	до 80
Таймери	7

Таким чином, запропонована локальна сенсорна підсистема забезпечує розподілене опитування 36 датчиків відстані [22], їх попередню обробку та високошвидкісний обмін інформацією із центральним контролером. Використання трьох незалежних локальних модулів дозволяє реалізувати паралельну обробку інформації, підвищити продуктивність системи, забезпечити її масштабованість та локалізацію можливих відмов окремих вузлів.

2.2.3 Розроблення підсистеми керування виконавчими механізмами

Після проектування центрального модуля керування та локальної сенсорної підсистеми наступним етапом розроблення програмно-апаратного комплексу є створення підсистеми керування виконавчими механізмами. Саме ця підсистема забезпечує безпосереднє перетворення керуючих команд,

сформованих центральним контролером, у механічний рух мобільної роботизованої платформи та інших виконавчих вузлів комплексу.

Основними вимогами до підсистеми керування виконавчими механізмами є висока точність позиціонування, стабільність роботи, можливість незалежного керування декількома приводами одночасно, підтримка роботи в режимі реального часу та можливість подальшого масштабування системи. Крім того, система повинна забезпечувати простоту інтеграції із центральним контролером та відповідати вимогам сучасних робототехнічних комплексів.

Для реалізації зазначених вимог у роботі використано крокові електродвигуни, керування якими здійснюється за допомогою спеціалізованих драйверів. Подібне рішення широко використовується у промисловій автоматизації, верстатах із числовим програмним керуванням, 3D-принтерах та мобільних роботизованих системах завдяки високій точності позиціонування, простоті реалізації та можливості формування складних траєкторій руху без використання додаткових механічних передач.

Керування кожним двигуном здійснюється за допомогою трьох цифрових сигналів STEP, DIR та EN, які формуються центральним контролером STM32H750VGT6 [25]. Сигнал STEP визначає кількість виконаних кроків двигуна, DIR задає напрямок обертання ротора, а EN використовується для увімкнення або вимкнення драйвера. Формування зазначених сигналів виконується апаратними таймерами мікроконтролера, що забезпечує високу точність часових інтервалів і мінімізує навантаження на центральне процесорне ядро.

Використання окремих драйверів дозволяє відокремити силову частину системи від логічних кіл керування, забезпечити необхідний рівень струму для обмоток двигунів та реалізувати електричний захист центрального контролера. Такий підхід підвищує надійність роботи комплексу та спрощує заміну окремих компонентів у разі модернізації або технічного обслуговування.

Окремою перевагою такого підходу є можливість незалежного

налаштування параметрів кожного приводу відповідно до його функціонального призначення. Зокрема, для різних виконавчих механізмів можуть задаватися відмінні значення швидкості, прискорення, напрямку руху та режиму утримання положення. Це є важливим для роботизованого комплексу, у якому окремі механічні вузли можуть виконувати різні технологічні операції та працювати з різним навантаженням.

Для наочного відображення принципу взаємодії центрального контролера з драйверами та виконавчими механізмами структурну організацію підсистеми керування наведено на рисунку 2.4.

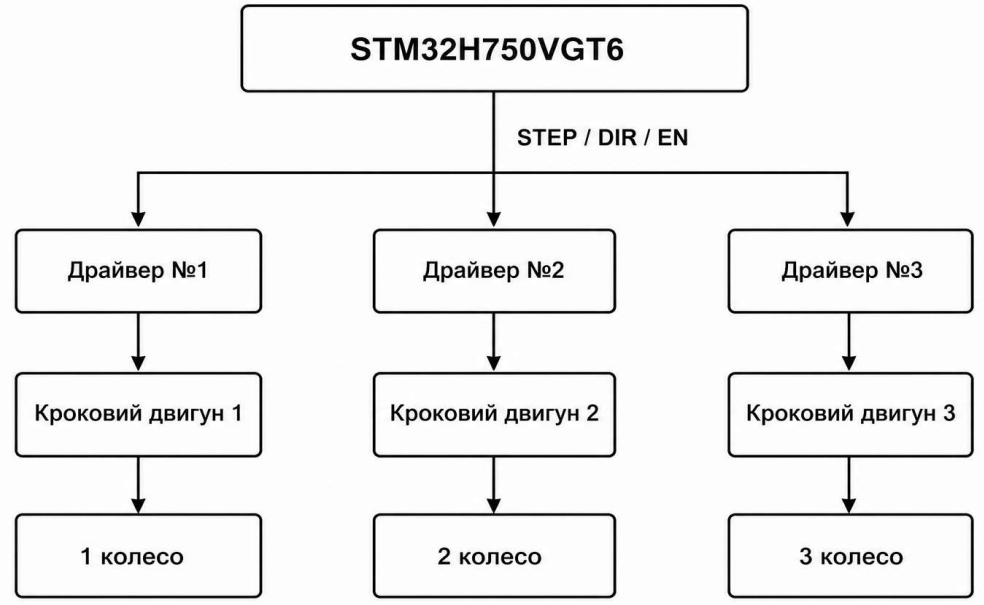


Рисунок 2.4 – Структурна організація підсистеми керування виконавчими механізмами

Як видно з рисунка 2.4, центральний контролер формує сигнали керування для кожного драйвера незалежно, що забезпечує синхронне або автономне керування окремими виконавчими механізмами. Використання окремих драйверів дозволяє ізолювати силову частину від логічної системи керування, підвищити надійність роботи комплексу та забезпечити масштабованість підсистеми. Призначення основних сигналів керування драйвером крокового двигуна наведено в таблиці 2.7.

Таблиця 2.7 – Основні сигнали керування драйвером крокового двигуна

Сигнал	Призначення
STEP	Формування імпульсів переміщення двигуна
DIR	Визначення напрямку обертання
EN	Увімкнення або вимкнення драйвера

Як видно з таблиці 2.7, для керування кожним виконавчим механізмом використовується лише три цифрові сигнали, що значно спрощує програмну реалізацію системи керування та забезпечує високу швидкість формування команд.

Для реалізації підсистеми керування виконавчими механізмами використано промислові драйвери крокових двигунів CW-5045 [35], які забезпечують стабільне керування двофазними кроковими двигунами із максимальним струмом до 4,5 А на фазу та підтримують режим мікрокрокового керування до 1/128 кроку. Використання промислових драйверів дозволяє забезпечити високу точність позиціонування, плавність руху виконавчих механізмів та значно знизити рівень механічних вібрацій. Основні технічні характеристики драйвера CW-5045 [35] наведено в таблиці 2.8.

Таблиця 2.8 – Основні технічні характеристики драйвера CW-5045 [35]

Параметр	Значення
Модель	CW-5045
Тип двигуна	Двофазний крковий
Напруга живлення	від 20 В до 50 В
Максимальний струм	4,5 А/фазу
Режими мікрокроку	До 1/128
Інтерфейс керування	STEP, DIR, EN
Захист	Від перегріву, перевантаження по струму, перенапруги

Як видно з таблиці 2.8, технічні характеристики драйвера CW-5045 [35] повністю відповідають вимогам програмно-апаратного комплексу та

забезпечують необхідний запас за струмом і точністю керування.

Використання режиму мікрокрокового керування дозволяє значно зменшити механічні коливання, характерні для повнокрокового режиму роботи крокових двигунів. Крім того, застосування мікрокроків забезпечує більш плавний рух мобільної платформи, підвищує точність позиціонування.

На відміну від компактних драйверів, що широко використовуються у навчальних стендах та 3D-принтерах, промисловий драйвер CW-5045 [35] має підвищений запас за струмом, кращу тепловіддачу та розрахований на тривалу безперервну експлуатацію. Це робить його більш придатним для використання у мобільних робототехнічних комплексах, де надійність роботи виконавчої системи має вирішальне значення.

2.2.4 Розроблення підсистеми електроживлення програмно-апаратного комплексу

Після проектування центрального модуля керування, локальної сенсорної підсистеми та підсистеми керування виконавчими механізмами наступним етапом розроблення програмно-апаратного комплексу є створення підсистеми електроживлення. Саме система електроживлення забезпечує безперервне функціонування всіх електронних модулів комплексу, стабільну роботу виконавчих механізмів та надійне функціонування цифрових інтерфейсів незалежно від режиму роботи мобільної роботизованої платформи.

Однією з основних вимог до підсистеми електроживлення є забезпечення стабільних рівнів напруги для різних груп споживачів, захист електронних компонентів від аварійних режимів роботи, можливість тривалої автономної експлуатації та мінімізація впливу силових кіл на цифрову електроніку. Крім того, система повинна забезпечувати можливість подальшого розширення комплексу без суттєвої зміни принципів побудови електроживлення.

Особливістю розроблюваного програмно-апаратного комплексу є наявність двох груп споживачів електроенергії, що істотно відрізняються за

характеристиками. До першої групи належать силові виконавчі механізми, представлені кроковими двигунами та драйверами CW-5045 [35], які споживають значний струм. До другої групи входять цифрові модулі керування: STM32H750VGT6, STM32F103 [26], Raspberry Pi 5 [28], графічний LCD-дисплей та допоміжні електронні компоненти, які потребують стабілізованого живлення з низьким рівнем електричних завад.

У зв'язку з цим під час проектування системи електроживлення було прийнято рішення виконати розділення силової та логічної частин комплексу. Такий підхід дозволяє мінімізувати вплив імпульсних струмів, що виникають під час роботи крокових двигунів, на цифрові модулі керування, підвищити електромагнітну сумісність системи та забезпечити стабільність роботи інформаційних інтерфейсів.

Джерелом живлення комплексу є акумуляторна батарея, яка забезпечує автономну роботу мобільної роботизованої платформи. Для захисту батареї використовується система керування акумулятором (BMS) [33], що виконує контроль напруги окремих елементів, струму заряду та розряду, температурного режиму, а також забезпечує захист від короткого замикання, перевантаження та глибокого розряду. Використання BMS [33] підвищує безпеку експлуатації комплексу та збільшує ресурс акумуляторної батареї.

Для формування необхідних рівнів напруги застосовуються імпульсні DC/DC-перетворювачі. Силова частина системи забезпечує живлення драйверів CW-5045 [35] безпосередньо від акумуляторної батареї, тоді як цифрові модулі отримують стабілізовану напругу 5 В або 3,3 В залежно від типу обладнання. Такий підхід дозволяє забезпечити високу ефективність перетворення енергії, мінімізувати теплові втрати та знизити енергоспоживання комплексу.

Для підвищення надійності роботи системи передбачено використання головного вимикача живлення, плавкого запобіжника та окремих ліній живлення для силових і логічних модулів. У разі виникнення аварійної ситуації зазначені елементи забезпечують локалізацію несправності та запобігають пошкодженню

дорогої електронної апаратури.

Окрему увагу під час розроблення підсистеми електроживлення приділено розділенню силових і логічних кіл. Живлення виконавчих механізмів та драйверів двигунів передбачено виконувати окремими лініями, оскільки під час роботи крокових двигунів можуть виникати імпульсні навантаження, просідання напруги та електромагнітні завади.

Логічні модулі керування, сенсорна підсистема та одноплатний комп’ютер повинні отримувати стабілізоване живлення через окремі DC/DC-перетворювачі, що дозволяє підвищити надійність роботи цифрової частини комплексу. Структурну організацію підсистеми електроживлення наведено на рисунку 2.5.

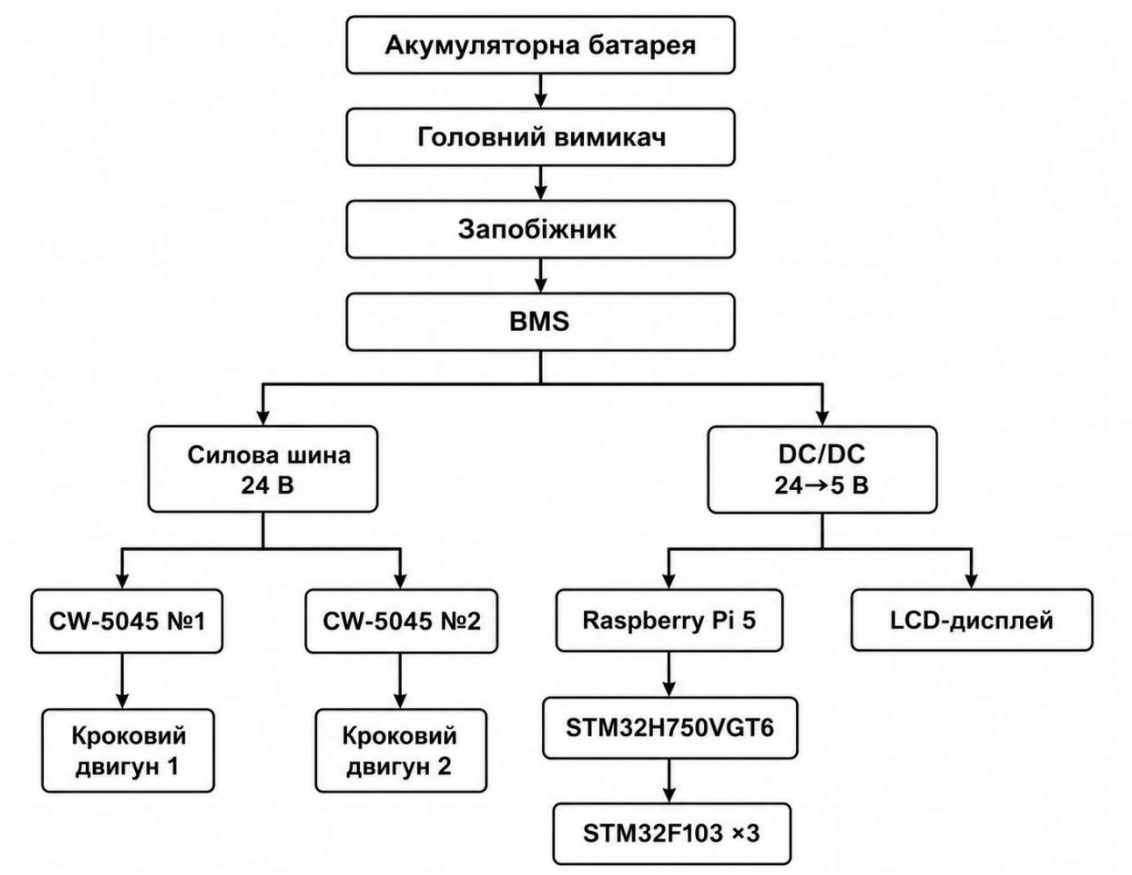


Рисунок 2.5 – Структурна схема підсистеми електроживлення програмно-апаратного комплексу

Як видно з рисунка 2.5, система електроживлення має багаторівневу структуру, у якій силові та цифрові кола живлення фізично розділено. Такий підхід дозволяє мінімізувати вплив високих імпульсних струмів виконавчих механізмів на цифрові модулі керування, підвищити електромагнітну сумісність системи та забезпечити стабільну роботу програмно-апаратного комплексу.

Основні елементи підсистеми електроживлення наведено в таблиці 2.9.

Таблиця 2.9 – Основні елементи підсистеми електроживлення

Елемент	Призначення
Акумуляторна батарея	Основне джерело живлення комплексу
BMS	Контроль та захист акумулятора
Головний вимикач	Повне відключення живлення
Запобіжник	Захист від короткого замикання
DC/DC-перетворювач	Формування стабілізованої напруги
Драйвери CW-5045	Живлення виконавчих механізмів
STM32H750VGT6	Центральний модуль керування
Raspberry Pi 5	Верхній рівень керування

Дані таблиці 2.9 свідчать, що кожний елемент підсистеми електроживлення виконує окрему функцію, спрямовану на забезпечення стабільної, безпечної та безперервної роботи програмно-апаратного комплексу.

Важливою особливістю запропонованої системи електроживлення є використання окремих ліній живлення для силових виконавчих механізмів і цифрової електроніки. Подібне рішення широко застосовується у сучасних промислових робототехнічних комплексах, оскільки забезпечує зниження рівня електромагнітних завад, покращує стабільність роботи мікроконтролерів та зменшує ймовірність виникнення збоїв під час одночасної роботи декількох приводів.

Крім того, модульна структура системи електроживлення забезпечує можливість подальшого розширення комплексу шляхом підключення нових електронних модулів без необхідності повного перепроектування силової

частини. Це підвищує універсальність запропонованого рішення та дозволяє адаптувати його до різних конфігурацій мобільних роботизованих платформ.

Таким чином, у результаті проєктування розроблено підсистему електроживлення програмно-апаратного комплексу для сортування поштових відправлень.

2.2.5 Розроблення системи інформаційної взаємодії між апаратними модулями програмно-апаратного комплексу

Після розроблення центрального модуля керування, локальної сенсорної підсистеми, підсистеми керування виконавчими механізмами та системи електроживлення наступним етапом проектування програмно-апаратного комплексу є організація інформаційної взаємодії між окремими апаратними модулями. Саме інформаційний обмін забезпечує узгоджену роботу всіх функціональних вузлів комплексу, своєчасну передачу команд керування, обробку інформації від сенсорної системи та синхронізацію роботи виконавчих механізмів.

Під час проектування системи інформаційної взаємодії особливу увагу приділено забезпеченню високої швидкості передачі даних, мінімізації часових затримок, підвищенню надійності обміну інформацією та можливості подальшого масштабування комплексу. Враховуючи розподілену архітектуру програмно-апаратної системи, інформаційний обмін повинен забезпечувати одночасну взаємодію між верхнім рівнем керування, центральним контролером, локальними сенсорними модулями та виконавчими механізмами без виникнення конфліктів доступу до спільних ресурсів.

Запропонована архітектура інформаційної взаємодії побудована за ієрархічним принципом. Верхній рівень представлений одноплатним комп'ютером Raspberry Pi 5 [6], який виконує високорівневі алгоритми керування, формує завдання для мобільної роботизованої платформи та взаємодіє з оператором. Центральним координатором інформаційного обміну

виступає мікроконтролер STM32H750VGT6 [27], який забезпечує виконання задач реального часу, обробляє інформацію від локальних контролерів, формує сигнали керування виконавчими механізмами та здійснює взаємодію між усіма апаратними модулями комплексу.

Для організації інформаційної взаємодії між Raspberry Pi 5 [6] та центральним контролером використовується високошвидкісний інтерфейс SPI, у якому Raspberry Pi виконує функції ведучого пристрою (Master), а STM32H750VGT6 [27] – підлеглого (Slave). Такий підхід дозволяє передавати керуючі команди, службову інформацію та результати обробки даних із мінімальними часовими затримками, що є особливо важливим під час роботи комплексу в режимі реального часу.

Обмін інформацією між центральним контролером та локальними сенсорними модулями також реалізовано за допомогою інтерфейсу SPI. При цьому мікроконтролер STM32H750VGT6 [26] виконує функції ведучого пристрою, а кожен із трьох мікроконтролерів STM32F103 [25] працює як окремий підлеглий модуль. Така організація дозволяє незалежно опитувати кожний локальний контролер, отримувати інформацію від відповідної групи датчиків та виконувати її подальшу обробку без взаємного впливу окремих інформаційних потоків.

Керування виконавчими механізмами здійснюється безпосередньо центральним контролером STM32H750VGT6 [26] шляхом формування цифрових сигналів STEP, DIR та EN для промислових драйверів CW-5045 [35]. Передача керуючих сигналів реалізована через окремі цифрові виходи мікроконтролера, що забезпечує високу точність формування часових інтервалів, синхронне керування декількома приводами та незалежне регулювання режимів роботи кожного двигуна.

Окрему інформаційну підсистему становить взаємодія центрального контролера із графічним LCD-дисплеєм. Через відповідний інтерфейс здійснюється відображення службової інформації, поточного стану комплексу,

результатів самодіагностики, параметрів роботи та повідомлень про можливі несправності. Використання локального засобу відображення інформації дозволяє оперативно контролювати роботу комплексу без необхідності використання зовнішнього комп'ютера.

Для забезпечення надійності роботи системи інформаційна взаємодія організована таким чином, щоб кожний функціональний модуль виконував лише власні спеціалізовані задачі. Raspberry Pi 5 відповідає за високорівневу логіку роботи комплексу, STM32H750VGT6 реалізує задачі керування в режимі реального часу, локальні контролери STM32F103 здійснюють опитування сенсорної системи, а драйвери CW-5045 забезпечують безпосереднє керування виконавчими механізмами. Подібний розподіл функцій дозволяє суттєво зменшити навантаження на центральний контролер, скоротити час обробки інформації та забезпечити високу продуктивність усієї системи.

Для наочного відображення організації інформаційної взаємодії між основними апаратними модулями структурну схему системи наведено на рисунку 2.6.

Як показано на рисунку 2.6, інформаційна взаємодія між апаратними модулями комплексу побудована за ієрархічним принципом. Верхній рівень керування представлений одноплатним комп'ютером Raspberry Pi 5, який передає команди центральному контролеру STM32H750VGT6 через інтерфейс SPI. Центральний контролер координує роботу локальних контролерів STM32F103, формуючи запити на опитування сенсорної системи та приймаючи результати вимірювань. Після аналізу отриманої інформації STM32H750VGT6 формує сигнали STEP, DIR та EN для драйверів CW-5045, які здійснюють керування кроковими двигунами мобільної платформи. Одночасно службова інформація про поточний стан комплексу передається на LCD-дисплей, а через інтерфейс UART забезпечується можливість сервісного налагодження системи.

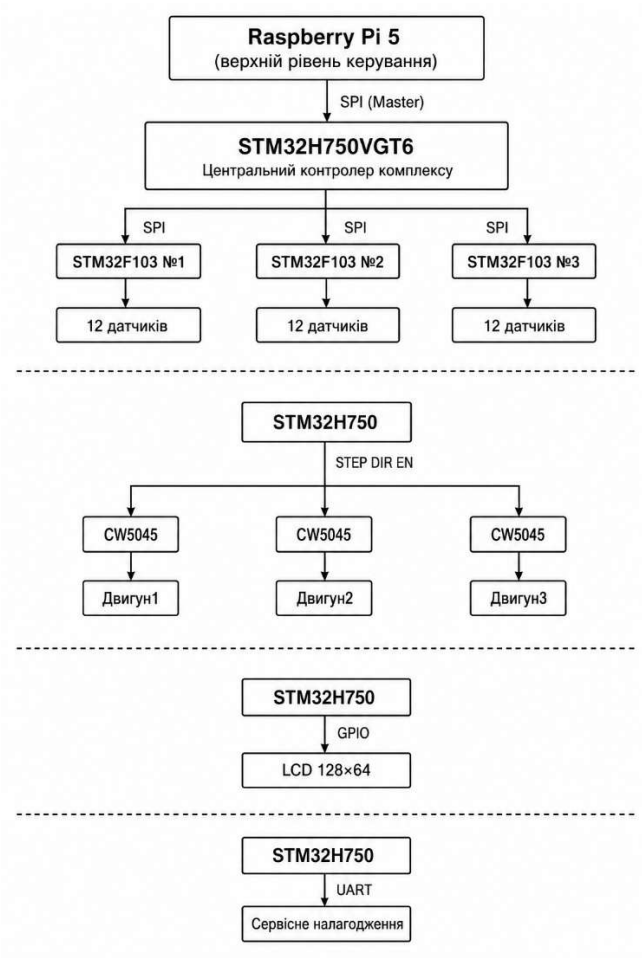


Рисунок 2.6 – Структурна схема інформаційної взаємодії
між апаратними модулями програмно-апаратного комплексу

Для узагальнення результатів проєктування та визначення ролі основних інтерфейсів у складі програмно-апаратного комплексу сформовано таблицю 2.10.

Використання окремих інформаційних інтерфейсів для різних функціональних підсистем дозволяє зменшити навантаження на центральний контролер та виключити конфлікти під час одночасної передачі інформації. Найбільш інтенсивний інформаційний обмін здійснюється між центральним контролером STM32H750VGT6 та локальними контролерами STM32F103, які передають результати опитування датчиків відстані. Передача команд від Raspberry Pi 5 має значно меншу інтенсивність, оскільки здійснюється лише під час зміни режимів роботи комплексу або формування нових завдань.

Таблиця 2.10 – Основні інформаційні інтерфейси програмно-апаратного комплексу

Інтерфейс	Модулі	Призначення
SPI	Raspberry Pi 5 ↔ STM32H750VGT6	Передача команд високого рівня та службових даних
SPI	STM32H750VGT6 ↔ STM32F103 №1	Передача даних від першої групи датчиків
SPI	STM32H750VGT6 ↔ STM32F103 №2	Передача даних від другої групи датчиків
SPI	STM32H750VGT6 ↔ STM32F103 №3	Передача даних від третьої групи датчиків
GPIO	STM32H750VGT6 → Драйвери CW-5045	Формування сигналів STEP, DIR, EN
GPIO	STM32H750VGT6 → LCD 128×64	Відображення службової інформації та параметрів роботи
UART	STM32H750VGT6 ↔ Персональний комп'ютер	Сервісне налагодження, діагностика та оновлення програмного забезпечення

Окремої уваги потребує забезпечення часової синхронізації інформаційного обміну. У запропонованій архітектурі головний контролер виконує функції координатора всієї системи та визначає порядок обміну інформацією між окремими модулями. Це дозволяє уникнути конфліктів на шині SPI, забезпечити детермінований час передачі даних та гарантувати виконання задач у режимі реального часу.

Аналіз таблиці 2.10 свідчить, що запропонована система інформаційної взаємодії використовує окремі інтерфейси для кожного типу інформаційного обміну. Це дозволяє мінімізувати конфлікти доступу до апаратних ресурсів, забезпечити високу швидкість передачі даних та спростити подальше розширення програмно-апаратного комплексу.

Таким чином, у результаті проєктування розроблено систему інформаційної взаємодії між основними модулями програмно-апаратного

комплексу. Запропонована архітектура забезпечує обмін даними між Raspberry Pi 5, STM32H750VGT6, локальними контролерами STM32F103, виконавчими механізмами та сервісними модулями. Використання SPI як основного каналу обміну забезпечує високу швидкість передавання даних і мінімальні затримки, а GPIO та UART підвищують надійність і зручність експлуатації. Система є модульною, масштабованою та відповідає вимогам сучасних робототехнічних програмно-апаратних комплексів.

2.3 Математичне моделювання та дослідження системи керування рухом мобільної роботизованої платформи

Розроблена мобільна роботизована платформа має триколісну всенапрямну ходову частину, побудовану із використанням коліс типу Omni. Кожне колесо приводиться в рух окремим кроковим двигуном через драйвер CW-5045, а незалежне керування швидкістю та напрямком обертання трьох приводів забезпечує можливість переміщення платформи у довільному напрямку без попередньої зміни орієнтації її корпусу.

Використання трьох приводних коліс, розташованих симетрично відносно геометричного центра платформи, дозволяє реалізувати три незалежні складові руху: поздовжнє переміщення, поперечне переміщення та обертання навколо вертикальної осі. Завдяки наявності вільно обертових роликів на поверхні Omni-коліс зменшується опір переміщенню у напрямку, перпендикулярному до основного напрямку кочення колеса.

Для побудови кінематичної моделі введено рухому систему координат Oxy, жорстко пов’язану з корпусом мобільної платформи. Початок координат Oрозташовано в геометричному центрі платформи. Вісь Oxвизначає поздовжній напрямок руху, а вісь Oy – поперечний напрямок. Положення платформи у нерухомій системі координат O_0XYвизначається координатами центра X, Yта кутом орієнтації ψ.

Кінематична модель будується за таких припущень:

– корпус платформи вважається абсолютно жорстким;

– усі приводні колеса мають однаковий радіус;

– відстані від центра платформи до осей коліс є однаковими;

– рух здійснюється по горизонтальній поверхні;

– проковзування коліс у напрямку їх основного кочення відсутнє;

– деформаціями коліс і механічних елементів конструкції можна знехтувати;

– кутові швидкості приводів формуються незалежно одна від одної.

Приймемо, що колеса розташовані на однаковій відстані L від центра платформи під кутами.

Основні параметри кінематичної моделі наведено в таблиці 2.11.

Таблиця 2.11 – Основні параметри кінематичної моделі мобільної платформи

r	Радіус приводного Omni-колеса	м
L	Відстань від центра платформи до осі колеса	м
v_x	Поздовжня складова швидкості платформи	м/с
v_y	Поперечна складова швидкості платформи	м/с
Ω	Кутова швидкість платформи	рад/с
ω_1	Кутова швидкість першого колеса	рад/с
ω_2	Кутова швидкість другого колеса	рад/с
ω_3	Кутова швидкість третього колеса	рад/с
ψ	Кут орієнтації платформи	рад

Вектор швидкості центра платформи в рухомій системі координат має вигляд [11]

$$\mathbf{v} = \begin{bmatrix} v_x \\ v_y \end{bmatrix}.$$

Крім поступального руху платформа може обертатися навколо вертикальної осі з кутовою швидкістю Ω. Тому швидкість точки встановлення кожного колеса визначається як сума поступальної швидкості центра платформи та швидкості, обумовленої її обертанням.

Радіус-вектор точки встановлення i-го колеса можна подати у вигляді

$$\mathbf{l}_i = L \begin{bmatrix} \cos\theta_i \\ \sin\theta_i \end{bmatrix}.$$

Одиничний вектор основного напрямку кочення відповідного колеса визначається залежністю

$$\mathbf{t}_i = \begin{bmatrix} -\sin\theta_i \\ \cos\theta_i \end{bmatrix}.$$

Швидкість точки встановлення i-го колеса дорівнює

$$\mathbf{v}_i = \begin{bmatrix} v_x \\ v_y \end{bmatrix} + \Omega \begin{bmatrix} -L\sin\theta_i \\ L\cos\theta_i \end{bmatrix}.$$

Проєкція отриманої швидкості на основний напрямок кочення колеса повинна дорівнювати його лінійній коловій швидкості:

$$r\omega_i = \mathbf{t}_i^T \mathbf{v}_i.$$

Після підстановки складових швидкості отримано загальне рівняння для кожного приводу:

$$r\omega_i = -\sin\theta_i\, v_x + \cos\theta_i\, v_y + L\Omega.$$

З урахуванням прийнятих кутів розташування коліс кінематичні рівняння платформи набувають вигляду [11]

$$\omega_1 = \frac{1}{r}\left(v_y + L\Omega\right),$$

$$\omega_2 = \frac{1}{r}\left(-\frac{\sqrt{3}}{2}v_x - \frac{1}{2}v_y + L\Omega\right),$$

$$\omega_3 = \frac{1}{r}\left(\frac{\sqrt{3}}{2}v_x - \frac{1}{2}v_y + L\Omega\right).$$

У матричній формі пряма кінематична залежність між швидкістю платформи та швидкостями приводних коліс має вигляд

$$\begin{bmatrix}\omega_1\\ \omega_2\\ \omega_3\end{bmatrix} = \frac{1}{r}\begin{bmatrix}0 & 1 & L\\ -\frac{\sqrt{3}}{2} & -\frac{1}{2} & L\\ \frac{\sqrt{3}}{2} & -\frac{1}{2} & L\end{bmatrix}\begin{bmatrix}v_x\\ v_y\\ \Omega\end{bmatrix}.$$

Наведена залежність використовується для визначення необхідних кутових швидкостей кожного колеса за заданими параметрами руху мобільної платформи. Команда верхнього рівня може містити бажані значення v_x, v_yта Ω, після чого центральний контролер STM32H750VGT6 обчислює відповідні значення ω_1, ω_2та ω_3 [11].

Для визначення фактичних складових руху платформи за відомими швидкостями приводів використовується обернена кінематична модель:

$$v_x = \frac{r}{\sqrt{3}}\left(\omega_3 - \omega_2\right),$$

$$v_y = \frac{r}{3}\left(2\omega_1 - \omega_2 - \omega_3\right),$$

$$\Omega = \frac{r}{3L}\left(\omega_1 + \omega_2 + \omega_3\right).$$

Відповідно, обернене кінематичне перетворення у матричному вигляді можна записати як

$$\begin{bmatrix} v_x \\ v_y \\ \Omega \end{bmatrix} = r \begin{bmatrix} 0 & -\frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \\ \frac{2}{3} & -\frac{1}{3} & -\frac{1}{3} \\ \frac{1}{3L} & \frac{1}{3L} & \frac{1}{3L} \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix}.$$

Таким чином, розроблена математична модель встановлює взаємозв’язок між заданими поздовжньою, поперечною та кутовою швидкостями мобільної роботизованої платформи і швидкостями обертання трьох приводних Omni-коліс. Отримані залежності дозволяють визначати напрямки обертання приводів, необхідну кількість керувальних імпульсів та частоту сигналів STEP, що формуються апаратними таймерами STM32H750VGT6.

Керування приводами здійснюється за розімкненим принципом, оскільки у конструкції не передбачено безпосереднього вимірювання кутового положення коліс. За умови роботи двигунів у межах допустимого моменту навантаження задане переміщення визначається кількістю сформованих імпульсів STEP. Для зменшення ймовірності втрати кроків програмне забезпечення реалізує обмеження максимальної швидкості, а також плавний розгін і гальмування приводів.

Побудована модель може бути використана програмним забезпеченням центрального контролера для перетворення команд верхнього рівня у сигнали керування окремими виконавчими механізмами та є основою для експериментальної перевірки точності руху мобільної роботизованої платформи.

3 РОЗРОБЛЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ПРОГРАМНО-АПАРАТНОГО КОМПЛЕКСУ ДЛЯ СОРТУВАННЯ ПОШТОВИХ ВІДПРАВЛЕНЬ

3.1 Загальна архітектура програмного забезпечення

Після розроблення апаратного забезпечення наступним етапом створення програмно-апаратного комплексу є проектування програмного забезпечення, яке забезпечує взаємодію всіх функціональних модулів системи та реалізує алгоритми автоматизованого сортування поштових відправлень. Саме програмне забезпечення визначає логіку роботи комплексу, координує обмін інформацією між окремими пристроями, здійснює обробку даних від сенсорної підсистеми та формує керуючі впливи для виконавчих механізмів.

Під час проектування програмного забезпечення було прийнято рішення використовувати багаторівневу архітектуру, яка відповідає структурі апаратної частини комплексу. Такий підхід дозволяє розділити задачі між окремими обчислювальними модулями відповідно до їх функціонального призначення та обчислювальних можливостей.

Верхній рівень програмного забезпечення реалізовано на одноплатному комп'ютері Raspberry Pi 5 [29]. Він відповідає за виконання високорівневих алгоритмів керування, взаємодію з оператором, формування маршрутів руху мобільної роботизованої платформи, аналіз інформації, що надходить від сенсорної системи, а також координацію роботи всього комплексу.

Задачі реального часу покладено на високопродуктивний мікроконтролер STM32H750VGT6 [26]. Його програмне забезпечення забезпечує керування виконавчими механізмами, формування сигналів STEP, DIR та EN для драйверів крокових двигунів, організацію обміну інформацією з локальними контролерами STM32F103, приймання команд від Raspberry Pi та координацію роботи всіх

апаратних модулів.

Для розвантаження центрального контролера використано три локальні мікроконтролери STM32F103. Програмне забезпечення кожного локального контролера виконує циклічне опитування підключеної групи датчиків відстані, здійснює первинну обробку отриманої інформації, перевірку її достовірності та передає результати центральному контролеру за допомогою інтерфейсу SPI.

Обмін інформацією між програмними модулями організовано відповідно до ієрархічної структури комплексу [22]. Raspberry Pi 5 передає команди верхнього рівня центральному контролеру, який виконує їх інтерпретацію, формує необхідні керуючі сигнали та координує роботу локальних контролерів. У свою чергу локальні модулі передають результати вимірювань, що використовуються центральним контролером для прийняття рішень щодо подальших дій мобільної роботизованої платформи.

Запропонована структура програмного забезпечення забезпечує модульність побудови системи, що дозволяє незалежно вдосконалювати окремі програмні модулі без зміни загальної архітектури комплексу. Крім того, такий підхід значно спрощує налагодження, тестування та подальшу модернізацію системи.

Для наочного відображення взаємодії програмних модулів загальну структуру програмного забезпечення наведено на рисунку 3.1.

Як показано на рисунку 3.1, програмне забезпечення побудовано за ієрархічним принципом, у якому кожний рівень виконує лише власні функції. Такий підхід дозволяє мінімізувати навантаження на центральний контролер, забезпечити виконання задач реального часу та підвищити загальну надійність роботи комплексу. Завдяки модульній організації програмного забезпечення окремі програмні компоненти можуть модернізуватися незалежно один від одного, що значно спрощує супровід системи та її подальший розвиток.

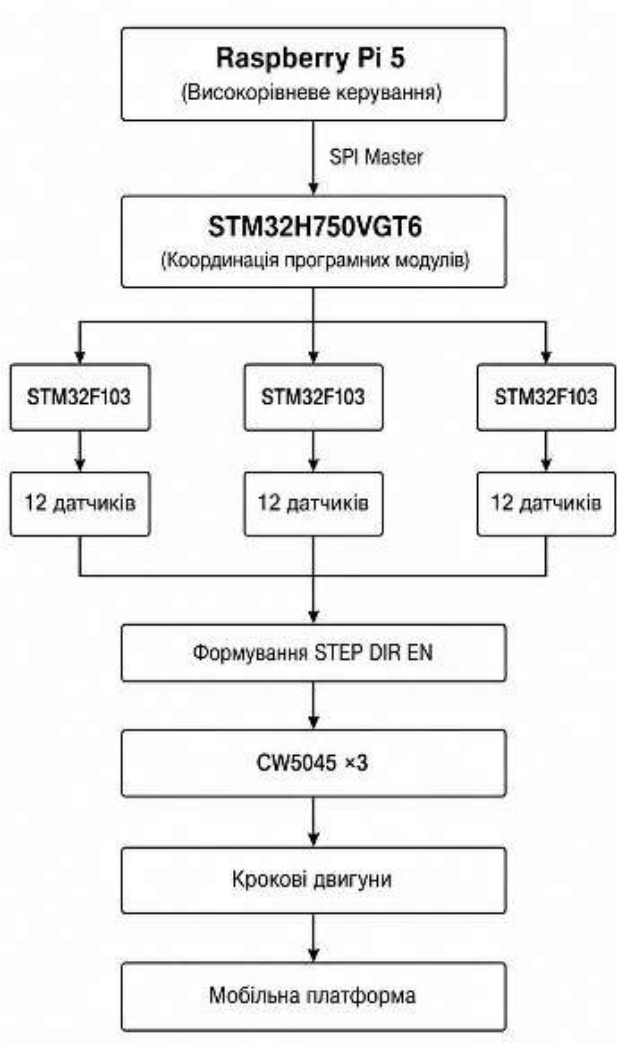


Рисунок 3.1 – Загальна архітектура програмного забезпечення програмно-апаратного комплексу

3.2 Програмне забезпечення центрального контролера STM32H750

Центральним елементом програмного забезпечення розробленого програмно-апаратного комплексу є високопродуктивний мікроконтролер STM32H750VGT6, який виконує функції головного контролера всієї системи сортування поштових відправлень. Саме він забезпечує координацію роботи виконавчих механізмів, взаємодію із периферійними мікроконтролерами, прийом команд від одноплатного компютера Raspberry Pi 5, а також контроль поточного стану обладнання.

Використання STM32H750VGT6 обумовлено його високою

продуктивністю, яка забезпечується ядром ARM Cortex-M7 з максимальною тактовою частотою до 480 МГц, наявністю великого обсягу оперативної пам'яті, сучасної системи шин, апаратної підтримки операцій із плаваючою комою, великої кількості периферійних інтерфейсів та таймерів високої точності. Це дозволяє реалізувати одночасне виконання великої кількості задач реального часу без зниження продуктивності системи.

Основними функціями центрального контролера є приймання команд від Raspberry Pi 5, аналіз отриманих команд, керування трьома кроковими двигунами маніпулятора, синхронізація роботи осей, обмін даними з периферійними контролерами STM32F103, контроль аварійних ситуацій, передавання службової інформації до Raspberry Pi та забезпечення роботи програмного комплексу в режимі реального часу.

Таким чином, STM32H750 є центральною ланкою системи керування, тоді як Raspberry Pi виконує задачі верхнього рівня, пов’язані з комп’ютерним зором, прийняттям логічних рішень та формуванням команд керування.

3.2.1 Архітектура програмного забезпечення STM32H750

Програмне забезпечення центрального контролера реалізовано за модульним принципом. Кожний функціональний блок відповідає лише за виконання власної задачі, що значно спрощує супровід програмного коду, локалізацію можливих помилок та подальшу модернізацію програмного забезпечення.

Структура програмного забезпечення складається із декількох незалежних модулів:

- модуль ініціалізації обладнання;
- модуль обробки команд;
- модуль керування приводами;
- модуль обміну даними;
- модуль контролю стану системи;

– модуль аварійного захисту;

– модуль діагностики.

На етапі запуску виконується послідовна ініціалізація всіх апаратних ресурсів мікроконтролера, після чого програма переходить у головний цикл виконання. У цьому циклі безперервно здійснюється прийом нових команд, оновлення стану виконавчих механізмів, контроль справності обладнання та передача інформації до інших модулів системи.

Подібний підхід забезпечує високу масштабованість програмного забезпечення та дозволяє додавати нові функції без внесення суттєвих змін у вже реалізовані модулі.

3.2.2 Ініціалізація апаратних ресурсів

Після подачі живлення або виконання апаратного скидання мікроконтролер переходить до процедури ініціалізації всіх необхідних периферійних пристроїв. Даний етап є одним із найважливіших, оскільки саме від правильності налаштування периферії залежить стабільність роботи всієї роботизованої системи.

Під час ініціалізації виконується:

– налаштування системної тактової частоти;

– конфігурування внутрішньої пам'яті;

– налаштування GPIO;

– запуск апаратних таймерів;

– конфігурування інтерфейсу SPI;

– запуск UART;

– ініціалізація DMA;

– запуск системи переривань NVIC;

– запуск системного таймера.

Після успішного завершення всіх процедур формується службове повідомлення про готовність центрального контролера до роботи, яке

передається Raspberry Pi.

Перед початком виконання основної програми також виконується перевірка працездатності критично важливих периферійних модулів. У випадку виявлення помилки подальший запуск системи блокується до моменту усунення несправності.

3.2.3 Організація головного циклу програми

Основна програма STM32H750 побудована відповідно до класичної архітектури нескінченного циклу (Super Loop), усередині якого виконуються всі основні функції системи керування.

Головний цикл можна умовно розділити на декілька послідовних етапів:

– прийом нових повідомлень від Raspberry Pi;
– аналіз отриманої команди;
– перевірка поточного стану обладнання;
– формування керуючих сигналів;
– передача команд драйверам двигунів;
– отримання інформації від STM32F103;
– оновлення внутрішніх змінних;
– передача телеметрії до Raspberry Pi.

Подібний алгоритм забезпечує безперервне функціонування системи та своєчасну реакцію на будь-які зміни зовнішніх умов.

Особливістю реалізованої архітектури є відсутність тривалих блокуючих затримок. Усі часові інтервали формуються апаратними таймерами або механізмом переривань, що дозволяє забезпечити високу швидкодію програмного забезпечення.

3.2.4 Організація обміну даними між Raspberry Pi та STM32H750

Ефективна робота програмно-апаратного комплексу сортування поштових відправлень безпосередньо залежить від швидкого та надійного інформаційного обміну між одноплатним комп'ютером Raspberry Pi 5 та центральним мікроконтролером STM32H750VGT6. Саме через цей канал передаються результати роботи системи комп'ютерного зору, координати об'єктів, службові команди керування, а також інформація про поточний стан обладнання [22].

У розробленій системі Raspberry Pi 5 виконує функції верхнього рівня керування. На ньому працюють алгоритми комп’ютерного зору, модулі обробки зображень, програмне забезпечення сортування поштових відправлень та формування команд для виконавчих механізмів. Центральний мікроконтролер STM32H750 виконує функції нижнього рівня керування, забезпечуючи точне позиціонування маніпулятора, синхронізацію роботи приводів, обробку сигналів від периферійних контролерів та контроль технічного стану обладнання.

Для організації інформаційного обміну було обрано інтерфейс SPI (Serial Peripheral Interface). Використання даного інтерфейсу обумовлено його високою швидкодією, мінімальними затримками передачі даних, простотою апаратної реалізації та широкою підтримкою як мікроконтролерами сімейства STM32, так і одноплатним комп'ютером Raspberry Pi. На відміну від інтерфейсів UART або I²C, SPI дозволяє здійснювати повнодуплексний обмін даними із тактовою частотою у декілька десятків мегагерц, що є достатнім для передачі координат об'єктів та команд керування у режимі реального часу [22].

У розробленій системі Raspberry Pi 5 працює у режимі Master, тоді як STM32H750 функціонує у режимі Slave. Передача інформації ініціюється Raspberry Pi після завершення обробки чергового кадру з камери. Центральний контролер приймає пакет даних, перевіряє його цілісність, після чого виконує аналіз отриманої команди та формує відповідні сигнали керування виконавчими механізмами.

Для забезпечення надійності роботи всі повідомлення мають фіксовану

структуру. Кожен пакет містить службову інформацію, ідентифікатор команди, параметри керування, контрольну суму та службовий байт завершення передачі. Подібна структура дозволяє виявляти помилки передачі даних, уникати виконання некоректних команд та забезпечує стабільну роботу системи навіть за наявності електромагнітних завад.

При отриманні нового пакета STM32H750 виконує послідовну перевірку правильності формату повідомлення, контрольної суми та допустимості параметрів команди. Якщо пакет сформовано коректно, команда передається до відповідних програмних модулів системи керування. У випадку виявлення помилки пакет відхиляється, а Raspberry Pi отримує повідомлення про необхідність повторної передачі інформації.

Такий підхід забезпечує надійний інформаційний обмін між усіма рівнями програмно-апаратного комплексу, мінімізує ймовірність виникнення помилок керування та гарантує своєчасне виконання команд навіть під час інтенсивної роботи системи сортування. Структуру інформаційного обміну між Raspberry Pi 5 та STM32H750VGT6 наведено на рисунку 3.2.

3.2.5 Алгоритм керування кроковими двигунами

Керування виконавчими механізмами є одним із ключових завдань центрального контролера STM32H750VGT6. Саме програмне забезпечення центрального контролера забезпечує формування керуючих сигналів для промислових драйверів CW-5045, які здійснюють безпосереднє керування кроковими двигунами мобільної роботизованої платформи. Від правильності реалізації алгоритму керування залежить точність позиціонування платформи, плавність її руху, стабільність роботи всієї системи та ефективність виконання операцій сортування поштових відправлень [36].

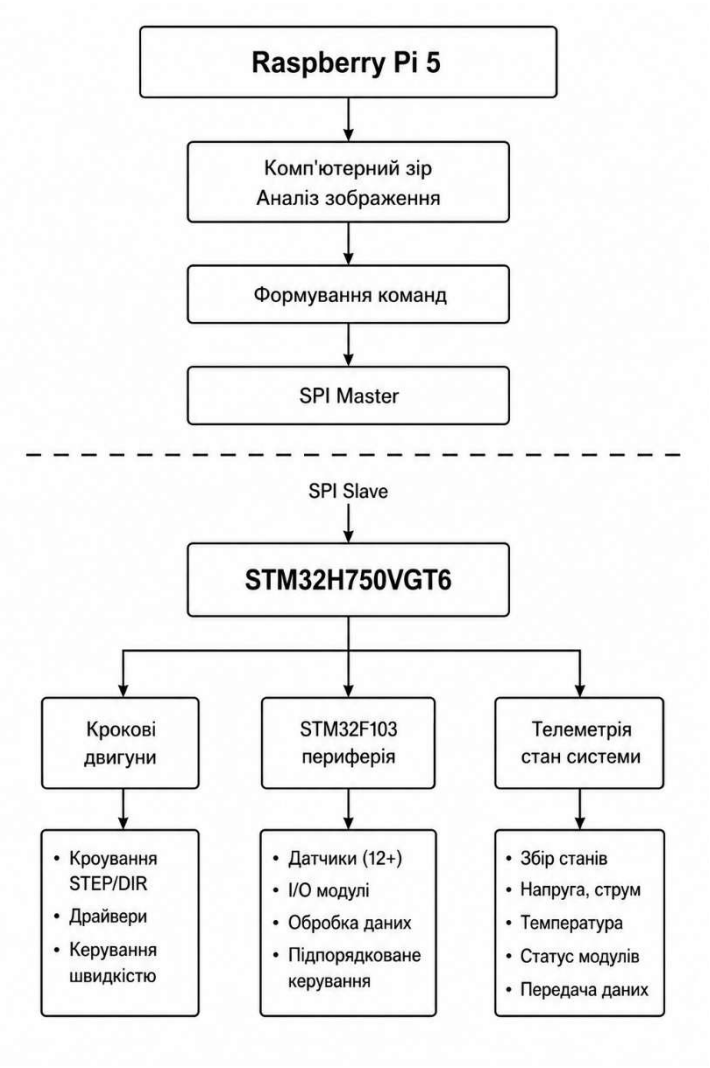


Рисунок 3.2 – Структура інформаційного обміну між Raspberry Pi 5 та STM32H750

На відміну від звичайних електродвигунів постійного струму, крокові двигуни дозволяють здійснювати точне позиціонування без використання додаткових датчиків положення за умови роботи в межах допустимого навантаження. Кожний керуючий імпульс відповідає одному кроку ротора, а використання режиму мікрокрокового керування, реалізованого драйверами CW-5045, забезпечує високу плавність руху та суттєве зменшення механічних вібрацій.

У розробленому програмно-апаратному комплексі для керування кожним драйвером використовуються три цифрові сигнали:

– STEP – формування імпульсів переміщення;

– DIR – вибір напрямку обертання;

– EN – увімкнення або вимкнення драйвера.

Формування зазначених сигналів здійснюється апаратними таймерами мікроконтролера STM32H750VGT6, що дозволяє забезпечити високу точність часових інтервалів між імпульсами та мінімізувати навантаження на центральне процесорне ядро.

Перед початком руху програмне забезпечення аналізує команду, отриману від Raspberry Pi 5, визначає необхідний напрямок переміщення, кількість кроків та швидкість руху кожного двигуна. На основі цих параметрів виконується підготовка таймерів і запуск процесу генерації імпульсів STEP. Послідовність формування керувальних сигналів STEP, DIR та EN для драйверів CW-5045 наведено на рисунку 3.3.



Рисунок 3.3 – Загальна структура алгоритму керування кроковими двигунами

Як показано на рисунку 3.3, алгоритм керування складається з декількох послідовних етапів, починаючи з аналізу отриманої команди та завершуючи контролем виконання переміщення. Такий підхід забезпечує передбачувану роботу системи керування та дозволяє реалізувати точне позиціонування мобільної роботизованої платформи.

Після завершення підготовчих операцій центральний контролер формує сигнали керування відповідно до параметрів отриманої команди. Напрямок руху визначається встановленням логічного рівня на вході DIR драйвера CW-5045, після чого виконується генерація необхідної кількості імпульсів STEP. Після завершення переміщення драйвер може бути переведений у режим очікування шляхом зміни стану сигналу EN.

Використання апаратних таймерів дозволяє забезпечити однакові часові інтервали між сусідніми імпульсами незалежно від навантаження на центральний процесор. Завдяки цьому виключаються нерівномірності руху, характерні для програмної генерації імпульсів.

Основні сигнали керування драйвером CW-5045 наведено в таблиці 3.2.

Таблиця 3.2 – Основні сигнали керування драйвером CW-5045

Сигнал	Призначення	Джерело формування
STEP	Формування кроків двигуна	Таймер STM32H750
DIR	Вибір напрямку руху	GPIO STM32H750
EN	Увімкнення або вимкнення драйвера	GPIO STM32H750

Як видно з таблиці 3.2, усі сигнали керування формуються безпосередньо центральним контролером STM32H750VGT6. Використання окремих цифрових ліній для кожного сигналу спрощує програмну реалізацію алгоритму керування та забезпечує високу швидкодію системи.

Для зменшення динамічних навантажень на механічну конструкцію мобільної платформи програмне забезпечення реалізує плавний розгін та плавне гальмування двигунів. Замість миттєвого переходу до максимальної швидкості частота імпульсів STEP поступово збільшується на етапі розгону, після чого підтримується сталою протягом основної частини переміщення. Перед завершенням руху швидкість аналогічно поступово зменшується.

Такий підхід дозволяє:

– зменшити втрату кроків;
– знизити механічні вібрації;
– підвищити точність позиціонування;
– збільшити ресурс механічних вузлів;
– забезпечити більш стабільний рух платформи.

Оскільки мобільна платформа використовує декілька крокових двигунів, програмне забезпечення STM32H750 забезпечує їх синхронізовану роботу. Центральний контролер формує незалежні сигнали STEP для кожного драйвера CW-5045, координуючи початок, швидкість та завершення руху кожної осі.

Використання декількох апаратних таймерів дозволяє реалізувати незалежне керування приводами без взаємного впливу між ними, що особливо важливо під час виконання складних маневрів або одночасного переміщення декількох виконавчих механізмів.

Таким чином, розроблений алгоритм керування кроковими двигунами забезпечує точне формування сигналів STEP, DIR та EN, синхронізоване керування декількома приводами та реалізацію плавного розгону і гальмування мобільної роботизованої платформи. Використання апаратних таймерів STM32H750VGT6 дозволяє мінімізувати навантаження на центральний процесор і забезпечити стабільну роботу системи в режимі реального часу, що є важливою умовою для надійного функціонування програмно-апаратного комплексу сортування поштових відправлень.

3.2.6 Розроблення програмного забезпечення Raspberry Pi 5

Програмне забезпечення верхнього рівня програмно-апаратного комплексу реалізоване на одноплатному комп'ютері Raspberry Pi 5, який виконує функції центрального обчислювального вузла системи. Основним призначенням Raspberry Pi є приймання інформації від камери машинного зору, виконання алгоритмів комп'ютерного бачення, формування команд керування мобільною платформою та організація інформаційної взаємодії з центральним контролером STM32H750VGT6.

Використання Raspberry Pi 5 як верхнього рівня керування обумовлено значною обчислювальною продуктивністю, наявністю багатоядерного процесора ARM Cortex-A76, достатнім обсягом оперативної пам'яті та можливістю використання сучасних бібліотек комп'ютерного зору. На відміну від мікроконтролерів сімейства STM32, Raspberry Pi працює під керуванням операційної системи Linux, що значно спрощує розроблення складного програмного забезпечення, використання багатопотокового виконання програм та інтеграцію з програмними бібліотеками OpenCV [17] і TensorFlow Lite [37].

Архітектура програмного забезпечення Raspberry Pi побудована за модульним принципом. Кожний програмний модуль виконує лише власні функції, що забезпечує простоту супроводу, тестування та подальшої модернізації системи. Обмін інформацією між окремими програмними компонентами здійснюється через внутрішні програмні інтерфейси, а взаємодія із центральним контролером реалізується за допомогою інтерфейсу SPI, у якому Raspberry Pi виконує функції Master-пристрою.

До складу програмного забезпечення Raspberry Pi входить модуль отримання відеопотоку з камери, модуль попередньої обробки зображення, модуль аналізу просторового положення поштових відправлень, модуль прийняття рішень та модуль формування команд керування. Отримане зображення проходить декілька етапів цифрової обробки, серед яких корекція освітлення, фільтрація шумів, виділення області інтересу та пошук контурів

об'єктів. Після цього виконується розпізнавання QR-коду [38] або іншого ідентифікатора, визначається місце призначення відправлення та розраховується необхідна траєкторія руху мобільної роботизованої платформи.

Результатом роботи алгоритмів комп’ютерного зору є формування компактного набору параметрів, який передається центральному контролеру STM32H750VGT6. До складу повідомлення входять координати цільової комірки, необхідний режим роботи, службові прапори та контрольна сума для перевірки цілісності переданих даних. Передача виконується пакетним способом, що дозволяє мінімізувати навантаження на канал зв'язку та скоротити час інформаційного обміну.

Після передачі пакета даних центральний контролер STM32H750VGT6 виконує перевірку правильності отриманої інформації, після чого розпочинає автономне виконання алгоритму переміщення платформи, керування приводами та опитування сенсорної підсистеми. Такий підхід дозволяє розділити задачі високорівневої цифрової обробки зображення та задачі реального часу між двома незалежними обчислювальними платформами.

Використання багатопотокового програмування дозволяє одночасно виконувати декілька незалежних процесів: захоплення відеозображення, аналіз кадрів, інформаційний обмін через SPI, ведення журналу подій та контроль працездатності системи. Завдяки цьому навіть під час інтенсивної обробки відеоданих забезпечується безперервний зв'язок із центральним контролером та своєчасне формування команд керування.

Таким чином, програмне забезпечення Raspberry Pi 5 реалізує функції верхнього рівня керування програмно-апаратним комплексом, забезпечує виконання алгоритмів комп'ютерного зору, прийняття рішень щодо маршруту переміщення мобільної платформи та формування команд для центрального контролера STM32H750VGT6. Запропонована архітектура забезпечує високу продуктивність, модульність програмного забезпечення та можливість подальшого розширення функціональних можливостей комплексу.

Загальну архітектуру програмного забезпечення Raspberry Pi 5 наведено на рисунку 3.4.

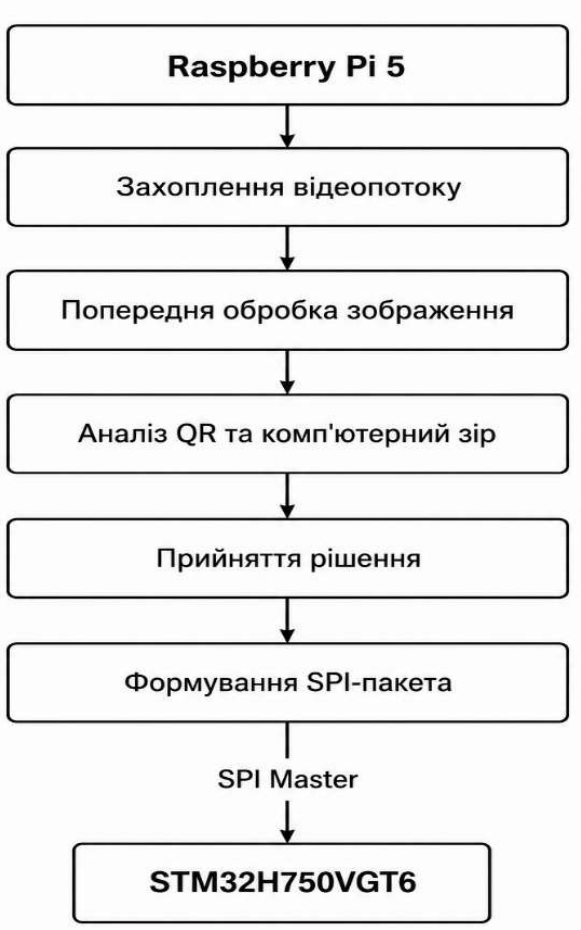


Рисунок 3.4 – Архітектура програмного забезпечення Raspberry Pi 5

3.2.7 Забезпечення надійності, діагностики та налагодження програмно-апаратного комплексу

Надійність функціонування програмно-апаратного комплексу є одним із найважливіших факторів, що визначають ефективність автоматизованого сортування поштових відправлень. Під час експлуатації робототехнічної системи можливе виникнення різноманітних програмних або апаратних несправностей, пов'язаних із порушенням інформаційного обміну, відмовою окремих периферійних модулів, помилками передавання даних або нестабільною роботою виконавчих механізмів. Для забезпечення безперервної та безпечної роботи комплексу у програмному забезпеченні реалізовано комплекс

засобів контролю, діагностики та аварійного реагування.

Основною задачею системи діагностики є постійний контроль працездатності всіх функціональних модулів комплексу. Перевірка здійснюється як на етапі запуску програмного забезпечення, так і безперервно під час роботи мобільної роботизованої платформи. Такий підхід дозволяє своєчасно виявляти можливі несправності та запобігати виникненню аварійних ситуацій.

На етапі ініціалізації центральний контролер STM32H750VGT6 виконує послідовну перевірку власних периферійних модулів, працездатності інтерфейсів SPI та UART, доступності локальних контролерів STM32F103, а також готовності драйверів CW-5045 до роботи. Лише після успішного завершення всіх перевірок система переходить до виконання основного алгоритму керування.

Під час роботи комплексу центральний контролер циклічно контролює надходження інформації від периферійних модулів. Для кожного локального контролера STM32F103 контролюється своєчасність отримання відповіді на запити, правильність структури отриманих пакетів та коректність даних, що надходять від сенсорної системи. У випадку відсутності відповіді або виявлення помилки передачі виконується повторне опитування відповідного модуля. Якщо після декількох спроб зв'язок не відновлюється, формується повідомлення про несправність, а система переходить до безпечного режиму роботи.

Особлива увага приділяється контролю інформаційного обміну між Raspberry Pi 5 та центральним контролером STM32H750VGT6. Кожен отриманий пакет проходить перевірку структури, контрольної суми та правильності службових полів. Це дозволяє виявляти випадкові помилки передачі даних, викликані електромагнітними завадами або порушенням роботи окремих апаратних модулів. У випадку отримання некоректного пакета команда не виконується, а Raspberry Pi отримує повідомлення про необхідність повторної передачі інформації.

Одним із засобів підвищення надійності програмного забезпечення є

використання апаратного сторожового таймера (Independent Watchdog) [25-27]. Центральний контролер періодично виконує його оновлення під час нормальної роботи головного циклу програми. Якщо внаслідок програмної помилки або непередбаченої ситуації виконання програми буде заблоковано, сторожовий таймер автоматично ініціює перезавантаження мікроконтролера, що дозволяє відновити працездатність системи без втручання оператора.

Крім контролю програмних модулів, реалізовано контроль стану виконавчих механізмів. Перед початком переміщення перевіряється готовність драйверів CW-5045 до роботи, після завершення руху контролюється коректність виконання команди, а також відсутність перевищення допустимого часу виконання операції. У випадку виникнення аварійної ситуації формується команда аварійної зупинки, драйвери вимикаються, а система переходить у режим очікування до усунення несправності.

Для спрощення процесу налагодження програмного забезпечення реалізовано механізм ведення службових повідомлень. Через інтерфейс UART центральний контролер передає інформацію про запуск системи, результати ініціалізації, встановлення зв’язку із периферійними контролерами, виконання команд керування, а також повідомлення про виявлені несправності. Отримані повідомлення можуть використовуватися під час тестування програмного забезпечення, аналізу роботи системи та пошуку причин можливих відмов.

Для наочного відображення послідовності виконання процедур контролю працездатності програмно-апаратного комплексу алгоритм роботи системи діагностики наведено на рисунку 3.5.

Як показано на рисунку 3.5, система діагностики функціонує безперервно протягом усього часу роботи комплексу. Після завершення початкової ініціалізації здійснюється циклічний контроль стану всіх основних модулів, перевірка інформаційного обміну та оновлення сторожового таймера.



Рисунок 3.5 – Алгоритм роботи програмної системи діагностики

У разі виникнення несправностей програмне забезпечення виконує повторну перевірку, а за неможливості усунення помилки переводить комплекс у безпечний режим роботи. Основні функції програмної системи діагностики, контрольовані параметри та дії у разі виявлення несправностей наведено в таблиці 3.3.

Таблиця 3.3 – Основні функції програмної системи діагностики

Контрольований модуль	Параметр контролю	Дії при виявленні несправності
Raspberry Pi 5	Наявність зв'язку SPI	Повторна передача пакета
STM32H750VGT6	Виконання головного циклу	Перезавантаження через IWDG
STM32F103	Отримання відповіді на SPI-запит	Повторне опитування
Драйвери CW-5045	Готовність до роботи	Блокування запуску двигуна
Датчики відстані	Коректність отриманих даних	Повторне вимірювання
UART	Передача службових повідомлень	Запис інформації про помилку

Дані таблиці 3.3 свідчать, що система діагностики охоплює всі критично важливі компоненти програмно-апаратного комплексу. Використання багаторівневого контролю дозволяє своєчасно виявляти програмні та апаратні несправності, мінімізувати ймовірність аварійних ситуацій і забезпечити стабільну роботу комплексу під час автоматизованого сортування поштових відправлень.

Таким чином, у результаті розроблення програмної системи діагностики реалізовано комплекс механізмів контролю працездатності, перевірки інформаційного обміну, моніторингу виконавчих механізмів та автоматичного відновлення роботи програмного забезпечення після виникнення окремих програмних збоїв. Використання сторожового таймера, перевірки цілісності пакетів даних, повторного опитування периферійних модулів та алгоритмів аварійної зупинки дозволяє значно підвищити надійність і безпечність експлуатації розробленого програмно-апаратного комплексу.

3.3 Розроблення конструкції мобільної роботизованої платформи

Після завершення розроблення електронної архітектури програмно-апаратного комплексу наступним етапом проектування стало створення конструкції мобільної роботизованої платформи, яка забезпечує розміщення всіх електронних модулів, виконавчих механізмів та сенсорної підсистеми в єдиній механічній конструкції. Від якості проектування корпусу безпосередньо залежать надійність роботи комплексу, зручність його складання, можливість подальшої модернізації та простота технічного обслуговування [22].

Під час розроблення конструкції було враховано особливості експлуатації мобільної роботизованої платформи у приміщеннях логістичних центрів. На відміну від промислового обладнання, яке працює у стаціонарному положенні, мобільна платформа постійно перебуває у русі, виконує численні повороти, прискорення та гальмування. Саме тому конструкція повинна забезпечувати

достатню жорсткість, невелику масу, правильний розподіл навантаження між колесами та стійкість до механічних навантажень.

Одним із перших етапів проектування стало визначення геометричної форми корпусу. Для даної роботи було обрано круглу конфоновку шасі, яка забезпечує однакові габарити платформи незалежно від напрямку її руху. Подібна форма значно спрощує маневрування у вузьких проходах між складськими стелажами, зменшує ризик механічного контакту корпусу із зовнішніми перешкодами та забезпечує можливість встановлення сенсорів по всьому периметру без утворення «мертвих зон».

Додатковою перевагою круглої конструкції є можливість рівномірного розташування силових елементів, електронних модулів та джерел живлення відносно геометричного центра платформи. Завдяки цьому центр мас знаходиться максимально близько до осі симетрії, що позитивно впливає на стійкість руху та зменшує навантаження на окремі приводи під час виконання маневрів.

Проєктування конструкції виконувалося із використанням сучасної CAD-системи тривимірного моделювання. Використання параметричного моделювання дозволило ще на етапі проектування визначити взаємне розташування всіх механічних та електронних компонентів, перевірити можливість їх монтажу, забезпечити необхідні монтажні зазори та оцінити доступність окремих вузлів під час технічного обслуговування.

Особливу увагу було приділено модульному принципу побудови корпусу. Замість виготовлення монолітної конструкції платформа складається з окремих деталей, кожна з яких виконує власну функцію. Подібний підхід значно спрощує процес виготовлення виробу за допомогою технології адитивного виробництва, дозволяє замінювати окремі елементи без повного демонтажу конструкції та забезпечує можливість модернізації окремих вузлів без зміни загальної архітектури платформи.

Конструкція корпусу передбачає використання двох основних

горизонтальних рівнів. Нижня частина використовується для встановлення силових приводів, механічних вузлів та елементів ходової частини. Верхній рівень призначений для розміщення електронних модулів, контролерів, системи живлення, комутаційних плат та допоміжного обладнання. Такий розподіл дозволяє відокремити механічні вузли від електроніки, спростити прокладання кабельних з'єднань та забезпечити кращий доступ до окремих компонентів під час налагодження.

При проектуванні також було передбачено достатню кількість технологічних отворів для встановлення кріпильних елементів, монтажу кабельних стяжок, вентиляції внутрішнього простору корпусу та виведення роз'ємів для підключення периферійного обладнання. Усі посадкові місця проектувалися відповідно до геометричних розмірів використовуваних електронних модулів, що дозволило мінімізувати використання додаткових перехідних кронштейнів.

Окрему увагу було приділено забезпеченню жорсткості конструкції. Незважаючи на використання полімерних матеріалів, форма основних деталей була спроєктована таким чином, щоб мінімізувати їх деформацію під час експлуатації. Для цього використовувалися ребра жорсткості, кільцеві елементи підсилення та достатня товщина несучих поверхонь, що забезпечують збереження геометрії корпусу навіть при тривалій роботі мобільної платформи.

На рисунку 3.6 наведено загальний вигляд конструктивних елементів корпусу мобільної роботизованої платформи після виготовлення методом тривимірного друку до початку остаточного складання.



Рисунок 3.6 – Конструктивні елементи корпусу мобільної роботизованої платформи після виготовлення

Після виготовлення основних елементів корпусу було виконано поетапне складання нижнього рівня мобільної роботизованої платформи. На цьому рівні передбачено посадкові місця для приводів, опорних вузлів, елементів кріплення, кабельних з’єднань та сенсорних модулів. Етапи формування внутрішнього компонування нижнього рівня платформи наведено на рисунку 3.7.



а) б)

а – корпус до встановлення ходової частини; б – корпус зі змонтованими посадковими та сенсорними вузлами

Рисунок 3.7 – Етапи формування внутрішньої конструкції нижнього рівня мобільної платформи

Як видно з рисунка 3.7, внутрішній простір платформи розділено на окремі функціональні зони. Радіальне розташування силових елементів та використання центральних ребер жорсткості забезпечують рівномірний розподіл механічного навантаження. Передбачені посадкові місця дозволяють установлювати й демонтувати окремі вузли без повного розбирання корпусу.

Після встановлення ходової частини, сенсорних модулів і верхнього монтажного рівня було отримано експериментальний зразок мобільної роботизованої платформи. Загальний вигляд складеного прототипу наведено на рисунку 3.8.

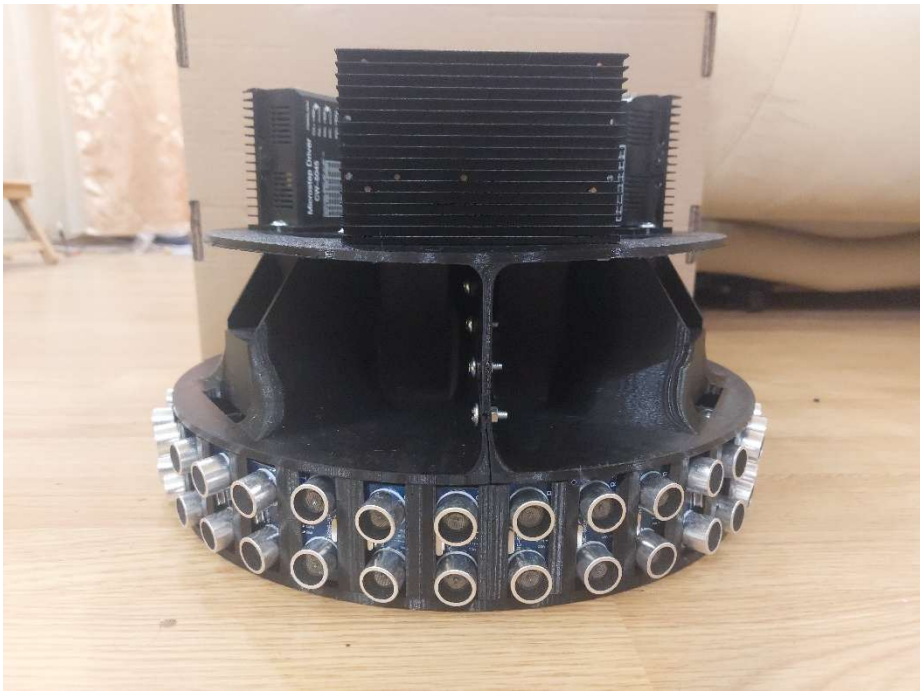


Рисунок 3.8 – Загальний вигляд складеного прототипу мобільної роботизованої платформи

Конструкція має багаторівневе компонування. Нижній рівень призначений для розміщення ходової частини та механічних вузлів, а верхній – для встановлення драйверів, контролерів, перетворювачів живлення та інших електронних модулів. Таке компонування забезпечує зручний доступ до обладнання під час налагодження та технічного обслуговування.

Проведене проектування дозволило сформувати конструкцію, яка поєднує невелику масу, достатню механічну міцність, високу ремонтопридатність та можливість подальшої модернізації. Використання модульного принципу побудови забезпечує просте складання виробу, зручний доступ до внутрішніх компонентів та можливість швидкої заміни окремих вузлів у процесі експлуатації. Розроблена конструкція стала основою для подальшого виготовлення фізичного прототипу мобільної роботизованої платформи, процес якого розглянуто у наступному підрозділі.

4 ЕКСПЕРИМЕНТАЛЬНА ПЕРЕВІРКА ПРАЦЕЗДАТНОСТІ ПРОГРАМНО-АПАРАТНОГО КОМПЛЕКСУ

4.1 Мета та порядок проведення експериментальної перевірки

Після розроблення апаратного забезпечення, програмної структури та конструкції мобільної роботизованої платформи необхідним етапом є експериментальна перевірка працездатності запропонованих технічних рішень. Метою експериментальної перевірки є підтвердження коректності взаємодії між основними модулями програмно-апаратного комплексу, перевірка працездатності центрального модуля керування, локальної сенсорної підсистеми, підсистеми керування виконавчими механізмами, системи інформаційного обміну та підсистеми електроживлення. Експериментальна перевірка проводилася поетапно. На першому етапі виконувалася перевірка працездатності окремих апаратних модулів, зокрема центрального контролера STM32H750VGT6, локальних мікроконтролерів STM32F103, одноплатного комп’ютера Raspberry Pi 5, драйверів крокових двигунів CW-5045 та графічного LCD-дисплея.

На другому етапі перевірялася інформаційна взаємодія між модулями комплексу. На третьому етапі оцінювалася робота сенсорної підсистеми, виконавчих механізмів та системи електроживлення у складі єдиної мобільної роботизованої платформи. Під час перевірки основну увагу приділено стабільності запуску системи, правильності ініціалізації апаратних модулів, коректності обміну даними через інтерфейс SPI, реакції системи на інформацію від датчиків відстані, формуванню сигналів STEP, DIR та EN для драйверів двигунів, а також стабільності електроживлення цифрових і силових модулів. Загальний порядок експериментальної перевірки наведено в таблиці 4.1.

Таблиця 4.1 – Етапи експериментальної перевірки програмно-апаратного комплексу

Етап перевірки	Об’єкт перевірки	Мета перевірки
Перевірка запуску системи	STM32H750VGT6, Raspberry Pi 5, STM32F103	Підтвердження коректної ініціалізації модулів
Перевірка інформаційного обміну	SPI, UART, GPIO	Оцінювання працездатності каналів зв’язку
Перевірка сенсорної підсистеми	STM32F103, датчики відстані	Контроль отримання та передавання даних
Перевірка приводів	CW-5045, крокові двигуни	Оцінювання формування сигналів керування
Перевірка електроживлення	Акумулятор, BMS, DC/DC-перетворювачі	Контроль стабільності живлення модулів
Комплексна перевірка	Усі модулі комплексу	Оцінювання узгодженої роботи системи

Наведений порядок дозволяє послідовно перевірити працездатність окремих вузлів та оцінити можливість їх спільної роботи у складі програмно-апаратного комплексу. Такий підхід дає змогу виявити можливі помилки на рівні окремих підсистем до проведення комплексної перевірки всієї мобільної роботизованої платформи.

4.2 Перевірка роботи центрального модуля керування

Центральний модуль керування на базі мікроконтролера STM32H750VGT6 є основним координатором роботи програмно-апаратного комплексу. Саме він забезпечує приймання команд від Raspberry Pi 5, обмін даними з локальними контролерами STM32F103, формування сигналів керування виконавчими механізмами та контроль поточного стану системи. Тому перевірка працездатності центрального модуля є одним із ключових етапів

експериментальної перевірки всього комплексу.

Під час перевірки центрального модуля контролювалася правильність запуску мікроконтролера, ініціалізація системної тактової частоти, налаштування портів GPIO, запуск апаратних таймерів, підготовка інтерфейсів SPI та UART, а також перехід програмного забезпечення до основного циклу роботи. Окремо перевірялася можливість формування службових повідомлень про стан системи та готовність центрального контролера до обміну даними з іншими модулями комплексу.

Перевірка виконувалася шляхом послідовного запуску програмних модулів центрального контролера та контролю їх реакції на тестові умови. Після подачі живлення мікроконтролер виконував початкову ініціалізацію периферійних пристроїв, перевірку готовності основних інтерфейсів та формував ознаку переходу до робочого режиму. У разі успішного завершення ініціалізації центральний модуль переходив до головного циклу, у якому здійснюється обробка команд, контроль стану підсистем та формування керуючих сигналів.

Особливу увагу приділено перевірці роботи апаратних таймерів STM32H750VGT6, які використовуються для формування імпульсних сигналів STEP. Застосування таймерів дозволяє отримувати стабільні часові інтервали між імпульсами без постійного навантаження на центральне процесорне ядро. Це є важливим для забезпечення точності керування кроковими двигунами та стабільної роботи програмно-апаратного комплексу в режимі реального часу.

Також перевірялася працездатність інтерфейсів SPI та UART. Інтерфейс SPI використовується для обміну даними з Raspberry Pi 5 та локальними контролерами STM32F103, а UART застосовується для передавання службових повідомлень під час налагодження. Наявність стабільного обміну даними через зазначені інтерфейси підтверджує можливість подальшої інтеграції центрального модуля з іншими підсистемами комплексу. Результати перевірки центрального модуля керування наведено в таблиці 4.2.

Таблиця 4.2 – Результати перевірки центрального модуля керування

Перевірюваний параметр	Очікуваний результат	Отриманий результат
Ініціалізація мікроконтролера	Успішний запуск програми	Запуск виконується коректно
Налаштування GPIO	Доступність цифрових входів-виходів	GPIO працюють у заданих режимах
Запуск SPI	Готовність до інформаційного обміну	Інтерфейс ініціалізується коректно
Запуск UART	Передача службових повідомлень	Передача повідомлень виконується
Робота таймерів	Формування імпульсів STEP	Імпульсні сигнали формуються стабільно
Основний цикл програми	Безперервне виконання алгоритму	Програма переходить до робочого режиму

Як видно з таблиці 4.2, центральний модуль керування забезпечує коректну ініціалізацію основних апаратних ресурсів, запуск інтерфейсів обміну даними та формування сигналів, необхідних для керування виконавчими механізмами. Отримані результати підтверджують працездатність STM32H750VGT6 як центрального координатора програмно-апаратного комплексу. Таким чином, перевірка центрального модуля керування показала, що обране технічне рішення забезпечує виконання основних функцій системи: обробку команд, обмін даними з периферійними модулями, формування керуючих сигналів та контроль стану комплексу. Це створює основу для подальшої перевірки інформаційного обміну між Raspberry Pi 5 та STM32H750VGT6.

4.3 Перевірка інформаційного обміну між Raspberry Pi 5 та STM32H750VGT6

Після перевірки працездатності центрального модуля керування наступним етапом експериментальної перевірки є оцінювання інформаційного обміну між одноплатним комп’ютером Raspberry Pi 5 та мікроконтролером STM32H750VGT6. Цей канал зв’язку є одним із ключових у структурі програмно-апаратного комплексу, оскільки саме через нього передаються команди верхнього рівня, службова інформація та дані, необхідні для подальшого керування виконавчими механізмами.

У розробленому комплексі Raspberry Pi 5 виконує функції верхнього рівня керування, а STM32H750VGT6 забезпечує виконання задач реального часу. Для обміну даними між цими модулями використовується інтерфейс SPI, який забезпечує високу швидкість передавання інформації, малі часові затримки та просту організацію взаємодії між ведучим і підлеглим пристроєм. У даній структурі Raspberry Pi 5 працює в режимі Master, а STM32H750VGT6 – у режимі Slave.

Під час перевірки інформаційного обміну контролювалася правильність ініціалізації інтерфейсу SPI, передавання тестових команд від Raspberry Pi 5 до центрального контролера, приймання відповідей від STM32H750VGT6, а також коректність обробки службових повідомлень. Окремо перевірялася стабільність обміну під час багаторазового передавання пакетів даних, що дозволяє оцінити надійність роботи каналу зв’язку в умовах циклічного функціонування системи.

Для перевірки коректності передавання даних використовувалися тестові пакети, що містили ідентифікатор команди, службові параметри та контрольне значення. Після приймання пакета центральний контролер виконував перевірку структури повідомлення, аналізував отриману команду та формував відповідь про результат її обробки. Такий підхід дозволяє перевірити не лише сам факт передавання даних, а й правильність логіки взаємодії між програмними

модулями.

Під час тестування було встановлено, що інтерфейс SPI забезпечує стабільний обмін даними між Raspberry Pi 5 та STM32H750VGT6. Передані команди коректно приймалися центральним контролером, після чого виконувалася їх обробка та формування відповідних службових повідомлень. Це підтверджує можливість використання SPI як основного каналу взаємодії між верхнім і середнім рівнями керування програмно-апаратного комплексу.

Результати перевірки інформаційного обміну між Raspberry Pi 5 та STM32H750VGT6 наведено в таблиці 4.3.

Таблиця 4.3 – Результати перевірки інформаційного обміну між Raspberry Pi 5 та STM32H750VGT6

Перевірюваний параметр	Очікуваний результат	Отриманий результат
Ініціалізація SPI	Готовність інтерфейсу до обміну	Інтерфейс ініціалізується коректно
Режим роботи Raspberry Pi 5	Робота в режимі Master	Режим Master встановлено
Режим роботи STM32H750VGT6	Робота в режимі Slave	Режим Slave встановлено
Передавання тестової команди	Приймання команди контролером	Команда приймається коректно
Формування відповіді	Передавання службового повідомлення	Відповідь формується стабільно
Повторний обмін пакетами	Відсутність збоїв під час циклічного обміну	Обмін виконується без втрати даних

Як видно з таблиці 4.3, інформаційний обмін між Raspberry Pi 5 та STM32H750VGT6 виконується коректно. Інтерфейс SPI забезпечує стабільне

передавання тестових команд, приймання відповідей та підтримку циклічного обміну даними без порушення логіки роботи системи.

Отримані результати підтверджують правильність обраної структури інформаційної взаємодії між верхнім рівнем керування та центральним контролером. Використання Raspberry Pi 5 для формування команд верхнього рівня та STM32H750VGT6 для їх виконання в режимі реального часу дозволяє розділити обчислювальне навантаження між окремими модулями та підвищити стабільність роботи програмно-апаратного комплексу.

Таким чином, перевірка інформаційного обміну підтвердила працездатність каналу зв’язку між Raspberry Pi 5 та STM32H750VGT6. Це створює основу для подальшої перевірки локальної сенсорної підсистеми, яка передає центральному контролеру інформацію про стан навколишнього простору та наявність перешкод.

4.4 Перевірка локальної сенсорної підсистеми

Локальна сенсорна підсистема розробленого комплексу побудована на базі трьох мікроконтролерів STM32F103, кожен із яких відповідає за опитування окремої групи датчиків відстані. Така структура дозволяє розподілити навантаження між декількома локальними модулями та зменшити обсяг задач, що виконуються центральним контролером.

Під час експериментальної перевірки контролювалася правильність ініціалізації локальних мікроконтролерів, працездатність підключених датчиків, коректність отримання вимірювань, формування інформаційних пакетів та передавання результатів до STM32H750VGT6 через SPI. Також перевірялася можливість незалежної роботи кожного локального контролера.

Результати перевірки показали, що використання декількох локальних контролерів дозволяє організувати паралельне опитування датчиків, скоротити час реакції системи на зміну зовнішніх умов та підвищити відмовостійкість

комплексу. У випадку несправності одного локального модуля інші контролери можуть продовжувати передавання інформації до центрального контролера, що дозволяє локалізувати проблему та зменшити її вплив на роботу системи.

Мікроконтролер STM32F103 має достатню продуктивність, кількість GPIO та підтримку інтерфейсу SPI для виконання задач локального збору й попередньої обробки даних від датчиків [15]. Результати перевірки локальної сенсорної підсистеми наведено в таблиці 4.4.

Таблиця 4.4 – Результати перевірки локальної сенсорної підсистеми

Перевірюваний параметр	Очікуваний результат	Отриманий результат
Ініціалізація STM32F103	Успішний запуск локального контролера	Контролери запускаються коректно
Опитування датчиків	Отримання даних від сенсорів	Дані зчитуються локальними модулями
Формування пакета	Передавання структурованої інформації	Пакети формуються коректно
Передавання через SPI	Отримання даних центральним контролером	Дані передаються до STM32H750VGT6
Незалежність модулів	Робота кожного контролера окремо	Локальні модулі працюють незалежно

Таким чином, локальна сенсорна підсистема забезпечує отримання інформації від датчиків відстані, її первинну обробку та передавання результатів до центрального контролера у режимі реального часу.

4.5 Перевірка підсистеми керування виконавчими механізмами

Підсистема керування виконавчими механізмами є однією з основних частин програмно-апаратного комплексу, оскільки вона забезпечує перетворення керуючих команд у механічний рух мобільної роботизованої платформи. У розробленій системі керування кроковими двигунами здійснюється за допомогою драйверів CW-5045, які отримують сигнали STEP, DIR та EN від центрального мікроконтролера STM32H750VGT6.

Під час експериментальної перевірки оцінювалася правильність формування сигналів керування, реакція драйверів на зміну сигналів STEP, DIR та EN, можливість запуску й зупинки двигунів, а також узгодженість роботи декількох приводів. Окремо перевірялося використання апаратних таймерів для формування імпульсів STEP, що дозволяє забезпечити стабільні часові інтервали між імпульсами.

Перевірка показала, що центральний контролер здатний формувати сигнали керування для драйверів двигунів, забезпечувати вибір напрямку обертання, запуск та зупинку виконавчих механізмів. Використання окремих драйверів дозволяє розділити логічну та силову частини системи, що підвищує надійність роботи комплексу та спрощує подальше технічне обслуговування. Результати перевірки підсистеми керування виконавчими механізмами наведено в таблиці 4.5.

Таблиця 4.5 – Результати перевірки керування виконавчими механізмами

Перевірюваний параметр	Очікуваний результат	Отриманий результат
Формування STEP	Генерація імпульсів переміщення	Імпульси формуються стабільно
Формування DIR	Зміна напрямку руху	Напрямок змінюється коректно

Продовження табл. 4.5

Перевірюваний параметр	Очікуваний результат	Отриманий результат
Формування EN	Увімкнення та вимкнення драйвера	Драйвер реагує на сигнал
Робота драйвера CW-5045	Керування крововим двигуном	Двигун виконує команди
Синхронізація приводів	Узгоджена робота декількох двигунів	Синхронізація забезпечується

Таким чином, підсистема керування виконавчими механізмами забезпечує виконання основних команд руху та може використовуватися для подальшої реалізації алгоритмів автоматизованого сортування поштових відправлень.

4.6 Перевірка підсистеми електроживлення

Підсистема електроживлення забезпечує стабільну роботу всіх електронних модулів, виконавчих механізмів та цифрових інтерфейсів програмно-апаратного комплексу. Під час експериментальної перевірки контролювалася працездатність акумуляторного джерела живлення, системи BMS, DC/DC-перетворювачів, головного вимикача, запобіжника та окремих ліній живлення для силових і логічних модулів.

Основною метою перевірки було підтвердження правильності розділення силової та логічної частин системи. Силова частина використовується для живлення драйверів CW-5045 і крокових двигунів, тоді як логічна частина забезпечує стабілізоване живлення мікроконтролерів, Raspberry Pi 5, LCD-дисплея та сенсорних модулів. Такий підхід дозволяє зменшити вплив імпульсних навантажень двигунів на цифрову електроніку.

Під час перевірки встановлено, що запропонована структура електроживлення дозволяє забезпечити окреме живлення силових і логічних модулів, а використання DC/DC-перетворювачів забезпечує необхідні рівні

напруги для електронних компонентів. Наявність BMS та запобіжника підвищує безпечність експлуатації системи і знижує ризик пошкодження електронних модулів у разі аварійної ситуації. Результати перевірки підсистеми електроживлення наведено в таблиці 4.6.

Таблиця 4.6 – Результати перевірки підсистеми електроживлення

Перевірюваний параметр	Очікуваний результат	Отриманий результат
Акумуляторне живлення	Автономна робота комплексу	Живлення подається на систему
BMS	Захист акумуляторної батареї	Захисні функції передбачені структурою
DC/DC-перетворювачі	Формування напруги 5 В та 3,3 В	Цифрові модулі отримують стабілізоване живлення
Силова лінія	Живлення драйверів двигунів	Драйвери підключені окремою лінією
Логічна лінія	Живлення контролерів і Raspberry Pi 5	Логічні модулі живляться окремо
Захист системи	Запобігання пошкодженню модулів	Передбачено вимикач і запобіжник

Таким чином, підсистема електроживлення забезпечує необхідні умови для стабільної роботи програмно-апаратного комплексу та відповідає вимогам автономної експлуатації мобільної роботизованої платформи.

4.7 Оцінювання результатів експериментальної перевірки

Проведена експериментальна перевірка підтвердила працездатність основних технічних рішень, запропонованих у кваліфікаційній роботі. Центральний контролер STM32H750VGT6 забезпечує виконання задач реального часу, організацію інформаційного обміну, формування сигналів керування виконавчими механізмами та контроль стану системи. Одноплатний комп’ютер Raspberry Pi 5 забезпечує виконання функцій верхнього рівня, формування команд керування та взаємодію із центральним контролером.

Локальна сенсорна підсистема на базі трьох мікроконтролерів STM32F103 забезпечує розподілене опитування датчиків відстані та передавання результатів до центрального контролера. Використання такої структури дозволяє зменшити навантаження на STM32H750VGT6 і підвищити масштабованість комплексу.

Підсистема керування виконавчими механізмами на базі драйверів CW-5045 забезпечує формування сигналів STEP, DIR та EN, необхідних для керування кроковими двигунами.

Перевірка підсистеми електроживлення підтвердила доцільність розділення силових і логічних кіл, використання DC/DC-перетворювачів, BMS, запобіжника та окремих ліній живлення для різних груп споживачів. Така структура підвищує стабільність роботи цифрової частини комплексу та зменшує вплив імпульсних навантажень виконавчих механізмів.

Узагальнені результати експериментальної перевірки наведено в таблиці 4.7.

Таблиця 4.7 – Узагальнені результати експериментальної перевірки

Підсистема	Результат перевірки	Висновок
Центральний модуль керування	Ініціалізація та основний цикл працюють коректно	Працездатність підтверджено
Інформаційний обмін	SPI-зв’язок забезпечує передавання команд і даних	Працездатність підтверджено
Локальна сенсорна підсистема	Дані від локальних контролерів передаються до STM32H750VGT6	Працездатність підтверджено
Керування приводами	Сигнали STEP, DIR та EN формуються центральним контролером	Працездатність підтверджено
Підсистема електроживлення	Забезпечено розділення силових і логічних кіл	Працездатність підтверджено
Програмна діагностика	Передбачено контроль помилок і аварійне реагування	Рішення є доцільним

Таким чином, результати експериментальної перевірки підтверджують працездатність розробленого програмно-апаратного комплексу та правильність обраних технічних рішень. Запропонована архітектура забезпечує узгоджену роботу Raspberry Pi 5, STM32H750VGT6, локальних контролерів STM32F103, сенсорної підсистеми, драйверів крокових двигунів та підсистеми електроживлення.

Отримані результати свідчать, що розроблений програмно-апаратний комплекс може бути використаний як основа для подальшої реалізації системи автоматизованого сортування поштових відправлень. Модульна структура комплексу забезпечує можливість подальшого вдосконалення, підключення додаткових сенсорних модулів, модернізації виконавчих механізмів та розширення функціональних можливостей системи.

5 ОХОРОНА ПРАЦІ

5.1 Аналіз умов експлуатації програмно-апаратного комплексу

Розроблений програмно-апаратний комплекс призначений для використання у приміщеннях логістичних або сортувальних центрів, де виконуються операції автоматизованого транспортування та сортування поштових відправлень. Під час експлуатації комплексу передбачається робота мобільної роботизованої платформи, електронних модулів керування, сенсорної підсистеми, виконавчих механізмів, акумуляторної батареї та засобів інформаційного обміну [39].

Умови експлуатації комплексу характеризуються наявністю низьковольтної цифрової електроніки, силових ліній живлення драйверів крокових двигунів, рухомих механічних частин, акумуляторного джерела живлення та програмних засобів керування. Тому під час розроблення й подальшого використання системи необхідно враховувати вимоги електробезпеки, механічної безпеки, пожежної безпеки, а також безпечної взаємодії оператора з мобільною роботизованою платформою.

До складу цифрової частини комплексу входять мікроконтролери STM32H750VGT6 і STM32F103, одноплатний комп’ютер Raspberry Pi 5, LCD-дисплей та сенсорні модулі. Силова частина містить драйвери крокових двигунів CW-5045, електродвигуни, акумуляторну батарею, систему BMS, DC/DC-перетворювачі, головний вимикач і запобіжник. Поєднання цифрових і силових модулів потребує правильного розділення електричних кіл, надійної ізоляції з’єднань, захисту від короткого замикання та контролю технічного стану обладнання [40].

Під час експлуатації оператор може виконувати запуск комплексу, контроль його стану, технічне обслуговування, заряджання акумуляторної

батареї, перевірку підключення модулів, налагодження програмного забезпечення та аварійне вимкнення системи. Усі зазначені операції повинні виконуватися з урахуванням вимог охорони праці та безпечної експлуатації електронного обладнання [40].

Таким чином, основними напрямами забезпечення безпеки під час роботи програмно-апаратного комплексу є електробезпека, захист від рухомих частин, запобігання перегріву й короткому замиканню, контроль стану акумуляторної батареї та забезпечення можливості швидкого аварійного відключення системи.

5.2 Небезпечні та шкідливі фактори під час експлуатації комплексу

Під час експлуатації програмно-апаратного комплексу можуть виникати небезпечні та шкідливі фактори, пов’язані з роботою електронних модулів, акумуляторного джерела живлення, рухомих механічних вузлів та засобів керування. Основні фактори ризику пов’язані з можливістю ураження електричним струмом у разі пошкодження ізоляції, короткого замикання в електричних колах, перегріву силових елементів, механічного контакту з рухомими частинами мобільної платформи та помилок під час технічного обслуговування [41].

Окрему увагу необхідно приділяти акумуляторній батареї, оскільки вона є основним джерелом живлення комплексу. Порушення правил її експлуатації може призвести до перегріву, глибокого розряду, перевантаження або пошкодження елементів. Для зниження таких ризиків у структурі комплексу передбачено використання системи керування батареєю BMS, запобіжника та головного вимикача живлення.

Наявність крокових двигунів і драйверів CW-5045 створює додаткові вимоги до безпеки. Під час роботи виконавчих механізмів можливе виникнення руху платформи, обертання коліс, різких змін напрямку переміщення або зупинки. Тому під час налагодження та випробувань необхідно виключати

перебування рук, інструментів або сторонніх предметів у зоні рухомих частин. Основні небезпечні фактори та заходи їх зменшення наведено в таблиці 5.1.

Таблиця 5.1 – Небезпечні фактори під час експлуатації програмно-апаратного комплексу [42]

Небезпечний фактор	Можлива причина виникнення	Заходи запобігання
Коротке замикання	Пошкодження ізоляції або неправильне підключення модулів	Використання запобіжника, перевірка з’єднань та ізоляція контактів
Перегрів електронних компонентів	Тривала робота драйверів або DC/DC-перетворювачів під навантаженням	Контроль температури, забезпечення вентиляції та недопущення перевантаження
Нестабільне живлення цифрових модулів	Просідання напруги під час роботи двигунів	Розділення силових і логічних кіл, використання DC/DC-перетворювачів
Пошкодження акумуляторної батареї	Перевантаження, глибокий розряд або неправильне заряджання	Використання BMS, контроль режимів заряджання та розряджання
Механічне травмування	Контакт із рухомими частинами платформи	Обмеження доступу до рухомих вузлів під час роботи та застосування аварійної зупинки
Помилки під час налагодження	Робота з відкритим корпусом або підключеними силовими лініями	Вимкнення живлення перед обслуговуванням і перевірка правильності монтажу

Як видно з таблиці 5.1, основні небезпечні фактори пов’язані з електричними колами, акумуляторним живленням та рухомими механічними вузлами. Їх вплив можна зменшити завдяки правильному проєктуванню системи живлення, використанню засобів захисту, контролю технічного стану обладнання та дотриманню правил безпечної експлуатації [43].

5.3 Заходи безпечної експлуатації програмно-апаратного комплексу

Для забезпечення безпечної експлуатації програмно-апаратного комплексу необхідно передбачити комплекс організаційних і технічних заходів. До організаційних заходів належать інструктаж користувача перед початком роботи, перевірка справності обладнання, дотримання порядку ввімкнення й вимкнення комплексу, заборона роботи з пошкодженими кабелями або відкритими силовими контактами, а також виконання технічного обслуговування лише після повного відключення живлення.

До технічних заходів належить використання головного вимикача живлення, плавкого запобіжника, системи BMS, окремих ліній живлення для силових і логічних модулів, стабілізованих DC/DC-перетворювачів, ізоляції електричних з’єднань та механічного кріплення всіх елементів конструкції. Такі рішення дозволяють зменшити ризик короткого замикання, перегріву, пошкодження електронних модулів та неконтрольованої роботи виконавчих механізмів [42].

Перед початком роботи необхідно перевірити надійність кріплення акумуляторної батареї, стан кабельних з’єднань, відсутність механічних пошкоджень корпусу, справність головного вимикача та готовність програмного забезпечення до запуску. Після подачі живлення система повинна виконати початкову ініціалізацію, перевірити доступність основних модулів і лише після цього переходити до робочого режиму.

Під час налагодження комплексу необхідно уникати дотику до відкритих електричних контактів, не виконувати підключення або відключення модулів під напругою, не залишати платформу без нагляду під час тестового руху та забезпечувати достатній вільний простір навколо неї. У разі виникнення нестандартної ситуації оператор повинен мати можливість швидко вимкнути живлення за допомогою головного вимикача [43].

Особливу увагу слід приділяти безпеці руху мобільної роботизованої

платформи. Під час роботи комплексу необхідно забезпечити контроль робочої зони, виключити перебування сторонніх предметів у зоні переміщення платформи та передбачити програмну реакцію на появу перешкод. Сенсорна підсистема повинна забезпечувати своєчасне виявлення об’єктів поблизу платформи, а програмне забезпечення – зупинку або зміну режиму роботи у разі небезпечної ситуації.

Для зниження ризику пожежної небезпеки необхідно контролювати стан акумуляторної батареї, не допускати її механічного пошкодження, перегріву або експлуатації за наявності ознак несправності. Заряджання акумулятора повинно виконуватися з використанням відповідного зарядного пристрою та під контролем оператора. У разі появи запаху перегріву, диму, іскріння або нестабільної роботи електронних модулів комплекс необхідно негайно знеструмити [44].

Узагальнені заходи безпечної експлуатації програмно-апаратного комплексу наведено в таблиці 5.2.

Таблиця 5.2 – Заходи безпечної експлуатації програмно-апаратного комплексу

Напрям безпеки	Основні заходи
Електробезпека	Використання запобіжника, головного вимикача, ізоляції контактів та перевірка кабельних з’єднань
Безпека акумуляторної батареї	Застосування BMS, контроль заряджання, недопущення перегріву та механічного пошкодження
Механічна безпека	Обмеження доступу до рухомих частин, контроль робочої зони та надійне кріплення вузлів
Безпека під час налагодження	Вимкнення живлення перед обслуговуванням та заборона перемикання модулів під напругою
Пожежна безпека	Контроль температури, недопущення короткого замикання та знеструмлення системи в разі аварійної ситуації
Програмна безпека	Діагностика модулів, контроль помилок, аварійна зупинка та перехід у безпечний режим

Наведені заходи дозволяють знизити ризик виникнення аварійних ситуацій, підвищити надійність експлуатації програмно-апаратного комплексу та забезпечити безпечну роботу оператора під час запуску, налагодження й технічного обслуговування системи [44].

5.4 Висновки до розділу

У даному розділі розглянуто питання охорони праці під час експлуатації програмно-апаратного комплексу для сортування поштових відправлень. Визначено основні умови роботи комплексу, пов’язані з використанням електронних модулів керування, акумуляторної батареї, DC/DC-перетворювачів, драйверів крокових двигунів, сенсорної підсистеми та рухомої мобільної роботизованої платформи.

Проаналізовано основні небезпечні фактори, що можуть виникати під час експлуатації комплексу, зокрема коротке замикання, перегрів електронних компонентів, пошкодження акумуляторної батареї, нестабільність живлення цифрових модулів, механічний контакт із рухомими частинами та помилки під час налагодження. Для зменшення впливу зазначених факторів запропоновано організаційні й технічні заходи безпечної експлуатації.

Запропоновані заходи передбачають використання головного вимикача живлення, запобіжника, системи BMS, окремих ліній живлення для силових і логічних модулів, стабілізованих DC/DC-перетворювачів, програмної діагностики, контролю робочої зони та можливості аварійного відключення комплексу. Реалізація таких заходів дозволяє підвищити безпечність експлуатації мобільної роботизованої платформи, зменшити ризик пошкодження електронних компонентів і забезпечити надійну роботу програмно-апаратного комплексу.

ВИСНОВКИ

У кваліфікаційній роботі вирішено задачу розроблення апаратного забезпечення програмно-апаратного комплексу для сортування поштових відправлень із використанням мобільної роботизованої платформи. У процесі виконання роботи було проведено аналіз сучасних систем автоматизованого сортування, мобільних роботизованих платформ та елементної бази, що використовується у сучасних робототехнічних і комп'ютерно-інтегрованих системах.

Проведений аналіз показав, що використання мобільних роботизованих платформ є перспективним напрямом автоматизації логістичних процесів. На відміну від стаціонарних конвеєрних систем, такі платформи забезпечують більшу гнучкість, можливість зміни маршрутів руху, масштабування системи та адаптацію до різних умов експлуатації. За результатами аналізу було обґрунтовано доцільність використання модульної багаторівневої архітектури програмно-апаратного комплексу.

У роботі розроблено загальну структуру програмно-апаратного комплексу, яка передбачає розподіл функцій між верхнім рівнем керування на базі Raspberry Pi 5, центральним контролером STM32H750VGT6, локальними мікроконтролерами STM32F103, сенсорною підсистемою, драйверами крокових двигунів CW-5045 та підсистемою електроживлення. Такий розподіл дозволяє відокремити задачі високорівневої обробки інформації від задач реального часу, зменшити навантаження на центральний контролер та підвищити загальну надійність роботи комплексу.

Розроблено центральний модуль керування на базі мікроконтролера STM32H750VGT6. Обґрунтовано його використання як основного координатора системи, що забезпечує обмін даними з Raspberry Pi 5, взаємодію з локальними контролерами STM32F103, формування сигналів STEP, DIR та EN для драйверів

CW-5045, а також передавання службової інформації на LCD-дисплей. Запропонована структура центрального модуля забезпечує виконання задач реального часу та дозволяє масштабувати систему без суттєвої зміни її архітектури.

Розроблено локальну сенсорну підсистему, побудовану на трьох мікроконтролерах STM32F103. Кожний локальний контролер виконує опитування окремої групи датчиків відстані, первинну обробку отриманої інформації та передавання результатів до центрального контролера через інтерфейс SPI. Такий підхід забезпечує паралельну обробку даних, скорочує час реакції системи на зміну зовнішніх умов та підвищує відмовостійкість комплексу.

Розроблено підсистему керування виконавчими механізмами на базі промислових драйверів крокових двигунів CW-5045. Керування двигунами здійснюється за допомогою цифрових сигналів STEP, DIR та EN, що формуються центральним контролером. Використання окремих драйверів дозволяє відокремити силову частину системи від логічних кіл керування, забезпечити необхідний рівень струму для обмоток двигунів, підвищити точність позиціонування та спростити модернізацію виконавчих механізмів.

Розроблено підсистему електроживлення програмно-апаратного комплексу. Запропонована структура передбачає використання акумуляторної батареї, системи керування батареєю BMS, DC/DC-перетворювачів, головного вимикача, запобіжника та окремих ліній живлення для силових і логічних модулів. Розділення силової та цифрової частин дозволяє зменшити вплив електромагнітних завад, підвищити стабільність роботи електронних компонентів та забезпечити безпечну автономну експлуатацію мобільної роботизованої платформи.

Розроблено систему інформаційної взаємодії між апаратними модулями комплексу. Основним інтерфейсом обміну даними обрано SPI, який використовується для взаємодії Raspberry Pi 5 із центральним контролером

STM32H750VGT6 та центрального контролера з локальними модулями STM32F103. Для керування виконавчими механізмами застосовано сигнали GPIO, а для сервісного налагодження передбачено використання UART. Така структура забезпечує високу швидкість передавання даних, мінімальні затримки та можливість подальшого розширення комплексу.

У роботі також розроблено програмне забезпечення основних модулів програмно-апаратного комплексу. Запропоновано багаторівневу структуру програмного забезпечення, у якій Raspberry Pi 5 виконує задачі верхнього рівня, пов’язані з комп’ютерним зором, прийняттям рішень та формуванням команд, а STM32H750VGT6 забезпечує виконання задач реального часу, керування приводами, обмін даними з локальними контролерами та контроль стану системи. Для підвищення надійності передбачено засоби діагностики, перевірки інформаційного обміну, контролю периферійних модулів та аварійного реагування.

Розроблено конструкцію мобільної роботизованої платформи з урахуванням вимог до розміщення електронних модулів, виконавчих механізмів, сенсорної підсистеми та елементів живлення. Запропонована конструкція має модульну будову, забезпечує зручність складання, доступ до внутрішніх компонентів, можливість подальшої модернізації та виготовлення окремих елементів методом тривимірного друку.

Практичне значення отриманих результатів полягає у створенні апаратної та програмної основи програмно-апаратного комплексу, який може бути використаний для автоматизації процесів сортування та транспортування поштових відправлень. Запропонована архітектура є масштабованою, модульною та придатною для подальшого розвитку, модернізації й адаптації до різних умов експлуатації у складі сучасних логістичних систем.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. ДСТУ 3008:2015. Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлювання. Київ : ДП «УкрНДНЦ», 2016. 31 с.

2. ДСТУ 8302:2015. Інформація та документація. Бібліографічне посилання. Загальні положення та правила складання. Чинний від 2016-07-01. Київ : УкрНДНЦ, 2016. 16 с.

3. Методичні вказівки з підготовки кваліфікаційної роботи для здобувачів першого (бакалаврського) рівня вищої освіти денної і заочної форм навчання спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології» освітньої програми «Автоматизація та комп'ютерно-інтегровані технології» / упоряд.: І. Ш. Невлюдов, О. І. Филипенко, О. В. Токарєва, С. П. Новоселов, О. В. Сичова. Харків : ХНУРЕ, 2023. 64 с.

4. Навчальний посібник з підготовки кваліфікаційної роботи бакалавра для здобувачів вищої освіти денної і заочної форм навчання спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології» освітньої програми «Автоматизація та комп'ютерно-інтегровані технології» :навч. посіб. / І. Ш. Невлюдов, О. І. Филипенко, О. В. Токарєва, С. П. Новоселов, О. В. Сичова. Харків : Видавництво Іванченка І. С., 2023. 151 с.

5. Оформлення технічної документації : навч. посіб. / В. В. Семенець, І. Ш. Невлюдов, А. М. Сінотін, С. В. Сотник ; М-во освіти і науки України, Харків. нац. ун-т радіоелектроніки. Харків : ХНУРЕ, 2021. 148 с.

6. Невлюдов І. Ш. Проектування мобільних роботів на базі одноплатних комп'ютерів (Raspberry Pi и мови Python 3.6) : підручник / І. Ш. Невлюдов, А. О. Андрусевич, В. В. Євсєєв. Харків, 2020. 257 с.

7. Виробничі процеси та обладнання об'єктів автоматизації. Збірник задач : навч. посіб. / І. Ш. Невлюдов, А. О. Андрусевич, Г. В. Пономарьова, А. О.

Функендорф. Кривий Ріг : Криворізький коледж НАУ, 2018. 332 с.

8. Nova Poshta opens first phase of KIT parcel sorting center in Kyiv // Interfax-Ukraine. 2018. URL: https://en.interfax.com.ua/news/general/489693.html (дата звернення: 11.07.2026).

9. Дебелий Я., Петрушко Л. Термінал «Нової пошти» сортує до 50 000 посилок на годину. Що відбувається з посилкою у надрах найсучаснішого логістичного центру України // Forbes Ukraine. 23 квітня 2021. URL: https://forbes.ua/company/nova-poshta-terminal-16042021-1387 (дата звернення: 11.07.2026).

10. Yan, B., Yao, B., & Zhang, C. (2023). Industrial structure, high-quality development of logistics industry and the economy. Plos one, 18(5), e0285229.

11 Невлюдов І. Ш. Автоматичне управління технологічними об’єктами : підручник / І. Ш. Невлюдов, О. В. Токарєва. Харків : ХНУРЕ, 2018. 190 с.

12. Невлюдов І. Ш. BEAM робототехніка : навч. посіб. / І. Ш. Невлюдов, В. В. Євсєєв, С. С. Максимова ; Харків. нац. ун-т радіоелектроніки, кафедра комп’ютерно-інтегрованих технологій, автоматизації та робототехніки. Кривий Ріг : Видавець Чернявський Д. О., 2024. 276 с. ISBN 978-617-8045-79-1.

13. Dresser, S., & Robotics, V. A. (2023). Amazon announces 2 new ways it’s using robots to assist employees and deliver for customers. About Amazon (Amazon Robotics blog).

14. Ciabatta, G. (2023). Equity Research on Ocado Group-Fee Based Long Term Value (Master's thesis, Universidade NOVA de Lisboa (Portugal)).

15. Prymachenko, H., & Hryhorova, Y. (2020). Research status of automation of logistics transport and distribution processes. Intellectualization of logistics and Supply Chain Management, (1), 39-50.

16. Гудзь, О. Є., & Берднікова, С. М. (2022). Модернізація організаційного контуру підприємств в нових економічних реаліях. Економіка. Менеджмент. Бізнес, (3-4), 4-11.

17. Цимбал О. М. Системи адаптації роботів і технологія OpenCV : навч.

посіб. / О. М. Цимбал, А. І. Бронніков ; Харків. нац. ун-т радіоелектроніки. Харків : ХНУРЕ, 2019. 144 с.

18. Nevliudov I., Yevsieiev V., Maksymova S., Demska N., Kolesnyk K., Miliutina O. Mobile Robot Navigation System Based on Ultrasonic Sensors // 2023 IEEE XXVIII International Seminar/Workshop on Direct and Inverse Problems of Electromagnetic and Acoustic Wave Theory (DIPED). 2023. Vol. 1. P. 247–251.

19. Chuchico, C., del Carmen Claudio, M., & Chicaiza, F. (2025). A Comprehensive Review of Four-Wheel Independent Drive and Steering Systems: Integrating Double Ackermann Kinematics, Advanced Trajectory Tracking, and Active Safety. International Journal of Engineering Insights, 3(1), 13-22.

20. Cao, R., Gu, J., Yu, C., & Rosendo, A. (2022, October). Omniwheg: An omnidirectional wheel-leg transformable robot. In 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (pp. 5626-5631). IEEE.

21. Burkacki, M., Łysy, I., Suchoń, S., Chrzan, M., & Kowolik, R. (2025). Systematic review of Mecanum and omni wheel technologies for motor impairments. Applied Sciences, 15(9), 4773.

22. Проектування мобільних маніпуляційних роботів : монографія / І. Ш. Невлюдов, А. О. Андрусевич, В. В. Євсєєв та ін. ; М-во освіти і науки України, Харків. нац. ун-т радіоелектроніки. – Харків : ХНУРЕ, 2022. 427 с.

23. Технічне та програмне забезпечення розробки малогабаритного мобільного робота: монографія / І.Ш. Невлюдов, В.В. Євсєєв, Д.В. Гурін. Кривий Ріг: Криворізький фаховий коледж Державного некомерційного підприємства «Державний університет «Київський авіаційний інститут»,2025. –355с.

24. STMicroelectronics. STM32H750VB, STM32H750ZB, STM32H750IB, STM32H750XB: 32-bit Arm Cortex-M7 480 MHz MCUs. Datasheet.URL: https://www.st.com/resource/en/datasheet/stm32h750vb.pdf (дата звернення: 11.07.2026).

25. STMicroelectronics. STM32F103x8, STM32F103xB: Medium-Density Performance Line Arm-Based 32-bit MCU. Datasheet. URL:

https://www.st.com/resource/en/datasheet/stm32f103c8.pdf (дата звернення: 11.07.2026).

26. STMicroelectronics. STM32Cube MCU Package for STM32H7 Series. URL: https://www.st.com/en/embedded-software/stm32cubeh7.html (дата звернення: 11.07.2026).

27. STMicroelectronics. STM32Cube MCU Package for STM32F1 Series. URL: https://www.st.com/en/embedded-software/stm32cubef1.html (дата звернення: 11.07.2026).

28. Raspberry Pi Ltd. Raspberry Pi 5 Product Brief. URL: https://datasheets.raspberrypi.com/rpi5/raspberry-pi-5-product-brief.pdf (дата звернення: 11.07.2026).

29. Raspberry Pi Ltd. Raspberry Pi Documentation. URL: https://www.raspberrypi.com/documentation/ (дата звернення: 11.07.2026).

30. Karumbunathan, L. S. (2022). Nvidia jetson agx orin series. A Giant leap forward for robotics and edge AI applications. Technical Brief.

31. Mubarik, M., Raza, A., Rashid, H., Zainuddin, A., & Abd Latiff, Z. I. (2022, August). Automatic Monitoring and Control System for Different Water Levels in Interconnected Multiple Twin Tanks with Dry Run Protection Using Microprocessor Orange Pi. In 2022 IEEE Symposium on Wireless Technology & Applications (ISWTA) (pp. 47-52). IEEE.

32. Saab, W., Racioppo, P., & Ben-Tzvi, P. (2019). A review of coupling mechanism designs for modular reconfigurable robots. Robotica, 37(2), 378-403.

33. Gabbar, H. A., Othman, A. M., & Abdussami, M. R. (2021). Review of battery management systems (BMS) development and industrial standards. Technologies, 9(2), 28.

34. Murphy R. R. Introduction to AI Robotics. 2nd ed. Cambridge : MIT Press, 2019. 648 p.

35. CW-5045 2-Phase Microstepping Motor Driver. User Manual. URL: https://cnc4you.co.uk/resources/CW5045.pdf (дата звернення: 14.07.2026).

36. Siciliano B., Khatib O. Springer Handbook of Robotics. 2nd ed. Berlin : Springer, 2016. 2227 p. 14. Corke P. Robotics, Vision and Control: Fundamental Algorithms in Python. 3rd ed. Cham : Springer, 2023. 833 p.

37. David, R., Duke, J., Jain, A., Janapa Reddi, V., Jeffries, N., Li, J., ... & Rhodes, R. (2021). Tensorflow lite micro: Embedded machine learning for tinyml systems. Proceedings of machine learning and systems, 3, 800-811.

38. Alam, N., Sagar, A. S., Zhang, W., Jin, T., Dosset, A., Dang, L. M., & Moon, H. (2025). A comprehensive study on enhanced QR extraction techniques with deep learning-based verification: N. Alam et al. Applied intelligence, 55(7), 676.

39. ISO 3691-4:2023. Industrial Trucks – Safety Requirements and Verification – Part 4: Driverless Industrial Trucks and Their Systems. Geneva : International Organization for Standardization, 2023.

40. ISO 12100:2010. Safety of Machinery – General Principles for Design – Risk Assessment and Risk Reduction. Geneva : International Organization for Standardization, 2010.

41. ISO 13849-1:2023. Safety of Machinery – Safety-Related Parts of Control Systems – Part 1: General Principles for Design. Geneva : International Organization for Standardization, 2023.

42. ДСТУ EN 60204-1:2019. Безпечність машин. Електричне устаткування машин. Частина 1. Загальні вимоги (EN 60204-1:2018, IDT; IEC 60204-1:2016, MOD).

43. ДСанПіН 3.3.2.007-98. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин.

44 Комплекс навчально-методичного забезпечення навчальної дисципліни «Безпека праці в індустрії ІТ-технологій» підготовки освітнього рівня бакалавр усіх спеціальностей та усіх напрямів університету / ХНУРЕ ; розроб.: Т. Є. Стиценко, Г. В. Пронюк, Н. М. Сердюк. Харків, 2017. 122 с.