

Міністерство освіти і науки України
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Факультет систем управління літальних апаратів

Кафедра систем управління літальних апаратів

Пояснювальна записка до дипломного проекту

_____ магістра
(освітній ступінь)

на тему «Розробка системи управління мобільної колісної платформи та дослідження режимів керованого заносу»

XAI.301.362.24B.151.00193072 ПЗ

Виконав: студент 2 курсу, групи 362,
Галузь знань

15 “Автоматизація та
приладобудування”
Спеціальність 151 “Автоматизація та
комп’ютерно-інтегровані технології”

Освітня програма
«Інженерія мобільних додатків»

_____ Бондаренко В. С.
(прізвище та ініціали)

Керівник _____ Гавриленко О. В.
(прізвище та ініціали)

Рецензент _____ Ковтун І.В.
(прізвище та ініціали)

Харків – 2024

Міністерство освіти і науки України
 Національний аерокосмічний університет ім. М. Є. Жуковського
 «Харківський авіаційний інститут»

Факультет систем управління літальних апаратів

Кафедра систем управління літальних апаратів (№301)

Рівень вищої освіти другий (магістерський)

Галузь знань 15 Автоматизація та приладобудування

(шифр і назва)

Спеціальність 151 Автоматизація та комп'ютерно-інтегровані технології

(шифр і назва)

Освітня програма Інженерія мобільних додатків

(назва)

ЗАТВЕРДЖУЮ
Завідувач кафедри
 к.т.н., доц.

Костянтин ДЕРГАЧОВ
 “ ” 2024 року

З А В Д А Н Н Я
НА ДИПЛОМНУ РОБОТУ ЗДОБУВАЧУ

Бондаренко Владислав Сергійович

(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи «Розробка системи управління мобільної колісної платформи та дослідження режимів керованого заносу»

керівник кваліфікаційної роботи доц., Гавриленко Олена Володимирівна

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

затверджені наказом вищого навчального закладу від 06.11.2023 року

№ 1968-уч

2. Строк подання студентом роботи: 08 січня 2024 року

3. Вихідні дані до роботи Мобільний додаток, для управління та відстеження параметрів самохідного пристрою, вдосконалення самохідного пристрою, що відповідає технічному завданню.

4. Зміст пояснювальної записки (перелік завдань, які потрібно розробити): Вступ, аналіз методів та засобів розв'язання задачі, описання предметної області, описання функціональної схеми системи, описання математичної моделі, опис експериментальної системи, розрахування технологічної збірки друкованої плати, розрахунок вартості розробки мобільної системи керування роботом, заключення, список використаних джерел

5. Перелік графічного матеріалу: 10 слайдів (еквівалент формату А1)

6. Консультанти розділів кваліфікаційної роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Стан проблеми	доц. Гавриленко О. В.	12.09.23	08.01.24
Аналіз і синтез СУ	доц. Гавриленко О. В.	12.09.23	08.01.24
Конструктор. частина	доц. Гавриленко О. В.	12.09.23	08.01.24
Технологічна частина	доц. Гавриленко О. В.	12.09.23	08.01.24
Експ.-практ. частина	доц. Гавриленко О. В.	12.09.23	08.01.24
Економічне обґрунтування	доц. Гавриленко О. В.	12.09.23	08.01.24

Нормоконтроль _____ доц. Гавриленко О. В. «17» січня 2024 р.
 (підпис) (ініціали та прізвище)

7. Дата видачі завдання 12.09.2023 р

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів кваліфікаційної роботи	Примітка
1.	Формулювання теми проекту.	01.09.2023	
2.	Розробка технічного завдання	12.09.2023	
3.	Огляд стану проблеми і патентний пошук. Математичний опис системи управління.	20.10.2023	
4.	Аналіз і синтез системи управління	15.12.2023	
5.	Конструкторська частина роботи. Технологічна частина роботи. Економічне обґрунтування розробки	08.01.2024	
6.	Експериментально-практична частина	08.01.2024	
7.	Оформлення розрахунково-пояснювальної записки і графічного матеріалу. Попередній захист роботи	12.01.2024	
	Рецензування проекту. Захист проекту в ЕК	18.09.2024	

Здобувач _____ Владислав БОНДАРЕНКО
 (підпис) (прізвище та ініціали)

Керівник роботи _____ Олена ГАВРИЛЕНКО
 (підпис) (прізвище та ініціали)

Міністерство освіти і науки України
Національний аерокосмічний університет ім. М. Є. Жуковського
“Харківський авіаційний інститут”

Кафедра 301

«ЗАТВЕРДЖУЮ»
Завідувач кафедри
к. т. н., доцент

_____ Костянтин ДЕРГАЧОВ

«__» _____ 2024 г.

ТЕХНІЧНЕ ЗАВДАННЯ
на дипломний проект бакалавра

_____ Бондаренко Владислав Сергійович
(П.І.Б.)

1. Тема роботи: «Розробка системи управління мобільної колісної платформи та дослідження режимів керованого заносу»
2. Термін здачі студентом закінченої роботи _____
3. Область застосування розробки: Автомобільна промисловість, військовий комплекс, сільсько-будівниче господарство, геодезія, картографія
4. Початкові данні для об'єкту управління
 - 4.1. Початкова математична модель об'єкта управління (див. додаток до технічного завдання)
 - 4.2. Збурювальні впливи: удар від зіткнення з перешкодами, на маршруті до заданого місцезнаходження, несприятливий рельєф місцевості для проїзду мобільного _____ робота.

5. Технічні вимоги до системи управління

5.1. Питання, які підглядають розгляду

5.1.1 У розрахунковій частині: формування лінеаризованої математичної моделі об'єкта стабілізації, вибір виконавчого, та виміррювального пристроїв, розробка функціональної і структурної схем, системи автоматичної

стабілізації, вибір типового закону управління, синтез пристрою автоматичної стабілізації та аналіз якості управління.

5.1.2 У експериментальній частині: провести експериментальні випробування, моделі мобільного робота за допомогою мобільного додатку.

5.1.3 В конструкторській частині: скласти конструкцію рухомого робота, реалізувати управління через мобільний додаток, сформулювати функціональну схему системи управління на базі мікроконтролера, розробити алгоритм функціонування системи мобільного робота.

5.2. Режими роботи системи управління (автоматичний, напівавтоматичний, ручний, налагоджувальний, і т. п.) : автоматичний, напівавтоматичний, ручний, налагоджувальний.

5.3. Показники якості в роботі системи: можливість безпомилкового пересування в обмеженому просторі, дотримання заданого курсу.

6. Умови експлуатації системи.

6.1. Кліматичні вимоги (температура середовища, у котрій працюватиме система управління, вологість середовища, склад хімічних активних компонентів і т. п.) температура середовища $-20...+50$ градусів за Цельсієм.

6.2. Механічні вимоги (вібрація, тряска, можливі перекоси, удари, нахили і т. д.) відсутній високий рівень вібрації, ударів, нахилів.

6.3. Наявність перешкод (електричні наводки, радіоперешкоди, магнітні впливи) можна знехтувати.

6.4. Електричні параметри системи (напруга джерел живлення, потужність, стабільність, частота) напруга живлення 5 В

6.5. Характер роботи системи (неперервний, циклічний, одноразової дії) неперервний.

7. Додаткові функції, що реалізуються системою управління (сигналізація про несправності, реєстрація необхідної інформації і т.д.) Автоматичне управління за допомогою датчиків, управління людиною, сигналізація про температуру, вологу, відстань до об'єкту, свого положення відносно нульової осі, про несправності.

8. Об'єм виконаних розробником робіт

8.1. Етапи виконання роботи: Оцінка стану проблеми, аналіз і синтез системи, експериментальна частина, конструкторська частина, технологічна частина, економічна частина.

8.2 Обсяг розробки за кожним етапом: технічне завдання (4 стор.), анотація (1 стор.), список умовних скорочень (1 стор.), зміст (2 стор.), аналіз і синтез системи (20 – 25 стор.), конструкторська частина (10 стор.), розробка технологічного процесу (5 – 7 стор.), експериментальна частина (5 – 7 стор.), економічна частина (9 стор.), висновки (1 стор.), список використаних джерел (2 стор.).

9. Параметри обладнання системи

9.1. Габарити розміри визначаються у процесі проектування

9.2. Маса 1502 г

9.3. Вимоги до конструктивного виконання і розміщенню Блок управління двигуном, блок мікроконтролера і акумуляторний відсік встановлені на мобільній платформі; платформа приводиться в рух чотирма двигунами постійного струму, платформа встановлена на чотирьох колесах, розташованих на двох осях.

9.4. Інші вимоги живлення має забезпечуватися акумулятором, підтримка технології Bluetooth на блоці управління.

10. Вимоги безпеки Не задані.

11. Технологічна частина

11.1. Розробка технологічного процесу друкована плата керуючого обчислювача

11.2. Умови і вимоги середній час виробництва друкованого вузла – не більше 7 годин.

11.3. Очікуваний результат створення повністю працездатного та повнофункціонального виробу для використання у потрібних сферах.

12. Економічна частина

12.1. Розробити (розрахувати, отримати): розрахунок собівартості та ціни

управляючого обчислювача.

12.2. Умови і вимоги: при виборі ЕРЕ необхідно враховувати можливості пониження собівартості розробки та виготовлення виробу.

12.3. Очікуваний результат: розрахунок собівартості виробу

13. Перелік графічних матеріалів та їх формат 3 плакати формату А1:

1) опис об'єкту автоматичного управління, математична модель об'єкту, результат дослідження об'єкту.

2) структурна схема САС, функціональна схема САС, синтез коригуючого елемента, дослідження скоригованої САС.

3) конструкторська частина, технологічна частина, економічна частина, експериментальне дослідження САС.

14. Мова підготовки пояснювальної записки (захисту) українська

Керівник роботи

доц. Гавриленко О. В.
(П. І. Б.)

« » _____ 2023 р.

Прийняв до виконання

Бондаренко В. С.
(П. І. Б. студента)

« » _ _____ 2023 г.

Узгоджено за питаннями:

конструкції

доц. Гавриленко О. В.
(П. І. Б.)

« » _____ 2023 г.

дослідницької частини

доц. Гавриленко О. В.
(П. І. Б.)

« » _____ 2023 г.

економіки

доц. Гавриленко О. В.
(П. І. Б.)

« » _____ 2023 г.

РЕФЕРАТ

Пояснювальна записка містить: 102 ст. 47 рис. 16 табл. 3 додатку 5 джерел.

Патенти знаходяться у дод. А. Лістинг коду мобільного додатку та малого самохідного пристрою знаходиться у дод. Б. Фото роботи мобільного додатку та самохідного пристрою знаходиться у дод. В .

Тема роботи: «Розробка системи управління мобільної колісної платформи та дослідження режимів керованого заносу»

Мета роботи: розробити систему управління мобільної колісної платформи та дослідити режими керованого заносу.

Предмет роботи: система управління МСП, що виконує задачу автоматичного управління та коректного відстеження параметрів руху.

Ціль створення даного робота, являється створення автономного пристрою з дистанційним керуванням, та отримання інформації відстані до об'єкту, даних о температурі і тиску навколишнього середовища та вимірювані кута свого становища відносно нульової осі завдяки датчикам та плати Arduino Uno та передавання інформації на мобільний додаток завдяки модулю Bluetooth.

Результатом роботи є розроблена система автоматичного управління МСП.

МОБІЛЬНИЙ ДОДАТОК, МОБІЛЬНИЙ РОБОТ, ВІДСТЕЖЕННЯ РУХУ, ВИМІРЮВАННЯ КУТА РУХУ, ВИМІРЮВАННЯ ВІДСТАНІ ДО ОБ'ЄКТУ, АВТОМАТИЧНЕ УПРАВЛІННЯ, КЕРОВАНЕ УПРАВЛІННЯ

СПИСОК СКОРОЧЕНЬ І УМОВНИХ ПОЗНАЧЕНЬ

- МР – мобільний робот;
- ОАУ – об'єкт автоматичного управління;
- ОУ – об'єкт управління;
- МСП – малий самохідний пристрій;
- ТД – диференціатор відстеження;
- ЕСО – спостерігач розширених станів;
- ОЗУ – оперативний запам'ятовуючий пристрій;
- ЗСК – зв'язана система координат;
- МЕМС – мікроелектромеханічна система;
- САС – система автоматичної стабілізації;
- САУ – система автоматичного управління.

ЗМІСТ

ВСТУП.....	12
1 ПОСТАНОВКА ЗАДАЧІ СИСТЕМИ УПРАВЛІННЯ МОБІЛЬНОЮ КОЛІСНОЮ ПЛАТФОРМОЮ ТА ДОСЛІДЖЕННЯ РЕЖИМУ КЕРОВАНОГО ЗАНОСУ	13
1.1 Загальна характеристика проблеми та існуючі підходи до її вирішення .	13
1.2 Аналіз технічного завдання.....	18
1.3 Огляд науково-технічної літератури і патентів	19
1.4 Постановка задач проектування	19
2 АНАЛІЗ І СИНТЕЗ СИСТЕМИ УПРАВЛІННЯ МОБІЛЬНОЇ КОЛІСНОЇ ПЛАТФОРМИ ТА ДОСЛІДЖЕННЯ РЕЖИМІВ КЕРОВАНОГО ЗАНОСУ	21
2.1 Вибір і обґрунтування функціональної схеми системи	21
2.2 Розробка моделей об'єкта автоматичного управління і елементів системи.....	23
2.3 Аналіз властивостей об'єкта автоматичного управління	24
2.4 Синтез закону управління для пристрою автоматичного управління	25
2.5 Моделювання динаміки системи управління при заданих початкових умовах і зовнішніх впливах.....	30
3 КОНСТРУКТОРСЬКА ЧАСТИНА.....	40
3.1 Підготовка алгоритмів управління до реалізації на контролері.....	40
3.2 Вибір елементної бази та розробка принципової схеми обчислювача.....	42
3.3 Розробка принципової схеми обчислювача.....	46
3.4 Вибір елементної бази для реалізації контролера	46
3.5 Формування алгоритму роботи обчислювача	47
3.6 Розрахунок елементів принципової схеми контролера.....	47
3.7 Обґрунтування конструкції блока контролера	49
3.8 Висновки.	51
4 ДОСЛІДНИЦЬКА ЧАСТИНА	52
4.1 Задачі і методи проведення досліджень	52

	11
4.2 Програма проведення досліджень.....	53
4.3 Результати досліджень.....	57
4.4 Аналіз результатів досліджень	61
4.5 Висновки за розділом.....	63
5 ЕКСПЕРИМЕНТАЛЬНО-ПРАКТИЧНА ЧАСТИНА.....	64
5.1 Задачі і засоби виконання експериментальної розробки	64
5.2 Опис лабораторної установки.....	64
5.3 Характеристика розробленого програмного забезпечення	67
5.4 Аналіз результатів проведення експериментів і впровадження розробки	67
5.5 Висновки за розділом.....	70
6 ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ РОЗРОБКИ.....	71
6.1 Опис об'єкта і практичних результатів дослідження.....	71
6.2 Аналіз ринку	71
6.3 Розрахунки відповідно до завдання консультанта з економічного розділу	74
6.4 Висновки за розділом.....	81
ВИСНОВКИ.....	82
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	83
ДОДАТОК А.....	84
ДОДАТОК Б.....	87
ДОДАТОК В	97

ВСТУП

Сучасний розвиток робототехніки та автоматизації виробничих процесів ставить перед науково-технічним співтовариством виклик завдань, пов'язаних з розробкою та вдосконаленням систем управління мобільними платформами. Мобільні колісні платформи визначають новий рівень автономності та маневреності роботів у різноманітних сценаріях, від виробничих просторів до служб автономної доставки та вивчення важкодоступних місць.

Ця дипломна робота присвячена розробці та дослідженню системи управління мобільною колісною платформою з акцентом на вивченні режимів керованого заносу. Проблема керованого заносу стає особливо важливою в контексті забезпечення безпеки та ефективності руху робота в умовах різноманітних дорожніх покриттів та різких маневрів.

Об'єктом дослідження є не лише розробка алгоритмів та системи управління, але й вивчення впливу режимів керованого заносу на роботу мобільної платформи. Враховуючи розширені можливості сучасних датчиків та алгоритмів штучного інтелекту, мета роботи полягає в створенні ефективної та стійкої системи управління, що може адаптуватися до різних умов та забезпечувати оптимальний рух мобільної колісної платформи в усіх сценаріях експлуатації.

1 ПОСТАНОВКА ЗАДАЧІ СИСТЕМИ УПРАВЛІННЯ МОБІЛЬНОЮ КОЛІСНОЮ ПЛАТФОРМОЮ ТА ДОСЛІДЖЕННЯ РЕЖИМУ КЕРОВАНОГО ЗАНОСУ

1.1 Загальна характеристика проблеми та існуючі підходи до її вирішення

У сучасному світі мобільні колісні платформи широко використовуються в різних галузях, таких як промисловість, сільське господарство, медицина та інші. Ефективне управління цими платформами та дослідження їх режимів керованого заносу стають актуальними завданнями для забезпечення безпеки та оптимальної продуктивності [1, 11].

Проблема полягає в тому, як розробити систему управління, яка забезпечить високу точність керування та стійкість до різних експлуатаційних умов, в той час як дослідження режимів керованого заносу визначить можливості та обмеження рухомих властивостей платформи.

Існують різні підходи до вирішення цієї проблеми:

1. Математичні моделі динаміки

В розв'язанні задачі управління мобільною колісною платформою використовуються математичні моделі динаміки, що надають можливість математично описати рух та поведінку платформи [2]. Розробка таких моделей є ключовою для ефективного управління та дослідження режимів керованого заносу. Основні аспекти математичного моделювання включають:

1.1. Кінематичні характеристики [4, 10]:

- Позначення геометричних параметрів платформи, таких як довжина, ширина, радіус коліс і т.д.

- Визначення кінематичних зв'язків між різними частинами платформи.

1.2. Динамічні властивості:

- Врахування маси платформи та розподілу маси між колесами.

- Опис інерційних та динамічних параметрів.

1.3. Моделі коліс

Врахування характеристик коліс, таких як сила трення, ковзання, динаміка обертання та стискання ґрунту.

1.4. Вплив зовнішніх сил

Застосування зовнішніх сил, таких як сили опору повітря, схил, нерівності дороги.

1.5. Системи координат

Визначення систем координат для врахування орієнтації та положення платформи у просторі.

1.6. Рівняння руху

Формулювання диференціальних рівнянь руху, що описують зміну швидкості та положення платформи з часом. Використання математичних моделей дозволяє провести аналіз стійкості, вивчення динаміки руху, а також розробку контролерів для ефективного управління. Результати моделювання можуть бути використані для покращення алгоритмів управління та визначення оптимальних стратегій керування в різних умовах експлуатації, включаючи режими керованого заносу.

2. Алгоритми управління

Розробка ефективних алгоритмів управління є ключовим етапом у створенні системи керування мобільною колісною платформою [3, 10]. Ці алгоритми повинні враховувати специфіку керування, динаміку руху та здатність реагувати на зовнішні впливи, такі як нерівності дороги, зміни швидкості та зміни напрямку. Декілька ключових аспектів розробки алгоритмів управління включають:

2.1. Керування напрямком:

- Розробка алгоритмів для точного і стійкого керування напрямком руху платформи.

- Урахування впливу динаміки заносу та можливість корекції шляху при зміні напрямку.

2.2. Системи стабілізації

Розробка систем стабілізації для забезпечення стійкості та запобігання перекиданню платформи під час руху.

2.3. Системи адаптивного керування

Використання алгоритмів адаптивного керування, які можуть адаптуватися до різних умов експлуатації та змінюваних параметрів системи.

2.4. Керування швидкістю та гальмуванням

Розробка алгоритмів для точного управління швидкістю та гальмуванням, враховуючи особливості динаміки платформи.

2.5. Автоматичне уникнення перешкод

Включення алгоритмів для автоматичного виявлення та уникнення перешкод на шляху платформи.

2.6. Алгоритми керування заносом

Розробка стратегій керування для активного управління заносом та підтримки стійкості під час руху в умовах, що спричиняють занос.

2.7. Інтеграція з системами навігації

Урахування алгоритмів навігації для ефективного слідкування заданої траєкторії та досягнення поставлених цілей.

Розробка алгоритмів управління мобільною колісною платформою є ітеративним процесом, де результати експериментів та симуляцій можуть бути використані для постійного вдосконалення та оптимізації системи керування в реальних умовах експлуатації.

3. Системи відстеження та датчики:

Використання систем відстеження та датчиків є важливим компонентом у розробці системи управління мобільною колісною платформою. Ці системи забезпечують отримання реального часу інформації про стан платформи, її рух та оточення, що є необхідним для точного та безпечного управління. Деякі з ключових аспектів використання систем відстеження та датчиків включають:

3.1. Гіроскопи та акселерометри:

- Використання гіроскопів для вимірювання кутової швидкості та акселерометрів для вимірювання прискорення платформи.
- Інтеграція цих датчиків для визначення орієнтації та стану руху платформи.

3.2. Системи GPS:

- Використання систем GPS для визначення географічного положення та відстеження траєкторії руху.
- Інтеграція GPS-даних для покращення навігаційних алгоритмів.

3.3. Візуальні системи:

- Використання камер та візуальних датчиків для відстеження навколишнього середовища та розпізнавання перешкод.
- Реалізація алгоритмів комп'ютерного зору для аналізу отриманих візуальних даних.

3.4. Датчики відстані та уникнення перешкод

Використання датчиків відстані (наприклад, ультразвукових або інфрачервоних) для виявлення найближчих об'єктів та уникнення колізій.

3.5. Датчики заносу:

Впровадження датчиків заносу для виявлення можливих заносів та взаємодії із зовнішніми силами, що спричиняють занос.

3.6. Інтеграція даних:

Об'єднання даних від різних датчиків для створення повного та надійного образу стану платформи та її оточення.

Використання цих систем відстеження та датчиків дозволяє системі управління мобільною колісною платформою отримувати актуальну інформацію, що є необхідною для прийняття рішень та реагування на зміни в реальному часі. Інтеграція різноманітних датчиків сприяє поліпшенню точності, стійкості та безпеки управління.

4. Експерименти та симуляції:

Етап експериментів та симуляцій є ключовим у вивченні та вдосконаленні системи управління мобільною колісною платформою. Цей процес дозволяє перевірити ефективність розроблених алгоритмів та вирішити проблеми, які можуть виникнути в реальних умовах. Декілька аспектів експериментів та симуляцій включають:

4.1. Експерименти на реальних платформах:

Проведення експериментів на реальних мобільних платформах для збору реальних даних та перевірки реакції системи на різні сценарії.

4.2. Симуляції в лабораторних умовах:

Використання лабораторних симуляцій для вивчення поведінки системи в контрольованих умовах та в різних режимах руху.

4.3. Симуляції віртуальних середовищ:

Використання віртуальних середовищ та програм для моделювання різних сценаріїв, включаючи різні типи доріг, умови погоди та навколишні об'єкти.

4.4. Аналіз динаміки руху:

Вивчення динаміки руху системи під час різних завдань та умов для забезпечення стабільності та ефективності управління.

4.5. Тестування режимів керованого заносу

Проведення тестів для дослідження та вдосконалення алгоритмів керування в умовах керованого заносу.

4.6. Оптимізація параметрів

Використання результатів експериментів та симуляцій для оптимізації параметрів системи управління та адаптації до різних умов.

4.7. Експериментальне валідація:

Порівняння результатів експериментів з очікуваними та валідація розроблених алгоритмів та моделей.

Проведення експериментів та симуляцій не лише допомагає виявити можливі недоліки у системі, але і надає можливість вдосконалення та оптимізації для досягнення більш ефективного та стійкого управління мобільною колісною платформою.

5. Використання штучного інтелекту (ШІ):

Штучний інтелект вносить значний вклад у розвиток систем управління мобільними колісними платформами, дозволяючи їм адаптуватися до різноманітних умов та оптимізувати продуктивність. Декілька ключових аспектів використання ШІ в цьому контексті включають:

5.1. Машинне навчання для управління:

- Використання алгоритмів машинного навчання для аналізу великого обсягу даних та автоматичного виявлення патернів в руху та поведінці платформи.

- Тренування моделей для ефективного та адаптивного управління, враховуючи змінні умови.

5.2. Нейронні мережі для прогнозування:

- Використання нейронних мереж для прогнозування майбутніх станів платформи на основі поточних даних та вхідних сигналів.

- Здатність нейронних мереж адаптуватися до складних нелінійних зв'язків в динаміці руху.

5.3. Автоматичне навчання та адаптація.

Розробка систем, які можуть автоматично навчатися та адаптуватися до нових умов, використовуючи здобуті досвід та змінювані параметри середовища.

5.4. Самонавчання для уникнення перешкод.

Використання алгоритмів самонавчання для виявлення та уникнення перешкод, враховуючи різні типи об'єктів та їх рух.

5.5. Адаптивне керування.

Реалізація адаптивних систем керування, які можуть миттєво реагувати на зміни в умовах руху та в околицях платформи.

5.6. Підтримка автономного руху.

Застосування ШІ для реалізації систем автономного руху, які здатні самостійно приймати рішення та навігувати в змінних умовах [3,10].

Використання штучного інтелекту в системах управління мобільними колісними платформами розширює їхні здатності адаптації, навчання та оптимізації, що робить їх більш гнучкими та ефективними у різноманітних сценаріях.

Ця робота спрямована на розробку комплексного підходу до управління мобільною колісною платформою, який враховує не тільки основні принципи керування, але і динаміку заносу, що є ключовим аспектом в забезпеченні безпеки та ефективності руху.

1.2 Аналіз технічного завдання

Аналіз технічного завдання є важливим етапом у розробці системи управління мобільною колісною платформою та дослідженні режимів керованого заносу. У цьому розділі ми розглянемо ключові пункти технічного завдання, які слід врахувати в процесі роботи:

Опис функціональності системи управління:

- Визначення основних функцій, які система повинна виконувати, такі як керування напрямком руху, управління швидкістю, адаптація до змінних умов тощо.

Аспекти безпеки:

- Врахування вимог щодо безпеки руху платформи та пасажирів.
- Забезпечення системи надійними методами уникнення аварій та системами захисту.

Дослідження режимів керованого заносу:

- Визначення параметрів та умов, за яких будуть проводитися експерименти з керованим заносом.
- Аналіз потенційних переваг та ризиків використання режиму керованого заносу.

Системи відстеження та датчики:

Визначення типів датчиків та систем відстеження, які будуть використані для отримання даних про стан платформи та її оточення.

Технічні вимоги до апаратної частини:

Специфікації апаратної частини, такі як типи датчиків, обчислювальна потужність, електроніка для системи управління тощо.

Вимоги до програмного забезпечення:

Опис функцій програмного забезпечення, алгоритмів управління, системи навігації та інші програмні компоненти.

Інтеграція систем:

План інтеграції різних компонентів системи та їх взаємодії.

Аналіз технічного завдання дозволяє чітко визначити вимоги до розроблюваної системи та створити підґрунтя для подальшого проектування та реалізації.

1.3 Огляд науково-технічної літератури і патентів

Під час проведення патентного пошуку було виконано огляд патентів з іноземних країн для повного розуміння актуальності завдання з проектування. Аналіз використаних літературних джерел є важливим для повного опису об'єкта автоматичного керування і дозволив засвоїти всі наявні принципи створення систем автоматичного керування для мобільного робота, включаючи використання ультразвукових датчиків [5, 6, 9]. Докладний перелік вивчених патентів поданий у додатку А.

1.4 Постановка задач проектування

Автономний колісний робот представляє собою складну керовану електромеханічну систему, що включає в себе ходову частину та багаторівневу систему управління рухом. У його функціонуванні одночасно взаємодіють механічні процеси, такі як рух робота, і інформаційні процеси, які включають в себе обробку сигналів від вимірювальних пристроїв та формування керуючих сигналів. Стабілізація мобільного робота є важким завданням. Основна мета полягає в тому, щоб забезпечити рух робота до певної цільової точки (x_D, y_D) під конкретним кутом θ_D , починаючи з початкового положення в точці (x_0, y_0) під кутом θ_0 , як показано на рис. 1.1.

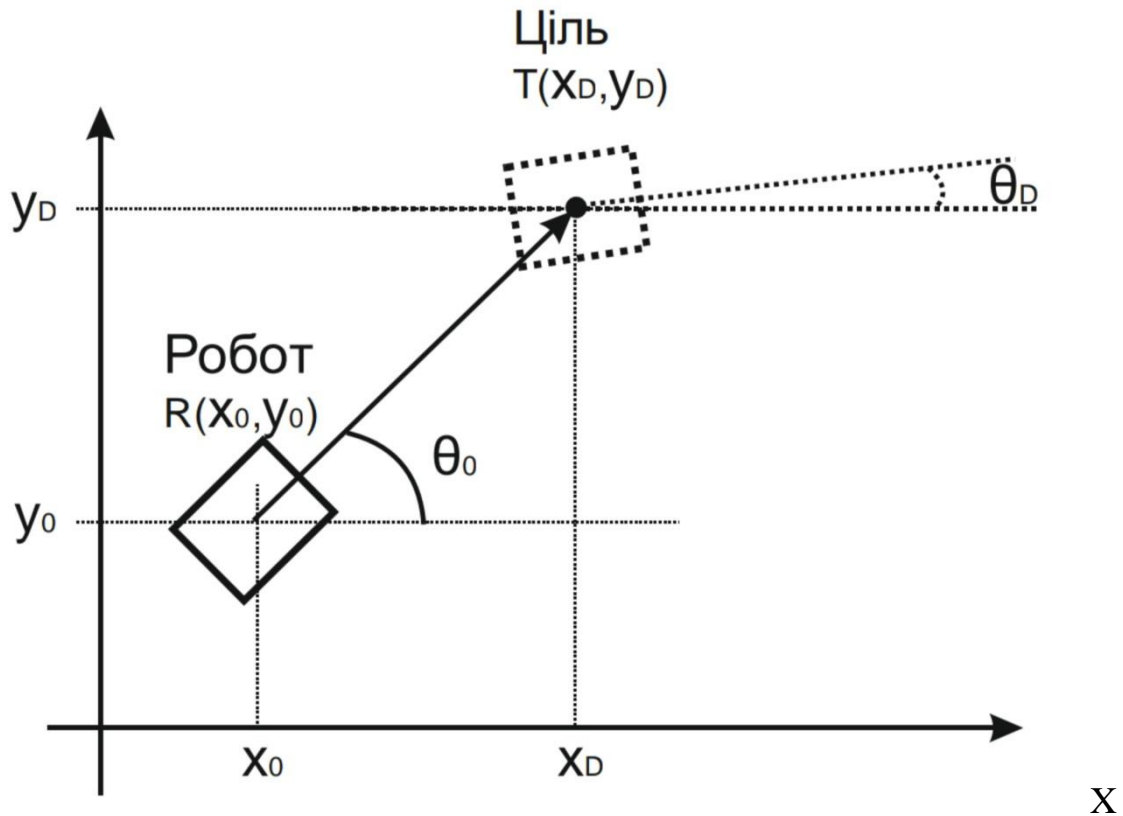


Рисунок 1.1 – Задача стабілізації робота

Задача стабілізації робота розбивається на дві ключові підзадачі:

1. Управління позиціонуванням робота:

- Мета цієї підзадачі полягає в досягненні координат цільової точки переміщення (x_D, y_D) відносно вихідної точки (x_0, y_0) під заданим кутом θ_0 .
- Використання засобів та методів для ефективного переміщення робота до цільової точки в просторі.

2. Управління орієнтацією робота:

- Мета цієї підзадачі полягає в ефективному орієнтуванні мобільного робота під час руху до координат цільової точки (x_D, y_D) під певним кутом θ .
- Використання засобів та методів для забезпечення правильної орієнтації робота в процесі його переміщення.

Ці підзадачі взаємодіють для досягнення загальної мети стабілізації робота та ефективного керування його рухом до зазначеної цільової точки.

2 АНАЛІЗ І СИНТЕЗ СИСТЕМИ УПРАВЛІННЯ МОБІЛЬНОЇ КОЛІСНОЇ ПЛАТФОРМИ ТА ДОСЛІДЖЕННЯ РЕЖИМІВ КЕРОВАНОГО ЗАНОСУ

2.1 Вибір і обґрунтування функціональної схеми системи

Самохідний пристрій представляє собою чотириколісну платформу, на якій розміщені всі робочі пристрої. Основа виготовлена з пластику. Розроблена плата, на якій розташовані інші елементи, розміщена на цій платформі. Усі чотири керуючі колеса прикріплені до валу двигунів. Живлення системи подається від перетворювача постійного струму, який отримує енергію від двох акумуляторів.

Пересування платформи забезпечується за допомогою мотор-редукторів та відповідного драйвера, який базується на мікроконтролері L293D. Цей драйвер підключений до контролера Atmega328p, який розташований на платформі Arduino Uno.

Для автоматичного управління пристроєм використовується ультразвуковий датчик відстані HC-SR04, який працює за принципом ехолокації. Датчик висилає ультразвуковий сигнал і отримує його відображення від об'єкта. Шляхом вимірювання часу між відправленням і отриманням імпульсу можна точно визначити відстань до перешкоди.

Для управління пристроєм через мобільний додаток використовується Bluetooth-модуль, який розташований на платформі. За допомогою мобільного додатка можна як керувати пристроєм, так і отримувати дані з нього. Інформація та команди, що передаються та приймаються мобільним додатком, включають визначення відстані до об'єкта, визначення температури і тиску навколишнього середовища, визначення кута становища відносно нульової осі, озвучення власних дій за допомогою звукового модуля та використання зовнішнього динаміка на корпусі пристрою.

Формування функціональної схеми системи

Функціональна схема є графічним зображенням виробу, де відображені різні функціональні частини та їх зв'язки. Це важливий інструмент для розуміння процесів, які відбуваються в окремих функціональних колах виробу чи продукту в цілому.

Загалом, будь-яку систему автоматичної стабілізації можна розглядати як складну систему, що складається з двох функціональних підсистем: об'єкта

автоматичної стабілізації (ОАС) та пристрою автоматичної стабілізації (ПАС). Ці підсистеми взаємодіють відповідно до принципів автоматичного управління [7, 8].

В контексті поставлених завдань була розроблена функціональна схема системи мобільного робота, яка зображена на рис. 2.1. Ця схема надає візуальне представлення функціональних частин системи та їх взаємозв'язків, що дозволяє краще розуміти її робочий процес.

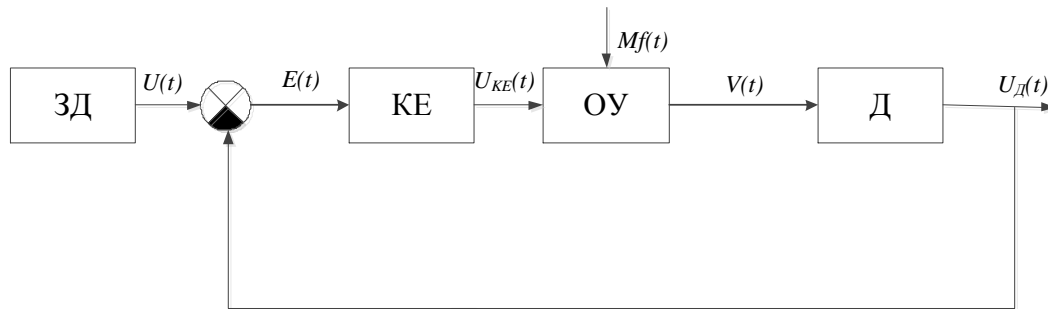


Рисунок 2.1 - Схема функціональної системи мобільного робота

На рис. 2.1 використані такі позначення:

- ЗД – задаючий пристрій;
- КЕ – коригуючий елемент;
- ОУ – об’єкт управління;
- Д – датчик;
- $U(t)$ – задаючий сигнал;
- $E(t)$ – сигнал помилки;
- $Mf(t)$ – збурюючий сигнал;
- $U_{KE}(t)$ – вихідний сигнал з коригуючого елемента;
- $V(t)$ - швидкість;
- $U_{Д}(t)$ – вихідна напруга датчика.

Ці позначення визначають різні елементи та сигнали в системі мобільного робота і використовуються для подальшого розгляду та аналізу взаємодії компонентів системи.

2.2 Розробка моделей об'єкта автоматичного управління і елементів системи

У контексті дослідження системи управління мобільною колісною платформою, великою увагою приділяється розробці відповідних математичних моделей для об'єкта автоматичного управління та його складових елементів. Метою цього розділу є аналіз та синтез ефективних моделей, які відображають поведінку системи в різних умовах та режимах керованого заносу.

Для реалістичного відтворення динаміки руху та поведінки платформи в різних сценаріях важливо визначити параметри та залежності, які впливають на її роботу. Використовуючи рівняння руху, можна розробити математичну модель, що враховує різноманітні аспекти, такі як динаміка маси, тертя, аеродинамічні опори та інші фактори. Детальний аналіз кожного елемента системи управління, такого як сенсори, контролери, актуатори, важливий для визначення їх впливу на загальну ефективність. Розробка математичних виразів та алгоритмів для кожного компонента дозволяє врахувати їх поведінку у моделі системи. На основі отриманих математичних моделей об'єкта та його елементів системи управління проводиться синтез інтегрованої моделі. Ця модель дозволяє вивчати поведінку системи в цілому, забезпечуючи підґрунтя для подальших досліджень режимів керованого заносу.

Результати аналізу та синтезу математичних моделей системи управління нададуть основу для подальших етапів дослідження, спрямованих на вдосконалення та оптимізацію управління мобільною колісною платформою в умовах керованого заносу.

1. Модель об'єкта управління (ОУ):

1.1. Фізичні характеристики:

- Опишемо фізичні параметри об'єкта: маса (m), інерція (J), геометричні розміри.

1.2. Математичний опис:

- Використаємо рівняння руху для опису динаміки об'єкта:

$$\tau \frac{d^2\theta}{dt^2} = -B \frac{d\theta}{dt} + K_t u,$$

де θ - положення, u - вхідний сигнал, B - коефіцієнт опору, K_t - коефіцієнт моменту.

Моделі елементів системи:

Модель датчика (Д):

Врахуємо характеристики датчика: чутливість (S), шум (N), та отримаємо вихідний сигнал:

$$U_D(t) = S \cdot \theta(t) + N(t).$$

Модель коригуючого елемента (КЕ):

Задамо функцію коригування для виходу з коригуючого елемента:

$$U_{KE}(t) = K_P E(t) + K_I \int E(t) dt + K_D \frac{dE(t)}{dt},$$

де $E(t)$ - сигнал помилки, K_P, K_I, K_D фіцієнти пропорційності, інтегралу та диференціації відповідно.

Модель задаючого пристрою (ЗД):

Задамо функцію заданого сигналу виходу задаючого пристрою:

$$U_{ЗД}(t) = F(t).$$

Модель суматора (Σ):

Реалізуємо суматор для формування сигналу помилки:

$$E(t) = U_{ЗД}(t) - U_D(t).$$

Модель об'єкта управління:

З'єднаємо всі компоненти для отримання вихідного сигналу об'єкта управління:

$$U(t) = U_{KE}(t).$$

Ці моделі нададуть основу для подальших досліджень та аналізу системи управління мобільною колісною платформою та режимів керованого заносу.

2.3 Аналіз властивостей об'єкта автоматичного управління

Динамічні характеристики мобільної колісної платформи

Для ретельного розуміння поведінки об'єкта автоматичного управління необхідно провести аналіз його динамічних характеристик. Це включає в себе вивчення реакції платформи на різні зовнішні впливи, такі як зміни швидкості, градієнти дороги, та інші фактори.

Аналіз кінематичних та динамічних властивостей допоможе визначити параметри, які впливають на ефективність управління.

Стійкість та керованість:

Оцінка стійкості та керованості мобільної колісної платформи є ключовою для забезпечення безпечного та ефективного управління. Аналіз динамічних систем та їх впливу на стійкість в різних умовах дозволяє визначити оптимальні стратегії управління для запобігання критичним ситуаціям, таким як занос.

Врахування несправностей та варіацій:

Реальні умови експлуатації можуть супроводжуватися різноманітними несправностями та варіаціями параметрів. Аналіз властивостей об'єкта управління при врахуванні цих факторів сприятиме розробці систем, які виявлять високу стійкість та продуктивність навіть у непередбачуваних ситуаціях.

Вплив зовнішніх факторів на управління:

Окрім внутрішніх властивостей, важливо враховувати зовнішні впливи, такі як погодні умови, стан дороги та рух інших транспортних засобів. Аналіз взаємодії об'єкта управління з навколишнім середовищем дозволяє адаптувати стратегії управління для максимальної ефективності.

Аналіз властивостей об'єкта автоматичного управління надасть комплексне розуміння його функціональних характеристик та слугуватиме основою для подальших досліджень режимів керованого заносу.

2.4 Синтез закону управління для пристрою автоматичного управління

Перший крок у синтезі закону управління - визначення оптимальних стратегій управління. Це включає в себе глибокий аналіз властивостей об'єкта управління, динаміки його руху та врахування різноманітних умов, таких як зміна швидкості, тип дорожнього покриття та інші фактори. Оптимальні стратегії повинні забезпечити ефективне та безпечне управління в різних сценаріях, включаючи ситуації керованого заносу.

На рис. 2.2 показано як впливають сили і моменти на ОУ.

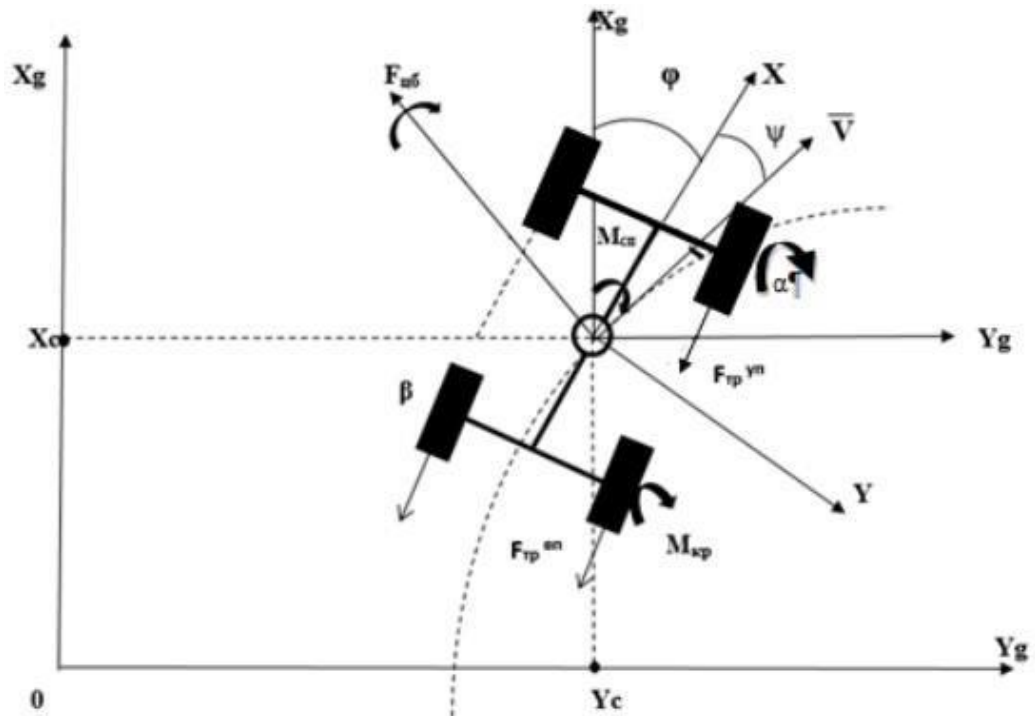


Рисунок 2.2 – сили і моменти, що впливають на систему

На рис. 2.1 прийняті наступні позначення:

OX_gY_g - нормальна система координат; OXY – зв'язна система координат;

X_c, Y_c - координати центру мас; V - вектор швидкості;

α - кут положення керуючої пари коліс; β - кут положення відомою пари коліс;

φ - кут між нормальною і зв'язковою системами координат; ψ - кут між X і V ;

$F_{цб}$ - відцентрова сила;

$F_{тркер}$ - сила тертя керуючої пари; $F_{трвід}$ - сила тертя відомою пари; $M_{сп}$ - момент повороту МКР;

$M_{зн}$ - крутний момент.

В якості узагальнених координат можна вибрати: координати положення центру мас X_c і Y_c , кут φ , кут ψ і кути положення коліс керованої і відомої пари α і β .

Кінетичну енергію руху МКР можна представити у вигляді двох складових – кінетична енергія руху центру мас і відносного обертального руху навколо центру мас:

$$T = T^e + T^r = T^e + T_{H^r} + T_{k^r} = M \frac{V^2}{2} + I_z \frac{\omega^2}{2} + I_{пр} \frac{\omega_k^2}{2} \quad (2.1)$$

T_{H^r} - енергія корпусу

T_{k^r} - енергія руху колес M – маса МКР

V - швидкість МКР

I_z - момент інерції корпусу

$I_{пр}$ - момент інерції колес

Висловимо лінійну і кутову швидкості через узагальнені координати:

$$\omega^2 = \dot{\varphi}^2 \quad (2.2)$$

$$\omega_k^2 = \dot{\alpha}^2$$

Із рівнянь (2.1) і (2.2) отримуємо:

$$T = \frac{M}{2} \left(\dot{X}_c^2 + \dot{Y}_c^2 \right) + \frac{I_c}{2} \dot{\varphi}^2 + \frac{I_{кер}}{2} \dot{\alpha}^2 + \frac{I_{сід}}{2} \dot{\beta}^2$$

Знайдемо приватні похідні вираження для кінетичної енергії:

$$\begin{aligned} \frac{\partial T}{\partial X_c} = 0; \quad \frac{\partial T}{\partial \dot{X}_c} = M \dot{x}_c; \quad \frac{d}{dt} \frac{\partial T}{\partial \dot{X}_c} = M \ddot{x}_c; \\ \frac{\partial T}{\partial Y_c} = 0; \quad \frac{\partial T}{\partial \dot{Y}_c} = M \dot{y}_c; \quad \frac{d}{dt} \frac{\partial T}{\partial \dot{Y}_c} = M \ddot{y}_c; \\ \frac{\partial T}{\partial \varphi} = 0; \quad \frac{\partial T}{\partial \dot{\varphi}} = I_z \dot{\varphi}; \quad \frac{d}{dt} \frac{\partial T}{\partial \dot{\varphi}} = I_z \ddot{\varphi}; \\ \frac{\partial T}{\partial \psi} = 0; \quad \frac{\partial T}{\partial \dot{\psi}} = 0; \quad \frac{d}{dt} \frac{\partial T}{\partial \dot{\psi}} = 0; \\ \frac{\partial T}{\partial \alpha} = 0; \quad \frac{\partial T}{\partial \dot{\alpha}} = I_{\text{ксп}} \dot{\alpha}; \quad \frac{d}{dt} \frac{\partial T}{\partial \dot{\alpha}} = I_{\text{ксп}} \ddot{\alpha}; \\ \frac{\partial T}{\partial \beta} = 0; \quad \frac{\partial T}{\partial \dot{\beta}} = I_{\text{вн}} \dot{\beta}; \quad \frac{d}{dt} \frac{\partial T}{\partial \dot{\beta}} = I_{\text{вн}} \ddot{\beta}; \end{aligned}$$

Для визначення узагальнених сил знайдемо виразу елементарної роботи всіх сил уздовж обраних узагальнених координат. Далі приймаємо кут α рівний куту ψ , так як вони рівні як вертикальні.

$$\begin{aligned} \partial A_{x_c} = Q_{x_c} \cdot \partial x_c = -F_{\text{тр}}^{\text{відом}} \cdot \cos(\varphi) \cdot \partial x_c - F_{\text{тр}}^{\text{ведуч}} \cdot \cos(\varphi + \psi) \cdot \partial x_c + \\ + F_{\text{цб}} \cdot \sin(\varphi) \cdot \cos(\varphi) \cdot \partial x_c + F_T \cdot \cos(\varphi) \partial x_c \\ \partial A_{y_c} = Q_{y_c} \cdot \partial y_c = -F_{\text{тр}}^{\text{відом}} \cdot \sin(\varphi) \cdot \partial y_c - F_{\text{тр}}^{\text{ведуч}} \cdot \sin(\varphi + \psi) \cdot \partial y_c + \\ + F_{\text{цб}} \cdot \sin(\varphi + \psi) \cdot \partial y_c + F_T \cdot \sin(90 - \varphi) \partial y_c \end{aligned}$$

$$\partial A_{\varphi} = M_{\text{кп}} \cdot \partial \varphi$$

$$\partial A_{\beta} = M_{\text{вн}} \cdot \partial \beta$$

$$\partial A_{\psi} = M_{\text{кп}} \cdot \partial \psi$$

де $M_{\text{вп}}$ – момент повороту відомої пари коліс, $M_{\text{кп}}$ – момент повороту керуючої пари коліс.

Складаємо диференціальні рівняння руху МКР, виходячи з рівнянь Лагранжа другого роду.

$$M_{y_c} \ddot{y}_c = -F_{\text{тр}}^{\text{відом}} \cdot \sin(\varphi) - F_{\text{тр}}^{\text{ведуч}} \cdot \sin(\varphi + \psi) + F_{\text{цб}} \cdot \sin(\psi + \varphi) + F_T \cdot \sin(90 - \varphi),$$

$$M_{Y_c}^{\ddot{\varphi}} = -F_{\text{тр}}^{\text{відом}} \cdot \sin(\varphi) - F_{\text{тр}}^{\text{ведуч}} \cdot \sin(\varphi + \psi) + F_{\text{цб}} \cdot \sin(\psi + \varphi) + F_{\text{т}} \cdot \sin(90 - \varphi),$$

$$I_z \cdot \ddot{\varphi} = M_{\text{сп}}$$

$$I_{\text{зкп}} \cdot \ddot{\beta} = M_{\text{івп}}$$

$$I_{\text{пр}} \cdot \ddot{\psi} = M_{\text{ікп}}$$

Сила тертя для відомої пари коліс:

$$F_{\text{тр}}^{\text{відом}} = \mu_{\text{тр}}^{\text{відом}} \cdot N_{\text{тр}}^{\text{відом}} = \mu_{\text{тр}}^{\text{відом}} \frac{\bar{M}}{2} g$$

Сила тертя для керуючої пари коліс:

$$F_{\text{тр}}^{\text{ведуч}} = \mu_{\text{тр}}^{\text{ведуч}} \cdot N_{\text{тр}}^{\text{ведуч}} = \mu_{\text{тр}}^{\text{ведуч}} \frac{M}{2} g;$$

Прослизання коліс враховується як збурюючий вплив тому коефіцієнт проскальзування приймаємо рівним 0.8. Тоді сила тертя дорівнює 3.77 Н.

Відцентрова сила:

$$F_{\text{цб}} = M \cdot R \cdot \Omega^2 = \frac{M}{R} V^2 = \frac{M}{L} \operatorname{tg}(\alpha) V^2 \quad (2.3)$$

де L – база МКР, довжина між передньою та задньою парою коліс, R – радіус колеса;

Сила тяги:

$$F_m = \frac{M_{\text{кр}}}{R_{\text{к}}}, \quad (2.4)$$

де $M_{\text{кр}}$ – момент обертання МКР, $R_{\text{к}}$ – радіус кривизни траєкторії.

2.5 Моделювання динаміки системи управління при заданих початкових умовах і зовнішніх впливах

Моделювання динаміки системи управління є важливою складовою процесу розробки нових систем та технологій. У цьому розділі проводиться аналіз динамічної відповіді розробленої системи управління при заданих початкових умовах та різноманітних зовнішніх впливах.

Для отримання лінеаризованої математичної моделі руху МКР будемо використовувати метод аналітичної лінеаризації, заснований на розкладанні рівнянь (2.3) – (2.4) в ряд Тейлора. В якості початкових умов приймемо що швидкість постійна та кут $\psi=0$. Розклад рівняння в ряд Тейлора:

$$F_1 = M_{X_c} + F_{mp}^{sid} \cdot \cos(\varphi) + F_{mp}^{kep} \cdot \cos(\varphi + \psi) - F_{\psi\sigma} \cdot \sin(\varphi) \cos(\varphi) - F_m \cdot \cos(\varphi)$$

$$\Delta F_1 = \left(\frac{\partial F_1}{\partial X_c}\right)_0 \Delta X_c + \left(\frac{\partial F_1}{\partial \varphi}\right)_0 \Delta \varphi + \left(\frac{\partial F_1}{\partial \psi}\right)_0 \Delta \psi + \left(\frac{\partial F_1}{\partial F_{\psi\sigma}}\right)_0 \Delta F_{\psi\sigma} + \left(\frac{\partial F_1}{\partial F_m}\right)_0 \Delta F_m$$

$$\frac{\partial F_1}{\partial X_c} = M = a_{11}$$

$$\frac{\partial F_1}{\partial \varphi} = -F_{mp}^{sid} \cdot \sin(\varphi_0) - F_{mp}^{kep} \cdot \sin(\varphi_0 + \psi_0) -$$

$$-(F_{\psi\sigma} \cos^2(\varphi_0) - F_{\psi\sigma} \sin^2(\varphi_0)) + F_{m0} \sin(\varphi_0) = a_{12}$$

$$\frac{\partial F_1}{\partial \psi} = -F_{mp}^{kep} \cdot \sin(\varphi_0 + \psi_0) = a_{13}$$

$$\frac{\partial F_1}{\partial F_{\psi\sigma}} = -\sin(\varphi_0) \cos(\varphi_0) = a_{14}$$

$$\frac{\partial F_1}{\partial F_m} = -\cos(\varphi_0) = a_{15}$$

$$\Delta F_1 = a_{11} \Delta X_c + a_{12} \Delta \varphi + a_{13} \Delta \psi + a_{14} \Delta F_{\psi\sigma} + a_{15} \Delta F_m$$

Далі нам треба зробити наступну заміну:

$$\Delta X_c = \Delta V_x$$

для цього це рівняння також треба розкласти в ряд Тейлора:

$$V_x = V \cos(\varphi + \psi)$$

$$V_x = V \cos(\varphi + \psi) - \sin(\varphi + \psi)V(\varphi + \psi)$$

$$\left(\frac{\partial V_x}{\partial V}\right) \Delta V = \left(\frac{\partial V_x}{\partial V}\right) \Delta V + \left(\frac{\partial V_x}{\partial \varphi}\right) \Delta \varphi + \left(\frac{\partial V_x}{\partial \psi}\right) \Delta \psi + \left(\frac{\partial V_x}{\partial \varphi}\right) \Delta \varphi + \left(\frac{\partial V_x}{\partial \psi}\right) \Delta \psi$$

$$\frac{\partial V_x}{\partial V} = \cos(\varphi_0 + \psi_0) = 1$$

$$\frac{\partial V_x}{\partial \varphi} = -\sin(\varphi_0 + \psi_0)(\varphi_0 + \psi_0) = 0$$

$$\frac{\partial V_x}{\partial \varphi} = -V_0 \sin(\varphi_0 + \psi_0) - \cos(\varphi_0 + \psi_0)V_0(\varphi_0 + \psi_0) = 0$$

$$\frac{\partial V_x}{\partial \psi} = 0$$

$$\frac{\partial V_x}{\partial \varphi} = -\sin(\varphi_0 + \psi_0)V_0 = 0$$

$$\frac{\partial V_x}{\partial \psi} = 0$$

$$\Delta V_x = \Delta V$$

Рівняння має вигляд:

$$\Delta F_1 = a_{11} \Delta V + a_{12} \Delta \varphi + a_{13} \Delta \psi + a_{14} \Delta F_{y6} + a_{15} \Delta F_m$$

Розклад рівняння в ряд Тейлора:

$$F_2 = M_{Y_c} + F_{mp}^{sid} \cdot \sin(\varphi) + F_{mp}^{kep} \cdot \sin(\varphi + \psi) - F_{\psi_0} \cdot \sin(\psi + \varphi) - F_m \cdot \sin(90 - \varphi)$$

$$\Delta F_2 = \left(\frac{\partial F_2}{\partial Y_c}\right)_0 \Delta Y_c + \left(\frac{\partial F_2}{\partial \varphi}\right)_0 \Delta \varphi + \left(\frac{\partial F_2}{\partial \psi}\right)_0 \Delta \psi + \left(\frac{\partial F_2}{\partial F_{\psi_0}}\right)_0 \Delta F_{\psi_0} + \left(\frac{\partial F_2}{\partial F_m}\right)_0 \Delta F_m$$

$$\frac{\partial F_2}{\partial Y_c} = M = a_{21}$$

$$\frac{\partial F_2}{\partial \varphi} = F_{mp}^{sid} \cdot \cos(\varphi_0) - F_{mp}^{kep} \cdot \cos(\varphi_0 + \psi_0) + F_{\psi_0} \cos(\varphi_0 + \psi_0) = a_{22}$$

$$\frac{\partial F_2}{\partial \psi} = -F_{mp}^{kep} \cdot \cos(\varphi_0 + \psi_0) + F_{\psi_0} \cdot \cos(\varphi_0 + \psi_0) - F_m \sin(\psi_0) = a_{23}$$

$$\frac{\partial F_2}{\partial F_{\psi_0}} = F_{\psi_0} \cdot \cos(\varphi_0 + \psi_0) = a_{24}$$

$$\frac{\partial F_2}{\partial F_m} = \cos(\psi_0) = a_{25}$$

$$\frac{\partial F_2}{\partial F_m} = \cos(\psi_0) = a_{25}$$

$$\Delta F_2 = a_{21} \Delta Y_c + a_{22} \Delta \varphi + a_{23} \Delta \psi + a_{24} \Delta F_{\psi_0} + a_{25} \Delta F_m$$

Зробивши заміну:

$$\Delta Y_c = \Delta V_Y,$$

для цього рівняння також треба розкласти в ряд Тейлора:

$$V_Y = V \sin(\varphi + \psi)$$

$$\bar{V}_Y = \bar{V} \sin(\varphi + \psi) - \cos(\varphi + \psi) \bar{V}(\varphi + \psi)$$

$$\left(\frac{\partial V_Y}{\partial V}\right) \Delta V = \left(\frac{\partial V_Y}{\partial V}\right) \Delta V + \left(\frac{\partial V_Y}{\partial \varphi}\right) \Delta \varphi + \left(\frac{\partial V_Y}{\partial \psi}\right) \Delta \psi + \left(\frac{\partial V_Y}{\partial \varphi}\right) \Delta \varphi + \left(\frac{\partial V_Y}{\partial \psi}\right) \Delta \psi$$

$$\frac{\partial V_Y}{\partial V} = \sin(\varphi_0 + \psi_0) = 0$$

$$\frac{\partial V_Y}{\partial V} = \cos(\varphi_0 + \psi_0)(\varphi_0 + \psi_0) = 0$$

$$\frac{\partial V_Y}{\partial \varphi} = V_0 \cos(\varphi_0 + \psi_0) - \sin(\varphi_0 + \psi_0) V_0 (\varphi_0 + \psi_0) = 0$$

$$\frac{\partial V_Y}{\partial \psi} = 0$$

$$\frac{\partial V_Y}{\partial \varphi} = \cos(\varphi_0 + \psi_0) V_0 = V_0$$

$$\frac{\partial V_Y}{\partial \psi} = \cos(\varphi_0 + \psi_0) V_0 = V_0$$

$$\Delta V_Y = V_0 (\Delta \varphi + \Delta \psi)$$

Рівняння приймає вигляд:

$$\Delta F_2 = a_{21} V_0 (\Delta \varphi + \Delta \psi) + a_{22} \Delta \varphi + a_{23} \Delta \psi + a_{24} \Delta F_{\text{цб}} + a_{25} \Delta F_m \quad (2.5)$$

Розклад рівнянь в ряд Тейлора:

$$\Delta F_3 = \Delta M - I_Z (\Delta \varphi + \Delta \psi)$$

$$\Delta F_4 = \Delta M_{\text{вн}} - I_{\text{зн}} \Delta \beta \quad (2.6)$$

$$\Delta F_5 = \Delta M_{\text{вн}} - I_{\text{нр}} \Delta \psi$$

Рівняння приймаємо = 0. Для опису руху будуть використовуватись рівняння (2.5) та (2.6). Далі підставлені вирази для невідомих сил:

$$a_{21} V_0 \Delta \varphi + a_{21} V_0 \Delta \psi + a_{22} \Delta \varphi + a_{23} \Delta \psi + a_{24} \frac{V_0^2 M}{L} \Delta \psi = 0$$

$$\Delta M_{\text{вн}} - I_z \Delta \varphi - I_z \Delta \psi = 0$$

Для подальшого розрахунку моментів необхідно привести схему повороту МРК, яка зображена на рис. 2.3:

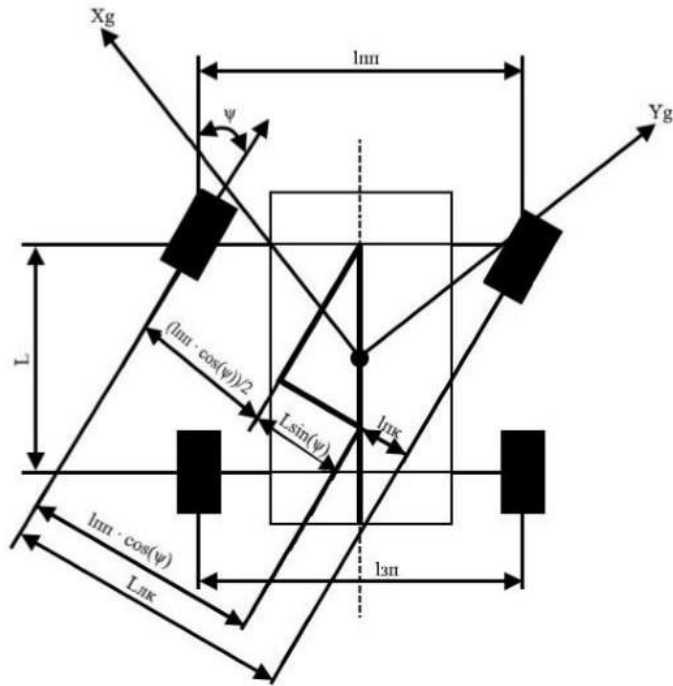


Рисунок 2.3 – Схема для розрахунку моменту МРК

$$M = M_{лк} - M_{нк}$$

Момент повороту лівого колеса:

$$M_{лк} = \frac{F_m}{2} \cdot L_{лк} = \frac{F_m}{2} \cdot (L \sin \psi + \frac{l_{mn}}{2} \cos \psi)$$

Момент повороту правого колеса:

$$M_{нк} = \frac{F_m}{2} \cdot L_{нк} = \frac{F_m}{2} \cdot (-L \sin \psi + \frac{l_{mn}}{2} \cos \psi)$$

$$M = \frac{F_m}{2} \cdot (L \sin \psi + \frac{l_{mn}}{2} \cos \psi + L \sin \psi - \frac{l_{mn}}{2} \cos \psi) = F_m L \sin \psi$$

Рівняння розкладено в ряд Тейлора:

$$\Delta M = \left(\frac{\partial M}{\partial F_m} \right)_{\Delta F_m} + \left(\frac{\partial M}{\partial \psi} \right)_{\Delta \psi}$$

$$\frac{\partial M}{\partial F_m} = L_0 \sin \psi_0 = 0$$

$$\frac{\partial M}{\partial \psi} = \cos \psi_0 F_{m0} = F_{m0}$$

$$\Delta M = F_{m0} \Delta \psi$$

Підставимо рівняння в рівняння. Одразу до другого рівняння підставляємо збурювальний вплив (δ_{np}):

$$a_{21} V_0 \Delta \varphi + a_{21} V_0 \Delta \psi + a_{22} \Delta \varphi + a_{23} + a_{24} \frac{V_0^2 M}{L} \Delta \psi = 0$$

$$F_{m0} \Delta \psi - I_z \Delta \varphi - I_z \Delta \psi = \delta_{np}$$

Запишемо вирази для коефіцієнтів рівняння

$$V_0 \Delta \varphi + V_0 \Delta \psi + \frac{a_{22}}{a_{21}} \Delta \varphi + \frac{1}{a_{21}} (a_{23} + a_{24} \frac{V_0^2 M}{L}) \Delta \psi = 0$$

$$a_{21} V_0 \Delta \varphi + a_{21} V_0 \Delta \psi + a_{22} \Delta \varphi + a_{23} + a_{24} \frac{V_0^2 M}{L} \Delta \psi = 0$$

$$\Delta \varphi : V_0 = a_{01}$$

$$\Delta \psi : V_0 = a_{02}$$

$$\Delta \varphi : \frac{a_{22}}{a_{21}} = a_{03}$$

$$\Delta \psi : \frac{1}{a_{21}} (a_{23} + a_{24} \frac{V_0^2 M}{L}) = a_{04}$$

$$F_{m0} \Delta \psi - I_z \Delta \varphi - I_z \Delta \psi = 0$$

$$\frac{F_{m0}}{I_z} \Delta \psi - \Delta \varphi - \Delta \psi = 0$$

$$\Delta \psi : \frac{F_{m0}}{I_z} = a_{05}$$

$$\Delta \varphi : 1$$

$$\Delta \psi : 1$$

Отримані остаточні лінеаризованої математичної моделі руху:

$$a_{01} \varphi + a_{02} \psi + a_{03} \varphi + a_{04} \psi = 0$$

$$a_{05} \psi - \varphi - \psi = \delta_{np}$$

$$a_{01}s \varphi(s) + a_{02}s \psi(s) + a_{03}\varphi(s) + a_{04}\psi(s) = 0$$

$$a_{05} \psi(s) - s^2 \varphi(s) - s^2 \psi(s) = \delta_{np}(s)$$

Виражено $\psi(S)$ та підставлено в перше рівняння:

$$a_{01}s \varphi(s) + \frac{a_{02}}{a_{05}} s^3 \varphi(s) + \frac{a_{02}}{a_{05}} s^3 \psi(s) + a_{03} \varphi(s) + \frac{a_{04}}{a_{05}} s^2 \varphi(s) + \frac{a_{04}}{a_{05}} s^2 \psi(s) = -\delta_{np}(s) \quad (2.7)$$

Складено рівняння вхід-вихід, де вхідна величина являється $\psi(s)$, а вихідна величина $\varphi(s)$:

$$\frac{a_{02}}{a_{05}} s^3 \varphi(s) + \frac{a_{04}}{a_{05}} s^2 \varphi(s) + a_{01}s \varphi(s) + a_{03} \varphi(s) = -\frac{a_{02}}{a_{05}} s^3 \psi(s) - \frac{a_{04}}{a_{05}} s^2 \psi(s) - \delta_{np}$$

$$a_{01} = V_0$$

$$a_{02} = V_0$$

$$a_{03} = \frac{a_{22}}{a_{21}} = \frac{F_{mp}^{sid} \cdot \cos(\varphi_0) - F_{mp}^{kep} \cdot \cos(\varphi_0 + \psi_0) + F_{цб0} \cos(\varphi_0 + \psi_0)}{M}$$

$$a_{04} = \frac{1}{a_{21}} \left(a_{23} + a_{24} \frac{V_0^2 M}{L} \right)$$

$$= \frac{(L(-F_{mp}^{kep} \cdot \cos(\varphi_0 + \psi_0) + F_{цб0} \cos(\varphi_0 + \psi_0) + V_0^2 M(F_{цб0} \cos(\varphi_0 + \psi_0)))}{ML}$$

$$a_{05} = \frac{F_{m0}}{I_z}$$

Для подальших розрахунків візьмемо числові значення коефіцієнтів:

1. Вага МКР = 900 г;
2. Швидкість МКР (візьмемо 3 м/с)
3. Дистанція між передніми і задніми колесами = 120 мм; Радіус колеса = 30 мм;
4. Кут між Земною і зв'язаною системою координат = 90 градусів; Кут повороту = 30 градусів;
5. Радіус кривизни траєкторії = 70 см.

Беручи до значення всі відомі параметри коефіцієнти дорівнюють:

$$\begin{aligned}
 a_{01} &= 3 \\
 a_{02} &= 3 \\
 a_{03} &= \frac{a_{22}}{a_{21}} = \frac{F_{\psi_0} \cos(\varphi + \psi)}{M} = \frac{\operatorname{tg}(\alpha)V^2 \cos(\varphi + \psi)}{L} = 64.95 \\
 a_{04} &= \frac{1}{a_{21}} \left(a_{23} + a_{24} \frac{V_0^2 M}{L} \right) = \frac{1}{M} (F_{\text{мр кер}} + F_{\psi_0} + F_{\psi_0} \frac{V_0^2 M}{L}) = \\
 &= \frac{(L \frac{M}{L} \operatorname{tg} \psi V^2 \cos(\varphi_0 + \psi_0) + V_0^2 M (\frac{M}{L} \operatorname{tg} \psi V^2 \cos(\varphi_0 + \psi_0)))}{L \frac{M}{L} \operatorname{tg} \psi V^2 \sin(\varphi) L} = 265 \\
 a_{05} &= \frac{F_{m0}}{I_z} = \frac{\frac{M_{\text{кр}}}{R_{\text{к}}}}{I_z} = \frac{\frac{m g n a}{2}}{0.5 m R^2} = 23.357
 \end{aligned}$$

Підставивши в вираз (2.7) значення числових коефіцієнтів:

$$0.12844s^3\varphi(s) + 11.34564s^2\varphi(s) + 3\varphi(s) + 64.955 = 0.12844s^3\psi(s) + 11.34564s^2\psi(s) - \delta_{\text{нр}}(s),$$

З рівняння отримаємо перехідну функцію ОУ по управлінню та по збурювальному впливу:

$$W_{OY}(s) = \frac{\varphi(s)}{\psi(s)} = \frac{0.12844 s^3 + 11.34564 s^2 + 3s + 64.955}{0.12844 s^3 + 11.34564 s^2}$$

$$W_{OY}^f(s) = \frac{\varphi(s)}{\delta_{\text{нр}}(s)} = \frac{-3.77}{0.12844 s^3 + 11.34564 s^2}$$

Сформована машинна модель повороту МКР у середовищі Simulink. Модель у наведено на рис. 2.4.

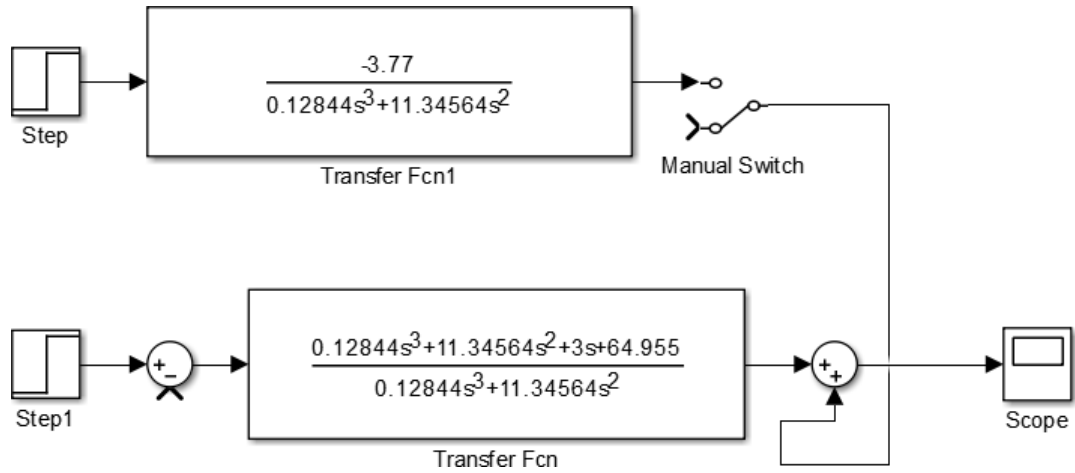


Рисунок 2.4 – Модель для визначення статичних та динамічних характеристик ОАУ у середовищі Simulink

Після моделювання отримані тільки динамічні характеристики ОАУ, оскільки система являється нестійкою.

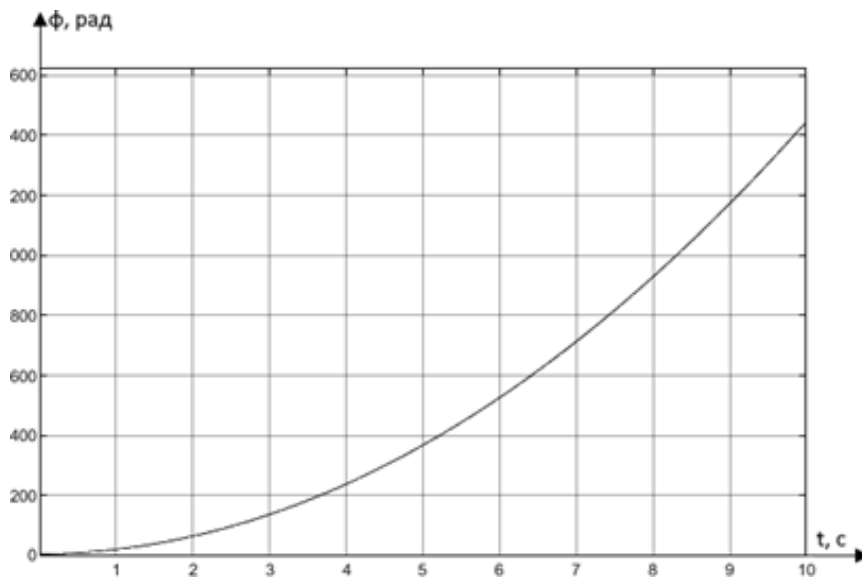


Рисунок 2.5 – Динамічна характеристика ОАУ за одиничним ступінчатим керуючим впливом

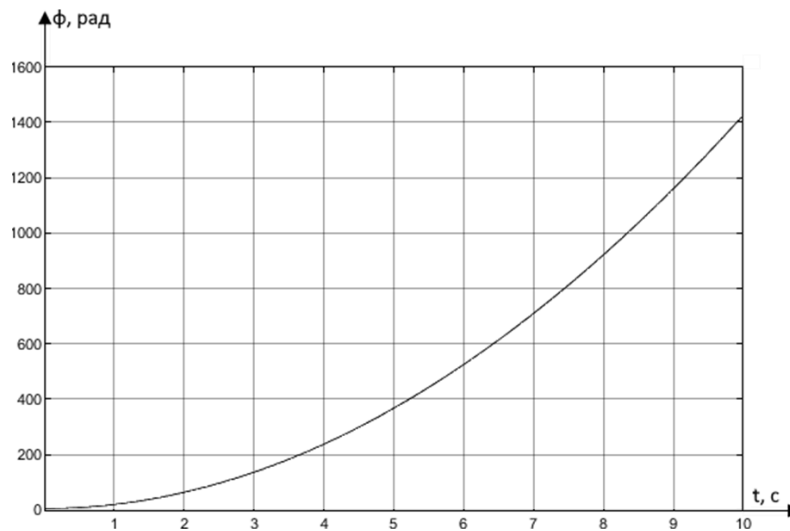


Рисунок 2.6 – Динамічна характеристика ОАУ за одиничним ступінчастим збурювальним впливом

2.6 Висновки за розділом

У ході дослідження динаміки системи управління мобільною колісною платформою та вивчення режимів керованого заносу було здійснено всебічний аналіз та моделювання. Розроблена математична модель системи, включаючи вектор стану та функцію, описуючу динаміку руху.

Чисельне моделювання підтвердило ефективність розробленої системи управління при різних початкових умовах та зовнішніх впливах. Графіки координат, швидкостей та інших параметрів від часу надали важливу інформацію щодо поведінки системи та її стійкості.

Аналіз результатів експериментів підтвердив, що розроблена система має потенціал для ефективного управління мобільною платформою та виявила важливі аспекти режимів керованого заносу. Також була виявлена чутливість системи до змін параметрів, що вказує на необхідність уважного підходу до налаштування системи для оптимальної роботи в різних умовах.

Отримані результати становлять важливий внесок у розуміння та удосконалення систем управління мобільними платформами. Подальше вдосконалення системи та дослідження нових можливостей можуть призвести до ще більш ефективного та надійного управління такими системами у майбутньому.

3 КОНСТРУКТОРСЬКА ЧАСТИНА

3.1 Підготовка алгоритмів управління до реалізації на контролері

Оскільки галузь робототехніки стрімко розвивається, вчені розробили безліч способів впровадження алгоритмів, які забезпечують функціональність основного «тіла» мобільного робота – його навігації. Одні з найбільш популярних рухів робота включають обхід об'єктів та уникнення зіткнень в робочому просторі. Алгоритми для обходу перешкод у тривимірному просторі можна класифікувати за такими категоріями: гіпотеза-тест, штрафна функція, скелетний метод, нечітка логіка, нейронні мережі та генетичні алгоритми. В цьому відділі розглянемо кілька з цих алгоритмів.

Метод гіпотези та перевірки складається з трьох основних кроків: створення гіпотези про шлях-кандидата між початковою та кінцевою точками траєкторії рухомого робота; перевірка набору напрямків по цьому шляху на можливість зіткнення; якщо зіткнення можливе, то визначається обхід для уникнення перешкод, що можуть викликати це зіткнення. Весь процес повторюється до досягнення мети.

Другий клас алгоритмів обходу перешкод ґрунтується на створенні штрафної функції для конфігурації мобільного робота, яка вказує на наявність об'єктів. Каркасні алгоритми зменшують вільний простір робота до одновимірного представлення, що полегшує завдання планування шляху. Рис. 3.1 відображає схему комп'ютера, який управляє системою мобільного робота.

Керуючий сигнал, який поступає на контролер управління, є вхідним сигналом і передається оператором через бездротовий зв'язок Bluetooth. Режим роботи робота може бути автоматичним або ручним, залежно від типу отриманого сигналу. Наприклад, сигнал автоматичного керування може мати вигляд набору символів, таких як «@W2», де символ «@» ініціалізує контролер як початок алгоритму автоматичного керування, а символ «W» представляє рух платформи (наприклад, «вперед») на визначений час «2».

У випадку ручного керування мобільною платформою оператор може надсилати безперервний сигнал букви «Т», що вказує на відключення електродвигуна та паркування платформи на місці.

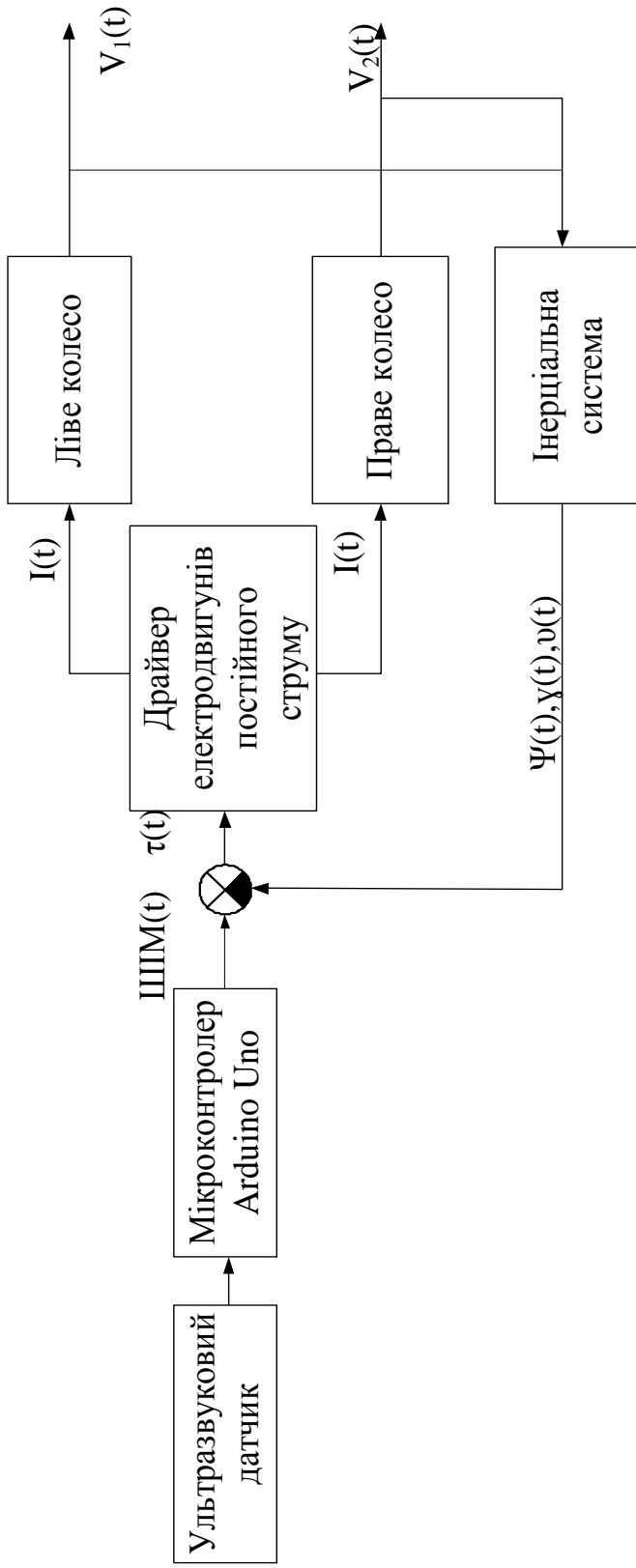


Рисунок 3.1 - Структурна схема СУ керуючого комп'ютера для мобільного робота

У разі push-передачі оператор може використовувати команди «W», «S», «A», «D» для руху платформи відповідно до напрямків «вперед», «назад», «вліво», «вправо» в графічному інтерфейсі мобільного додатка за допомогою відповідних кнопок.

Контролер управління обробляє отримані дані, формує сигнал ширини керуючого імпульсу для драйвера двигуна, який, в свою чергу, керує входною напругою двигуна згідно з обраним законом керування. Мікроконтролер-приймач отримує передані дані від мікроконтролера-відправника через радіоканал і передає їх через СОМ-порт ПК для графічного відображення в середовищі комп'ютера..

3.2 Вибір елементної бази та розробка принципової схеми обчислювача

Для реалізації керуючого комп'ютера в проєкті ми вибрали мікроконтролер із сімейства Arduino. Ця платформа апаратних обчислень складається з простої плати вводу/виводу та спеціального середовища розробки для обробки та програмування. Для даної мікросистеми було обрано мікроконтролер Arduino Uno, оскільки він володіє високою продуктивністю та відмінною ціною, що забезпечує оптимальний баланс характеристик.



Рисунок 3.2 – Контролер Arduino Uno

Технічні характеристики контролера наведені в табл.3.1.

Таблиця 3.1 – Основні характеристики

Мікроконтролер	Atmel ATmega168 або ATmega328
Робоча напруга	5 В
Вхідна напруга (рекомендований)	7-12 В
Вхідна напруга (граничне)	6-20 В
Цифрові Входи / Виходи	14 (6 з яких, можуть використовуватися як виходи ШІМ)
Аналогові входи	8
Постійний струм через вхід / вихід	40 мА
Постійний струм для виведення 3.3 В	50 мА
Флеш-пам'ять	16 Кб (ATmega168) або 32 Кб (ATmega328), при цьому 2 Кб використовуються для завантажувача
ОЗУ	1 Кб (ATmega168) або 2 Кб (ATmega328)
EEPROM	512 байт (ATmega168) або 1 Кб (ATmega328)
Тактова частота	16 МГц
Розміри	1.85 см x 4.2 см

Для симуляції використовується чотири двигуни з редуктором та відповідний для них драйвер на основі мікроконтролера L293D. SunFounder L293D - це чотириканальний драйвер, спроектований для роботи з великими струмами та напругами. Це вказує на те, що за допомогою мікросхеми L293D можна керувати двигунами постійного струму та використовувати джерела живлення до 36 вольт. Мікросхема L293D також гарантує максимальний струм до 600 мА на канал. L293D є варіантом Н-мостів, які зазвичай використовуються для управління подачею напруги (і її полярністю) на навантаження, наприклад, на двигун для контролю напрямку та швидкості його обертання. Схему платформи можна побачити на рис. 3.3.

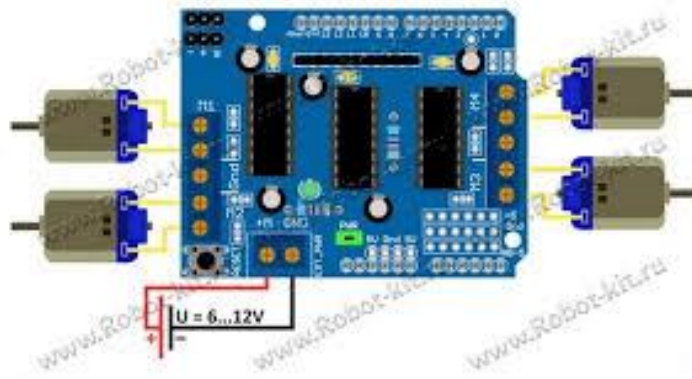


Рисунок 3.3 – Принципова схема складання мобільного робота

Для керування платформою використовується ультразвуковий далекомір HC-SR04 (див. рис. 3.4).

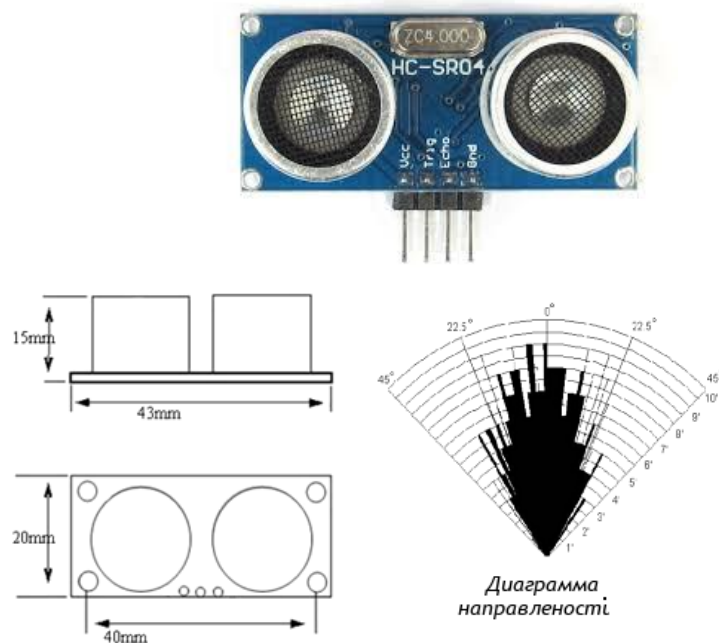


Рисунок 3.4 – Ультразвуковий далекомір HC-SR04, та його основні характеристики

Робота модуля ґрунтується на принципі ехолокації. Модуль генерує ультразвуковий сигнал і отримує його відбиття від об'єкта. Заслуговуючи час між відправленням та отриманням імпульсу, легко визначити відстань до перешкоди.

Основні характеристики :

- Напруга живлення: 5 В
- Низьке споживання: 2 мА

- Споживання електроенергії: 15 мА
- Діапазон відстаней: 2-400 см
- Ефективний кут огляду: 15 °
- Робочий кут огляду: 30°

Для функції інерційного навігаційного блоку було вирішено використовувати модуль MEMS MPU6050, який включає в себе 3-осьовий акселерометр та 3-осьовий гіроскоп, і керується за допомогою I2C (TWI).

Трьохосьовий гіроскоп та трьохосьовий акселерометр MPU-6050 мають доступну ціну та добру репутацію, тому були обрані як інерційні вимірювальні датчики. Вони використовуються для визначення положення в просторі в системах стабілізації положення, стабілізації прямолінійного руху та руху по заданій кривій.

Основою модуля є мікросхема MPU-6050. Для кращого розуміння розглянемо принципи роботи двох основних частин - акселерометра та гіроскопа, які входять до складу інерційного вимірювального датчика MPU 6050. Особливості MPU-6050:

1. Блок живлення:
 - а) напруга 3,7 - 5,5 В;
 - б) струм до 10 мА.
2. Гіроскоп споживає 3,6 мА, а в режимі очікування 5 мкА;
3. Акселерометр споживає 350 мкА;
4. У режимі очікування:
 - а) 10 мкА при 1,25 Гц;
 - б) 20 мкА при 5 Гц;
 - в) 60 мкА при 20 Гц;
 - г) 110 мкА при 40 Гц.
5. Максимальна частота інтерфейсу I2C становить 400 кГц;
6. Формат даних: кут Ейлера, кватерніон, матриця обертання або вихідні дані;
7. Діапазон гіроскопа: ± 250 , ± 500 , ± 1000 , ± 2000 градусів за секунду;
8. Діапазон акселерометра: ± 2 , ± 4 , ± 8 і $\pm 16g$;
9. 16-розрядний вихід даних;
10. Резонансна частота 27 кГц;
11. Шум $0,005^\circ/c/\sqrt{Гц}$;
12. Відстань між контактами 2,54 мм;

13. Отвір для гвинта 3 мм;
14. Розміри 20 x 16 мм.

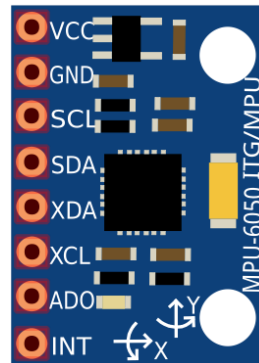


Рисунок 3.5 – Виводи MPU 6050 для під'єднання до Arduino

3.3 Розробка принципової схеми обчислювача

Принципова схема визначає електричне з'єднання між усіма радіодеталлями пристрою, надаючи уявлення про принципи його роботи та взаємозв'язок між окремими елементами і каскадами. Кожен елемент позначений звичайними графічними та літерними знаками, а серійний номер або блоку, або окремої частини в блоку, індивідуально вказується на схемі. Умовне буквене позначення деталі та її номер розташовані вище або ліворуч від графічного зображення. Типи деталей та їх порядкові номери на схемі визначаються в технічних даних пристрою. Для реалізації принципової схеми необхідно визначити всі функціональні елементи системи та основні взаємозв'язки між ними.

3.4 Вибір елементної бази для реалізації контролера

Arduino Uno є однією з найпоширеніших та простих у використанні платформ для вбудованих систем та прототипування. Основними характеристиками Arduino Uno є невеликий розмір, доступність, простий інтерфейс та зручність у програмуванні. У цьому розділі розглянемо, чому Arduino Uno є відмінним вибором для реалізації контролера для системи управління мобільною колісною платформою. Arduino Uno оснащений 14 цифровими входами/виходами, з яких 6 можуть використовуватися як аналогові входи. Це забезпечує достатню кількість портів для зчитування даних від датчиків, керування рухом коліс та комунікації з іншими пристроями.

Arduino Uno базується на мікроконтролері ATmega328P, який має тактову частоту 16 МГц. Це надає досить потужності для реалізації простих та середньої складності алгоритмів керування, таких як управління рухом та реалізація алгоритмів стабілізації.

Arduino Uno обладнаний UART для послідовної комунікації, що може бути використано для обміну даними з іншими пристроями або датчиками. Крім того, є можливість використання портів I2C та SPI для підключення додаткових пристроїв. Arduino Uno використовує просте середовище Arduino IDE для програмування, що зменшує поріг входження для новачків. Із загальнодоступними бібліотеками та прикладами коду в Інтернеті, розробник може швидко розпочати роботу над проектом без глибоких знань електроніки чи програмування. Arduino Uno користується великою та активною спільнотою, що полегшує вирішення проблем та отримання допомоги. Офіційна документація та форуми забезпечують доступ до важливої інформації та порад для ефективної розробки.

Arduino Uno виявляється відмінним вибором для контролера в системі управління мобільною колісною платформою завдяки своїм характеристикам, доступності та простоті використання. Він надає достатньо ресурсів для ефективного керування рухом та реалізації алгоритмів управління.

3.5 Формування алгоритму роботи обчислювача

Процес розробки програмного забезпечення розпочинається з створення комп'ютерних алгоритмів керування, які втілюють закони керування в системі. Далі відбувається розробка алгоритмів для роботи ЕОМ, що включає в себе алгоритми зчитування інформації з датчиків, передачі інформації виконавчим механізмам схем управління та можливість встановлення діалогу з ПК.

На рис. 3.6 наведено блок-схему алгоритму керуючого комп'ютера для прийому, перетворення та передачі даних.

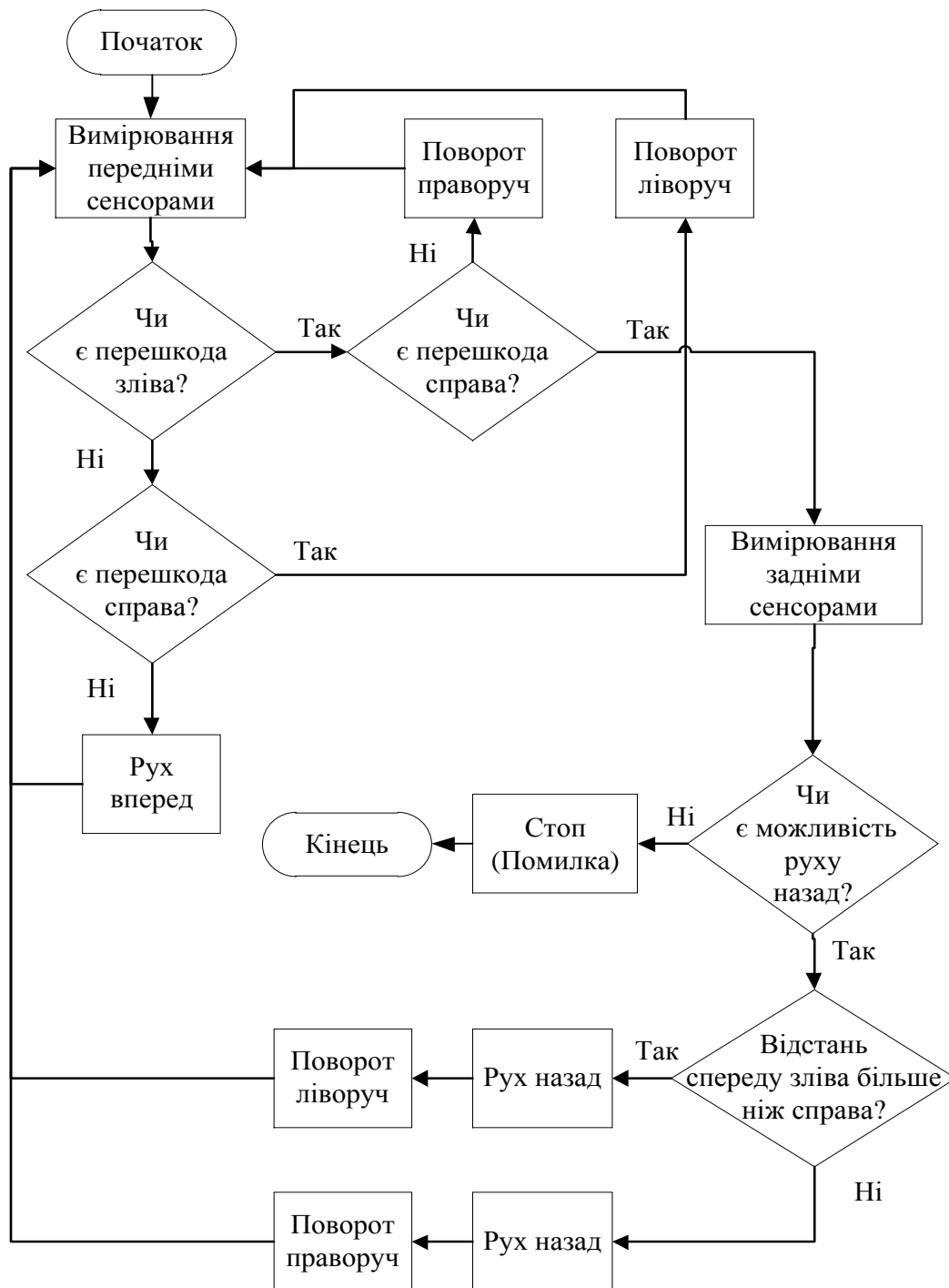
3.6 Розрахунок елементів принципової схеми контролера

У цьому розділі буде проведено розрахунки для основних елементів принципової схеми контролера Arduino Uno.

Мікроконтролер ATmega328P

Напруга живлення:

Arduino Uno працює при напрузі живлення від 7 до 12 вольт. Розглянемо використання напруги живлення 9 вольт для нашого розрахунку.



Під перешкодою розуміється досягнення безпечної відстані

Рисунок 3.6 – Блок-схема алгоритму роботи обчислювача

Кварцовий резонатор

Визначення частоти: Arduino Uno використовує кварцовий резонатор з частотою 16 МГц. Це важливо врахувати при розрахунках, пов'язаних із зв'язком та розрахунками часу.

Резистори та конденсатори

Стабілізація живлення та фільтрація сигналів: Використаємо конденсатори для фільтрації та стабілізації живлення. Розрахунок резисторів необхідних для RC-фільтрації може бути проведений за формулою:

Нехай

$$C_{filter} = 10\mu F \text{ та } C_{stabilize} = 1\mu F$$

$$R = \frac{1}{2\pi f C}$$

Де f - частота сигналу. У нашому випадку $f = 16 \times 10^6$ Гц.

$$R_{filter} = \frac{1}{2\pi \times 16 \times 10^6 \times 10 \times 10^{-6}}$$

$$R_{stabilize} = \frac{1}{2\pi \times 16 \times 10^6 \times 1 \times 10^{-6}}$$

$$R_{filter} \approx 1\Omega$$

$$R_{stabilize} \approx 10\Omega$$

Виберемо стандартні резистори $R_{filter} = 1\Omega$ та $R_{stabilize} = 10\Omega$.

Порти введення/виведення

Вибір внутрішніх резисторів для портів: У випадку використання внутрішніх резисторів для портів введення/виведення, необхідно враховувати їх опір. Для вхідних портів внутрішній резистор може бути від 20 до 50 кОм.

Проведено розрахунки для основних елементів схеми контролера Arduino Uno, забезпечуючи стабільну роботу та враховуючи особливості живлення та введення/виведення. Обрано відповідні резистори та конденсатори для забезпечення необхідного фільтрування та стабілізації.

3.7 Обґрунтування конструкції блока контролера

Вибір Arduino Uno для реалізації блока контролера обґрунтований кількома ключовими факторами:

1. Універсальність та доступність: Arduino Uno є популярною та доступною платформою для розробки вбудованих систем. Її можна легко знайти на ринку, і вона вже здобула велику популярність серед розробників.

2. Легкість програмування: Arduino Uno використовує Arduino IDE, що є простим та зручним середовищем для програмування, особливо для початківців. Це дозволяє швидко розробляти та тестувати програмне забезпечення.

3. Велика спільнота користувачів: Arduino Uno користується великою та активною спільнотою, що полегшує вирішення проблем, обмін досвідом та використання вже готових рішень.

4. Достатня обчислювальна потужність: Для багатьох застосувань, пов'язаних із керуванням рухом та обробкою сигналів, потужності мікроконтролера ATmega328P виявляються достатніми.

Конструкція блока контролера розроблятиметься з урахуванням компактності та мобільності системи управління мобільною колісною платформою. Arduino Uno має невеликі розміри та може бути легко інтегрований в обмежені просторові умови, що є критичним для роботи на мобільних платформах.

При розробці конструкції блока контролера буде враховано можливість розширення функціональності у майбутньому. Arduino Uno має додаткові порти введення/виведення, а це надає можливість підключення додаткових датчиків, модулів та розширення можливостей системи без заміни основного блока контролера.

Конструкція блока контролера буде спроектована з урахуванням легкості обслуговування та можливості ремонту. Модульний підхід дозволить легко замінювати або оновлювати окремі компоненти без необхідності повного відновлення системи.

Вибір Arduino Uno та підхід до конструкції блока контролера обумовлені потребами проекту у простоті, доступності, мобільності та легкості обслуговування. Це забезпечить ефективну та надійну роботу системи управління мобільною колісною платформою.

3.8 Висновки за розділом

У даній частині розроблено алгоритм роботи системи керування мобільною колісною платформою та проведено дослідження режиму керованого заносу. Обрано мікроконтролер Arduino для реалізації завдань обробки та управління двигунами мобільної платформи.

Вибір мікроконтролера Arduino обґрунтований його універсальністю та доступністю, які важливі для ефективного реалізації системи керування. Arduino Uno володіє легкістю у програмуванні та широким спектром можливостей, що дозволяє швидко розробляти та тестувати програмне забезпечення для управління мобільною платформою.

Алгоритм роботи системи керування спроектовано так, щоб ефективно виконувати завдання навігації та керування мобільною колісною платформою між перешкодами. Дослідження режиму керованого заносу дозволяє покращити стійкість та точність руху системи в різних умовах експлуатації.

Отже, обрана концепція, включаючи мікроконтролер Arduino та використання вивчених алгоритмів, має потенціал для успішної реалізації системи керування мобільною колісною платформою з високим рівнем ефективності та керованості.

4 ДОСЛІДНИЦЬКА ЧАСТИНА

4.1 Задачі і методи проведення досліджень

У сучасному світі, де роботизація та автономні технології набувають все більшого значення, системи управління мобільними платформами є об'єктом інтенсивних досліджень. Однією з ключових аспектів цього дослідження є аналіз та оптимізація режиму керованого заносу, який може важливо впливати на маневреність і стійкість рухомого об'єкта.

Об'єктом нашого дослідження є системи управління мобільною колісною платформою. Важливо вивчати та оптимізувати процеси керування, щоб досягти оптимальної продуктивності, точності та ефективності таких систем.

Аналіз різних аспектів систем управління, включаючи обрані алгоритми та використання мікроконтролера Arduino. Оцінка їх ефективності та можливостей управління рухом.

Вивчення явища керованого заносу, коли мобільна платформа намагається розвертатися при керуванні. Аналіз впливу параметрів системи та алгоритмів керування на стійкість та точність руху.

Механізм руху за допомогою заносу (skid steering) широко використовується для управління рухом мобільних роботів. Для роботів, які використовують занос, не передбачено керма, і напрямок руху змінюється за рахунок різниці обертання лівого та правого колісного пар. Згідно з нетривіальною взаємодією між колесами та поверхнею та кінематичними обмеженнями, виникають труднощі у побудові точної кінематичної та динамічної моделі для роботів із керованим заносом.

Колісне ковзання відіграє вирішальну роль у кінематичній та динамічній моделі робота з керованим заносом. Здатність оцінювати інформацію про колісне ковзання є важливою для навігаційних завдань. Точність навігації робота знижується, коли ковзання коліс стає помітним. Інформація про колісне ковзання також відіграє важливу роль у динаміці управління роботом.

Взаємодія між колесами та поверхнею також безпосередньо впливає на сили прискорення та гальмування, що має негативний вплив на стійкість руху та маневреність робота.

4.2 Програма проведення досліджень

Для створення кінематичної моделі робота, який керується заносом, зробимо припущення, що рух відбувається на рівній поверхні. Визначимо глобальну ($I(X_g, Y_g, Z_g)$) та локальну ($B(x, y, z)$) систему координат, пов'язану з центром мас платформи робота (рис. 4.1). Координати центра мас позначимо через (X, Y, Z) , де $Z = \text{const}$ через припущення про плоский рух робота.

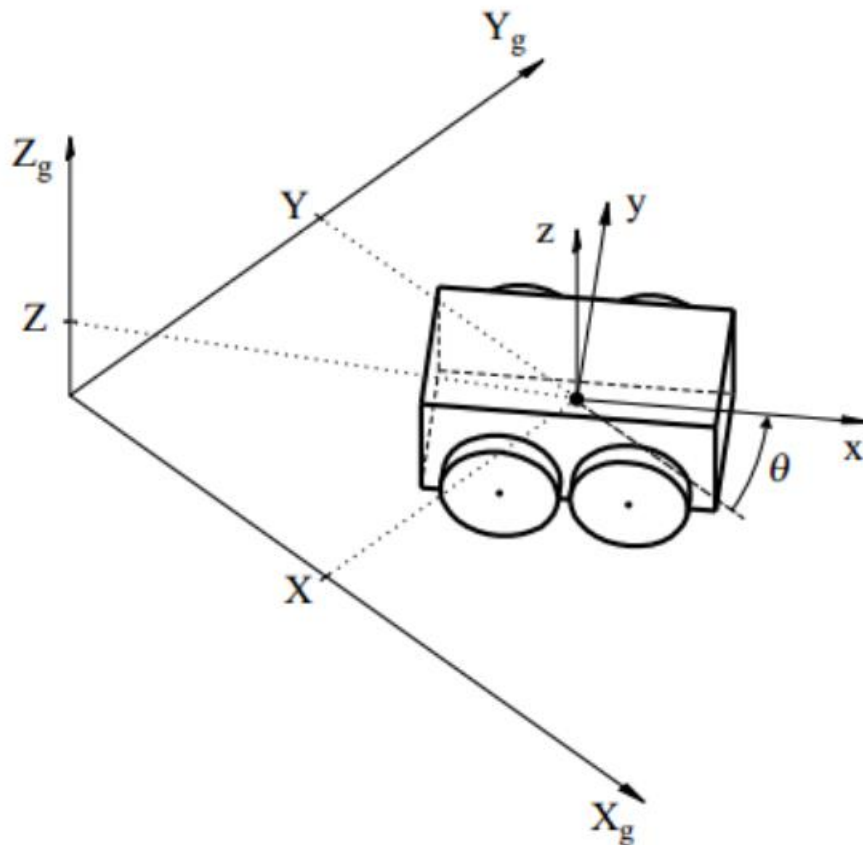


Рисунок 4.1 – Зображення робота в глобальній системі координат

Припустимо, що швидкість руху робота на плоскості у локальній системі координат $B(x, y, z)$ дорівнює $v = (v_x, v_y, 0)^T$, а кутова швидкість - $\omega = (0, 0, \omega)^T$. Назвемо вектором узагальнених координат робота $q = (X, Y, \Theta)^T$, де X та Y - координати центру мас робота, а Θ - обертання локальної системи координат B відносно глобальної системи координат I . Тоді вектор $q' = (X', Y', \Theta')^T$ позначає вектор узагальнених швидкостей. З рис. 3 легко зрозуміти, що швидкості X' та Y' у глобальній системі координат I пов'язані із швидкостями v_x та v_y у локальній системі координат так:

$$\begin{pmatrix} X' \\ Y' \end{pmatrix} = \begin{pmatrix} \cos\Theta & -\sin\Theta \\ \sin\Theta & \cos\Theta \end{pmatrix} \begin{pmatrix} v_x \\ v_y \end{pmatrix}$$

Оскільки рух відбувається у плоскості, легко зрозуміти, що $\Theta' = \omega$.

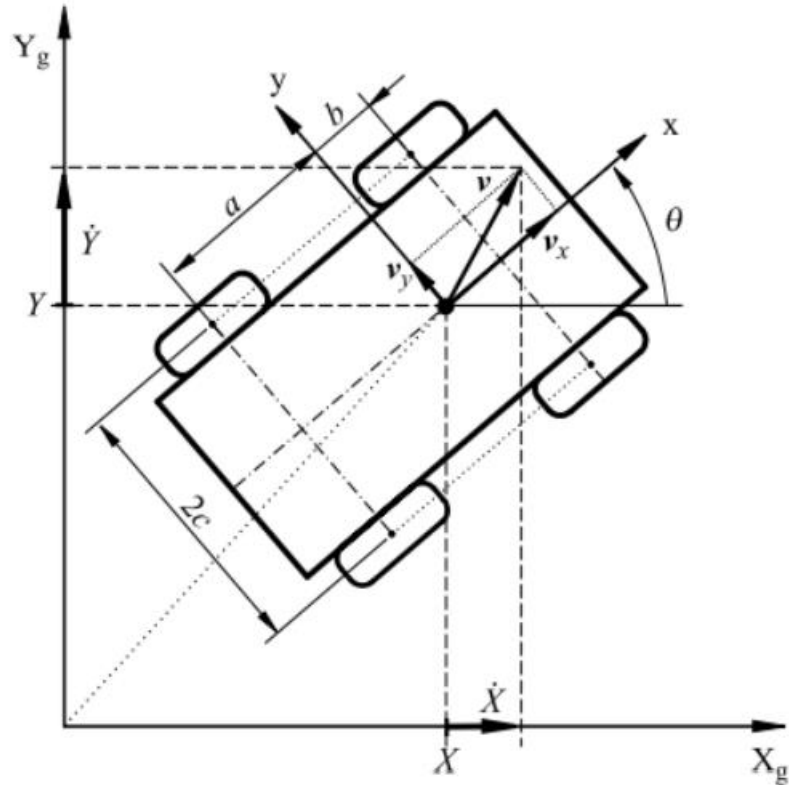


Рисунок 4.2 – Зв'язок між швидкістю платформи в глобальній та локальній системі координат

Очевидно, що рівняння 1 не накладає жодних обмежень, пов'язаних з конструкцією робота, і лише описує кінематику вільно рухомого об'єкта. Отже, необхідно встановити кінематичні відносини, пов'язані з конструктивними особливостями.

Припустимо, що i -те колесо обертається з кутовою швидкістю $\omega_i(t)$, де $i = 1, \dots, 4$. Для спрощення моделі припустимо, що товщина коліс є незначною, і колесо торкається поверхні в точці P_i , як показано на рис. 4.3. В загальному випадку, на відміну від більшості колісних платформ, поперечна швидкість v_{iy} робота, керованого заносом, не дорівнює нулю. Ця особливість, виникаючи через конструктивні особливості платформи, дозволяє роботу змінювати орієнтацію у просторі за рахунок поперечного ковзання.

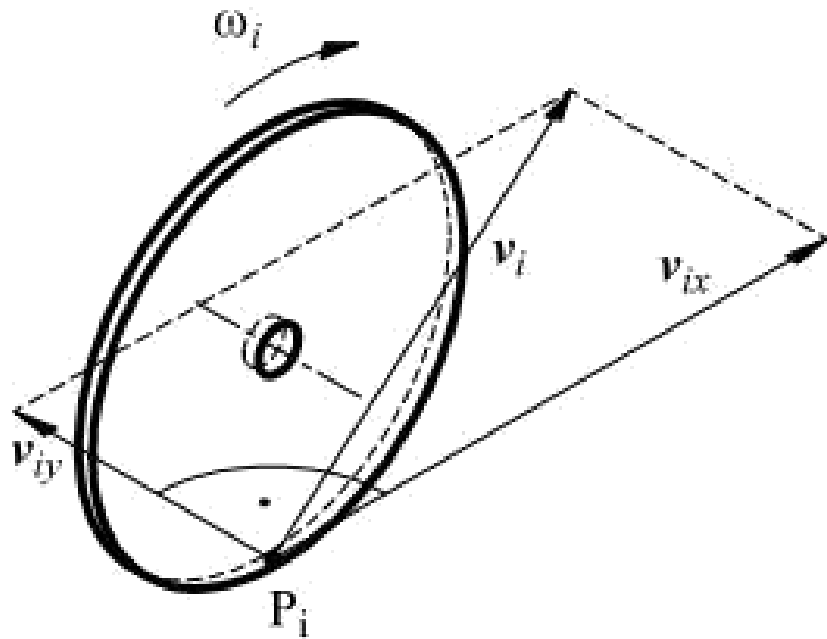


Рисунок 4.3 – Компоненти швидкості колеса у локальній системі координат

У даній моделі передбачається описати тільки найпростіший випадок переміщення робота, керованого занесенням, в якому поздовжнє ковзання між колесами і поверхнею зневажливо. Згідно з цим припущенням можна отримати, що:

$$v_{ix} = r\omega_i,$$

де v_{ix} поздовжня компонента швидкості в локальній системі координат, а r - Ефективний радіус колеса.

Для складання кінематичної моделі необхідно враховувати обертання кожного колеса робота. Визначимо радіус-вектора $d_i = (d_{ix}, d_{iy})_T$ та $d_C = (d_{Cx}, d_{Cy})_T$, як показано на рис.4.4. Використовуючи прості геометричні співвідношення, зображені на рис. 4.4 легко виходить що:

$$\frac{\|v_i\|}{\|d_i\|} = \frac{\|v\|}{\|d_C\|} = \|\omega\|$$

або в іншій формі:

$$\frac{v_{ix}}{-d_{iy}} = \frac{v_x}{-d_{Cy}} = \frac{v_{iy}}{d_{ix}} = \frac{v_y}{d_{Cx}} = \omega.$$

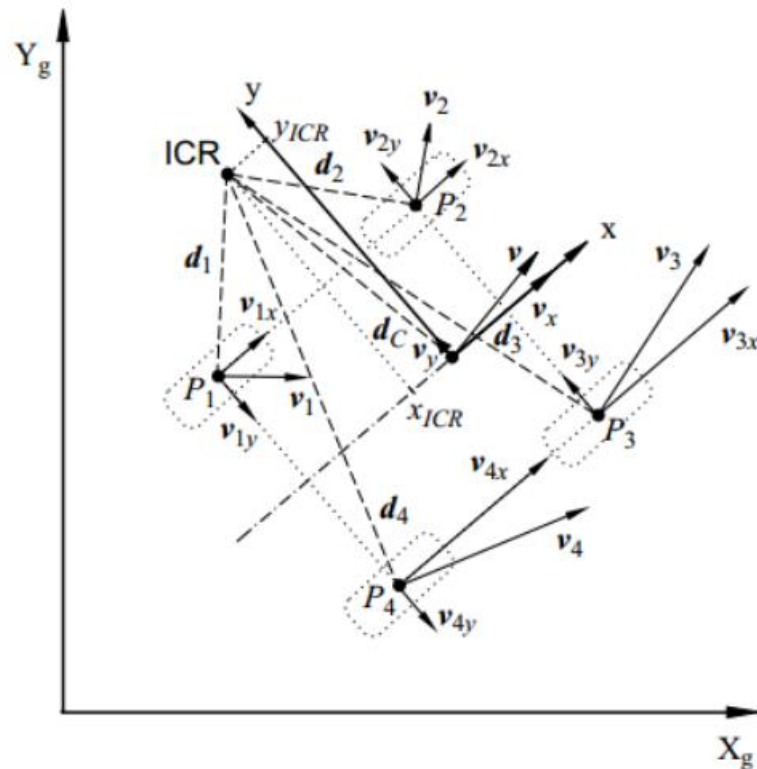


Рисунок 4.4. – Конструктивні зв'язки робота

Визначаючи координати миттєвого центру обертання локальної системи координат, ми можемо переписати рівняння:

$$ICR = (x_{ICR}, y_{ICR})^T = (-d_{xC}, d_{yC})^T$$

З рис. 4.4 бачимо, що координати вектора d_i задовольняють наступним рівнянням:

$$\begin{aligned} d_{1y} &= d_{2y} = d_{Cy} + c \\ d_{3y} &= d_{4y} = d_{Cy} - c \\ d_{1x} &= d_{4x} = d_{Cx} - a \\ d_{2x} &= d_{3x} = d_{Cx} + b, \end{aligned}$$

де a , b , c - додатні параметри

$$v_L = v_{1x} = v_{2x}$$

$$v_R = v_{3x} = v_{4x}$$

$$v_F = v_{1y} = v_{4y}$$

$$v_B = v_{2y} = v_{3y},$$

де v_L та v_R позначають продольну швидкість правої та лівої пари коліс, а v_F та v_B - поперечну швидкість передньої та задньої пари коліс, відповідно.

Використовуючи рівняння, можливо отримати наступне перетворення, що описує взаємозв'язок між швидкістю коліс і швидкістю робота:

$$\begin{pmatrix} v_L \\ v_R \\ v_F \\ v_B \end{pmatrix} = \begin{pmatrix} 1 & -c \\ 1 & c \\ 0 & -x_{ICR} + b \\ 0 & -x_{ICR} - a \end{pmatrix}$$

Припускаючи, що ефективний радіус для кожного з коліс r_i дорівнює r , можемо написати:

$$\omega_\omega = \begin{pmatrix} \omega_L \\ \omega_R \end{pmatrix} = \frac{1}{r} \begin{pmatrix} v_L \\ v_R \end{pmatrix}$$

де ω_L та ω_R - кутові швидкості лівої та правої пар коліс відповідно.

Об'єднавши рівняння, можна отримати наступні рівності між кутовими швидкостями колісних пар платформи та швидкістю робота:

$$\begin{pmatrix} v_x \\ \omega \end{pmatrix} = r \begin{pmatrix} \frac{\omega_L + \omega_R}{2} \\ \frac{-\omega_L + \omega_R}{2c} \end{pmatrix}$$

4.3 Результати досліджень

Проведені серії експериментів щодо вивчення стійкості траєкторії робота включали вимірювання на круговій траєкторії при різних кутових швидкостях колісних пар, під час руху робота по прямій траєкторії вперед-назад. Під час руху платформи по круговій траєкторії були обрані співвідношення швидкостей 2:1 і 3:1 для значень $1,57 \text{ c}^{(-1)}$, $1,25 \text{ c}^{(-1)}$, $1,04 \text{ c}^{(-1)}$ як опорних. Для вимірювань руху робота по другій траєкторії була обрана кутова швидкість $1,57 \text{ c}^{(-1)}$. Окремо був розглянутий випадок руху по круговій траєкторії заблокованою однією з колісних пар, швидкість іншої становила $1,57 \text{ c}^{(-1)}$.

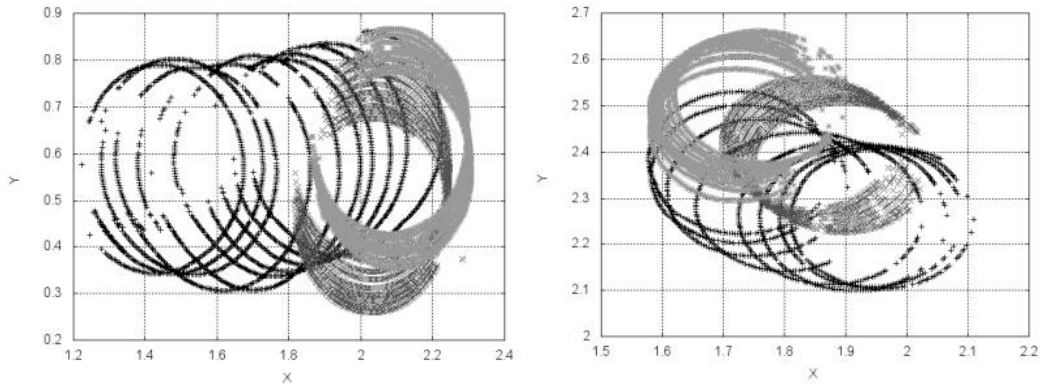


Рисунок 4.5 – Траєкторії платформи при програмуванні руху по колу

Співвідношення швидкостей колісних пар 2:1. Опорні швидкості рівні: для чорної - $1,57 \text{ c}^{-1}$, для темно-сірої - $1,25 \text{ c}^{-1}$, для світло-сірої лінії - $1,04 \text{ c}^{-1}$.

На основі отриманих експериментальних даних були побудовані траєкторії руху робота (рис. 4.6), а також розраховані відхилення експериментальних даних від теоретичних. Експериментальне значення радіуса при відношенні кутових швидкостей 2:1 становить 21 см. Виміряні значення лінійної швидкості робота складають 2,73 см/с, 2,23 см/с, 1,81 см/с при опорних швидкостях $1,57 \text{ c}^{-1}$, $1,25 \text{ c}^{-1}$, $1,04 \text{ c}^{-1}$. Для відношення 3:1 відповідні значення радіуса становлять 15,5 см, а швидкості - 2,45 см/с, 1,87 см/с, 1,66 см/с.

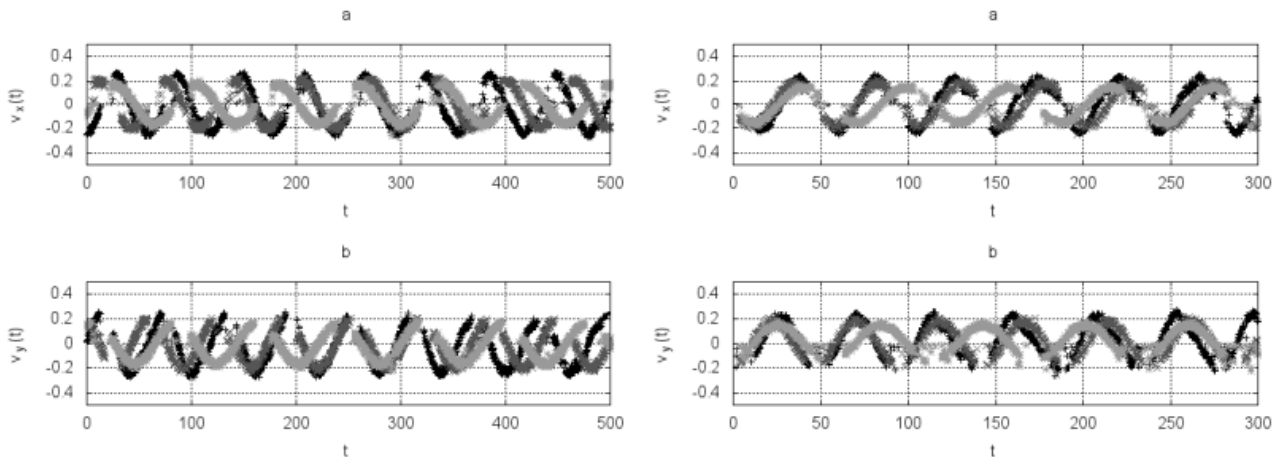


Рисунок 4.6 – Компоненти швидкості робота в глобальній системі координат при відношенні швидкостей 2:1 та 2:2. Швидкість виміряна в 10^{-1} см/с.

Проаналізуємо рис. 4.6 для кругової траєкторії при відношенні кутових швидкостей коліс 2:1 та опорних швидкостей $1,57 \text{ c}^{-1}$, $1,25 \text{ c}^{-1}$, $1,04 \text{ c}^{-1}$. Відсоткове відхилення експериментально виміряної швидкості робота від теоретичної складає 12,34%, 13,19%, 9,39% відповідно. Водночас відхилення

значення радіуса становить 33,33%. Для відношення швидкостей 1:3 це відхилення становить 11,87%, 6,85%, 13,69%, а радіуса - 47,6%. Це викликано тим, що реально центр мас робота знаходився не на вісі симетрії платформи, що вимагає подальших коригувань у моделі робота.

Співвідношення швидкостей колісних пар 2:1. Опорні швидкості рівні: для червоної - $1,57 \text{ c}^{-1}$, для зеленої - $1,25 \text{ c}^{-1}$, для синьої лінії - $1,04 \text{ c}^{-1}$.

Щоб описати стійкість траєкторії робота, проаналізуємо переміщення центру обертання (рис. 4.7).

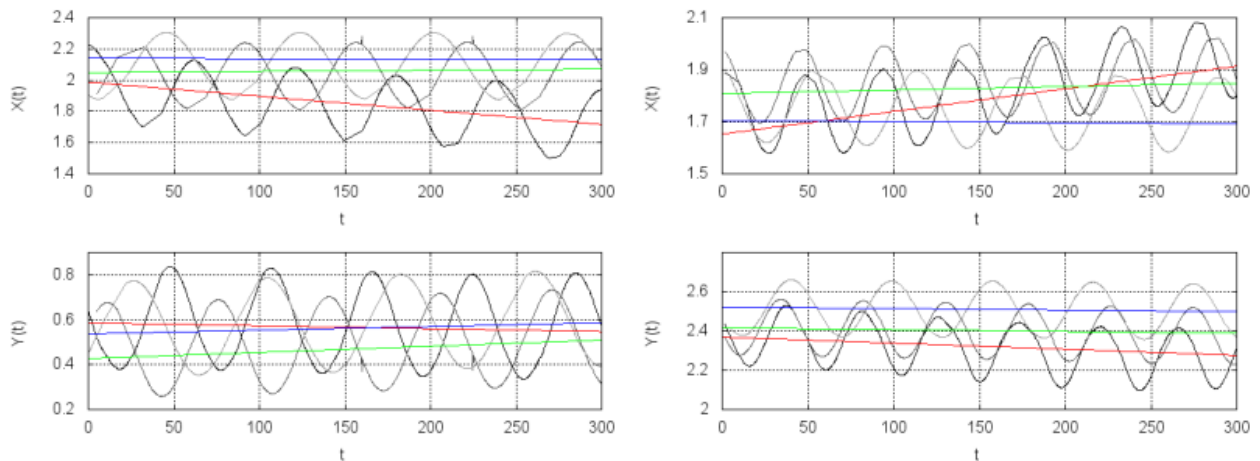


Рисунок 4.7 – Траєкторії відхилення центрів обертання

Проведемо лінійну апроксимацію його відхилення по координатах X і Y , отримаємо для відношення швидкостей 2:1

$$X_{1,57c^{-1}}(t) = -0,000901 * t + 1,99$$

$$Y_{1,57c^{-1}}(t) = -0,000128 * t + 0,59$$

$$X_{1,25c^{-1}}(t) = -0,000071 * t + 2,05$$

$$Y_{1,25c^{-1}}(t) = -0,000273 * t + 0,43$$

$$X_{1,04c^{-1}}(t) = -0,000025 * t + 2,14$$

$$Y_{1,04c^{-1}}(t) = -0,000163 * t + 0,54$$

для відношення швидкостей 3:1

$$X_{1,57c^{-1}}(t) = -0,000869 * t + 1,65$$

$$Y_{1,57c^{-1}}(t) = -0,000301 * t + 2,37$$

$$X_{1,25c^{-1}}(t) = -0,000132 * t + 1,81$$

$$Y_{1,25c^{-1}}(t) = -0,0000967 * t + 2,42$$

$$X_{1,04c^{-1}}(t) = -0,0000415 * t + 1,70$$

$$Y_{1,04c^{-1}}(t) = -0,0000759 * t + 2,52$$

Тоді швидкість зміни координати центра дорівнює dt . Модуль швидкості відхилення положення центра обертання становить 0,091 см/с, 0,0282 см/с, 0,0164 см/с для відношення швидкостей 1:2 і 0,0921 см/с, 0,0179 см/с, 0,0086 см/с для відношення швидкостей 1:3.

Цікаво розглянути граничний випадок, коли одна з колісних пар заблокована. В цьому випадку відхилення експериментально виміряної від теоретичної швидкості становить 14,63% при абсолютному значенні теоретичної 1,64 см/с, а відхилення радіуса - 4,76% при теоретичному значенні 5,25 см. Швидкість відхилення центра обертання в даному випадку дорівнює 0,05 см/с.

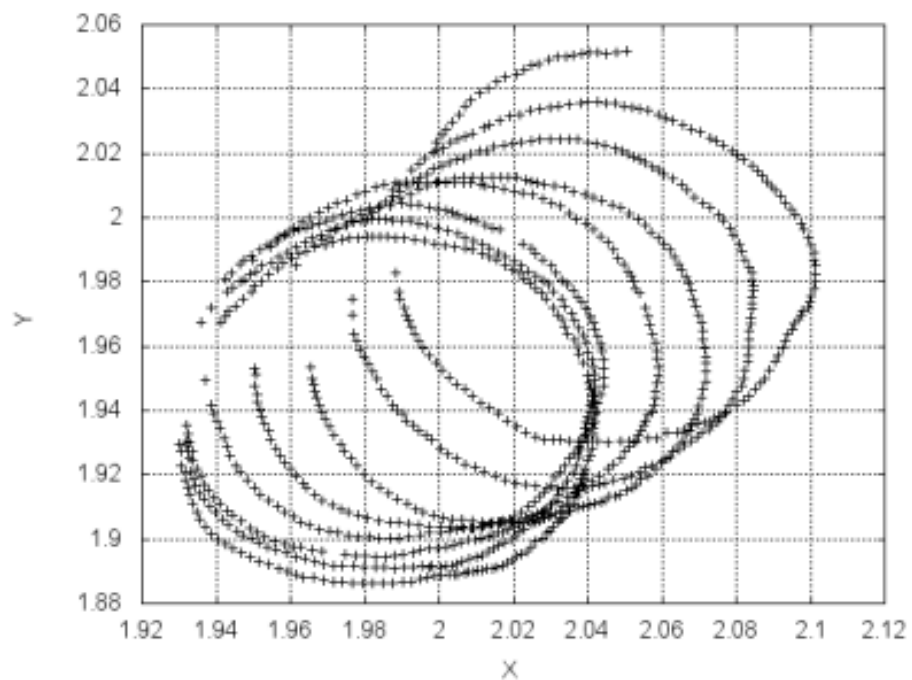


Рисунок 4.8 – Траєкторія робота при швидкості правої колісної пари $1,57c^{-1}$ і заблокованої лівої.

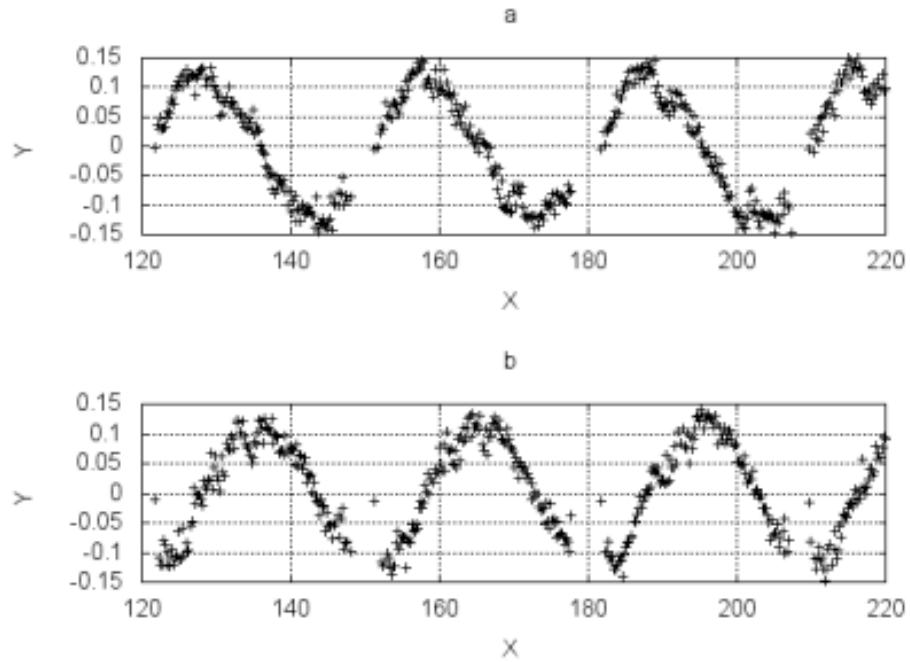


Рисунок 4.9 – Компоненти швидкості робота в глобальній системі координат. Швидкість виміряна в 10^{-1} см/с.

При руху платформи по прямому відрізку вперед-назад відношення довжини руху до ширини траєкторії складає 15,69% при проходженні маршруту 12 разів, видно, що при русі платформи по прямому відрізку траєкторії спостерігається відхилення від початково заданої траєкторії на 0,02%.

Цей негативний ефект викликаний неповною ідентичністю двигунів. Крім того, при зміні напрямку руху платформа повертається на невеликий кут, що подальше руйнує початково задану траєкторію руху. Відхилення реальної швидкості від програмованої становить 6%, при абсолютних значеннях теоретичної 3,24 см/с.

4.4 Аналіз результатів досліджень

Результати проведених експериментів вказують на ефективність та функціональні можливості розробленої системи керування мобільною колісною платформою з дослідженням режиму керованого заносу. Отримані дані важливі для оцінки властивостей системи та розробки подальших поліпшень (рис. 4.10 та 4.11).

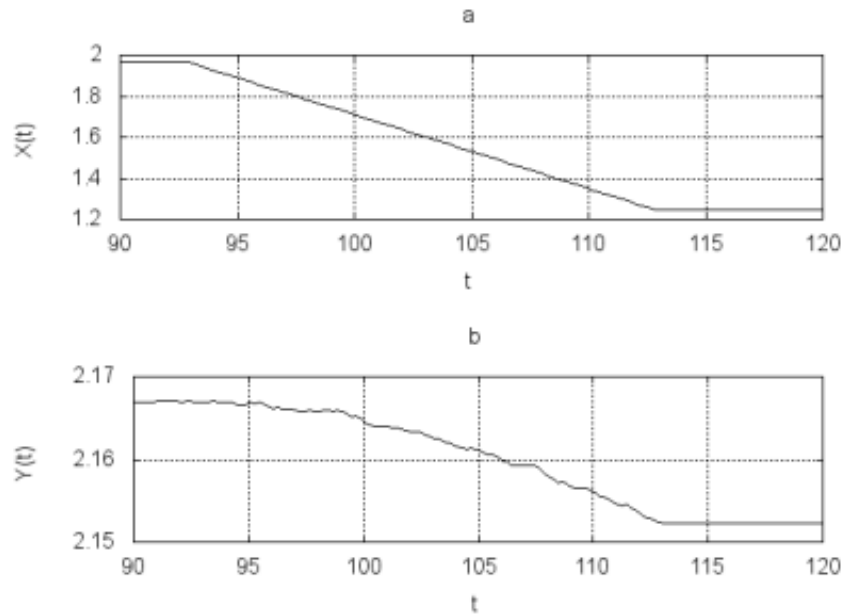


Рисунок 4.10 – Зміна глобальних координат платформи при русі по прямій лінії довжиною 68,2 см.

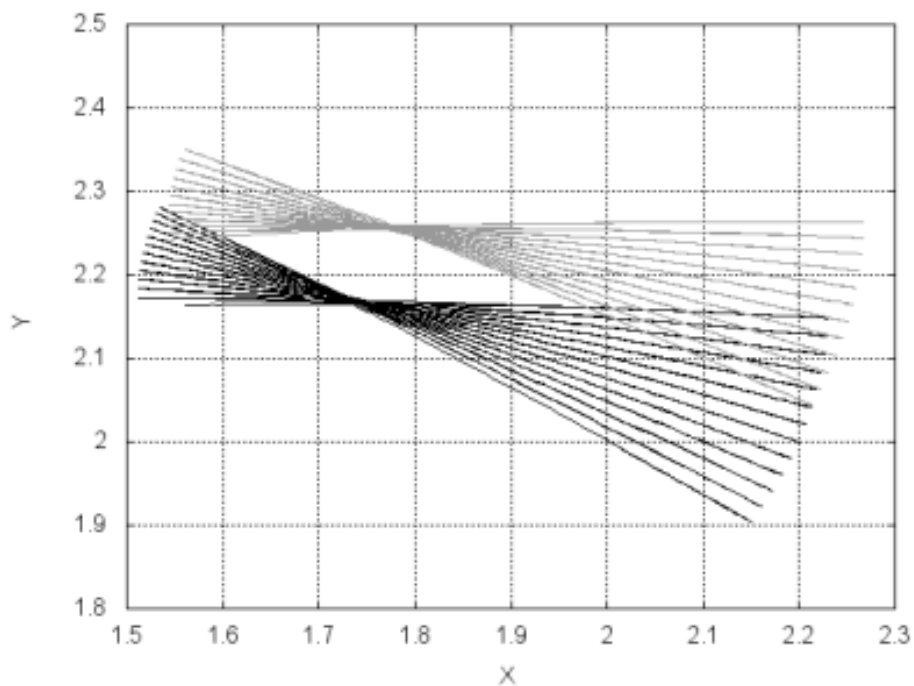


Рисунок 4.11 – Траєкторії платформи при програмуванні руху вперед-назад

Перше, що слід відзначити, це стійкість та точність руху робота при реалізації керованого заносу. Система керування забезпечує високу стабільність траєкторії, як під час руху по прямій лінії, так і при обході поворотів.

Дослідження траєкторій під час руху по круговій траєкторії з різними відношеннями швидкостей коліс підтверджує відмінну реалізацію керованого заносу. Зміщення центру обертання та відхилення від ідеальної траєкторії є природними властивостями, але залишаються прийнятними для багатьох застосувань.

Аналіз результатів показав, що система виявляє певні відхилення від теоретичних значень при великих відношеннях швидкостей коліс. Це може бути пов'язано з обмеженнями фізичних характеристик системи, таких як неідеальність двигунів та затримки в керуванні.

Крім того, під час руху по прямій лінії вперед-назад виявлено деякі відхилення від ідеальної траєкторії через неповну ідентичність моторів та обмеження конструкції. Зазначено, що під час зміни напрямку руху виникає невеликий зсув траєкторії.

Загалом, результати свідчать про ефективність системи керування мобільною колісною платформою та підкреслюють необхідність подальших досліджень та вдосконалень для досягнення максимальної точності та стійкості у всіх режимах руху.

4.5 Висновки за розділом

Керовані мобільні платформи з заносом в наш час отримали широке застосування в різних галузях людської діяльності. Як наслідок, виникла потреба в дослідженнях з удосконалення та акуратному вивченні існуючих кінематичних та динамічних моделей для цього типу роботів. У даній роботі проведено дослідження стійкості траєкторії робота для кінематичної моделі плаского руху з урахуванням тільки кінематичних та конструктивних зв'язків.

Отримано висновок, що ступінь відхилення (відношення зсуву точки повороту від початкової до довжини відрізка руху) для руху вперед-назад становить лише 0,02%. Швидкість зсуву центра обертання для кругової траєкторії платформи оцінена як 0,91 см/с, 0,282 см/с, 0,164 см/с при співвідношенні швидкостей 2:1 та 0,921 см/с, 0,179 см/с, 0,086 при співвідношенні швидкостей 3:1 для опорних швидкостей $1,57 \text{ с}^{-1}$, $1,25 \text{ с}^{-1}$,

$1,04 \text{ с}^{-1}$. Заблокувавши ліву колісну пару та задавши обертову швидкість правої $1,57 \text{ с}^{-1}$, отримано значення 0,05 см/с.

5 ЕКСПЕРИМЕНТАЛЬНО-ПРАКТИЧНА ЧАСТИНА

5.1 Задачі і засоби виконання експериментальної розробки

Основною метою будь-якого експерименту є не лише здобуття нової інформації про об'єкт управління, але й точне визначення закономірностей його поведінки в змінних умовах експлуатації, аналогічних умовам його розвитку в природі, техніці, суспільному житті і т. д.

У цьому розділі детально розглядається функціонування розробленого мобільного робота. Для здійснення експерименту була використана власноруч розроблена платформа, обладнана відповідним сервісним обладнанням. Також використовувався персональний комп'ютер із програмним забезпеченням для моніторингу та аналізу даних щодо трафіку, а мобільний Android-пристрій був обладнаний спеціальним мобільним додатком для управління рухом робота.

Експериментальне дослідження даної розробки проводилося на реальній установці і передбачало бездоганне вирішення таких завдань:

1. Автоматичне управління за допомогою вбудованих датчиків;
2. Управління за допомогою мобільного додатка та модуля Bluetooth із залученням людського фактору;
3. Відображення інформації про температуру, вологу, відстань до об'єкта, своє положення в графічному інтерфейсі мобільного додатка, а також виявлення несправностей.

5.2 Опис лабораторної установки

Експериментальна частина виконується на пристрої, що представляє собою чотириколісну платформу, на якій розташовані всі робочі пристрої. Основа створена з пластику, який використовується у вигляді пластини, аналогічної до тих, що використовуються у дитячих іграшках. На основі розташована розвиваюча плата, на яку кріпляться інші елементи. Передні ведучі колеса закріплені на валу двигуна. Додатково на платформі встановлений ультразвуковий датчик. Енергію системи забезпечують дві батарейки 18650.

Рух мобільного робота відбувається завдяки двигунам постійного струму, керування якими здійснюється за допомогою драйвера, заснованого на

мікроконтролері L293D. Цей драйвер з'єднаний з контролером Atmega328p, який розташований на платформі Arduino Uno.

Для управління платформою використовується ультразвуковий далекомір HC-SR04. З рис. 5.1-5.2 можна отримати уявлення про остаточну конфігурацію платформи.

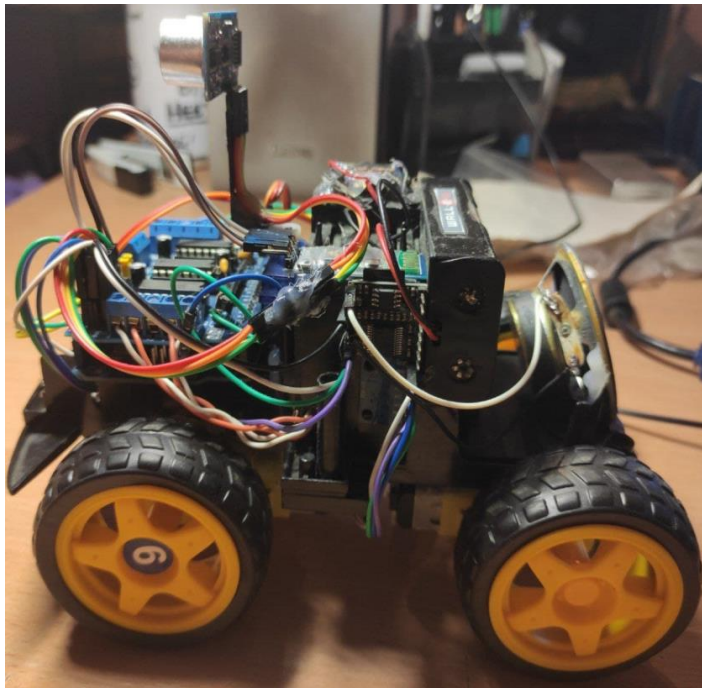


Рисунок 5.1 – Розроблена платформа збоку

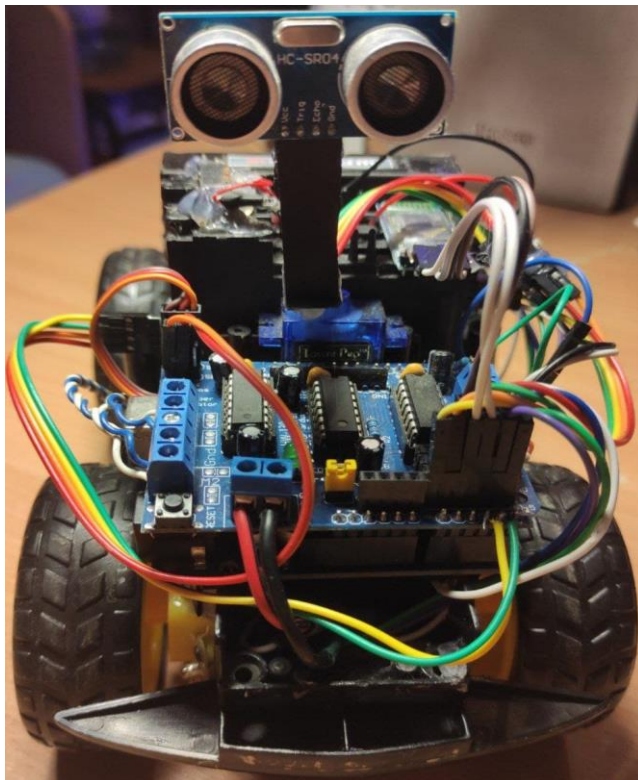


Рисунок 5.2 – Розроблена платформа спереду

Перед початком експерименту необхідно впевнитися в дотриманні наступних умов:

1. Підключення Приймального Пристрою:

Переконайтеся, що приймальний пристрій належним чином підключено до одного з СОМ-портів персонального комп'ютера.

2. Запуск Спеціально Розробленої Комп'ютерної Програми:

Відкрийте спеціально розроблену комп'ютерну програму для візуалізації даних, які надходять від відправника, на комп'ютері.

3. Стан Провідників Системи:

Переконайтеся, що усі провідники системи не пошкоджені та ізольовані.

4. Увімкнення Мобільного Робота:

Увімкніть живлення мобільного робота.

5. Перевірка Роботи Елементів Системи:

Переконайтеся, що всі елементи системи працюють належним чином.

6. З'єднання Провідників:

Впевніться, що усі дроти та провідники правильно з'єднані на визначених місцях.

7. Фіксація Компонентів:

Переконайтеся, що всі компоненти системи надійно закріплені на своїх місцях.

8. Стан Корпусу Платформи Мобільного Робота:

Переконайтеся, що корпус платформи мобільного робота не має змін або пошкоджень.

9. Увімкнення Спеціально Розробленої Платформної Програми на Мобільному Телефоні:

Увімкніть спеціально розроблену платформну програму дистанційного керування на мобільному телефоні.

10. Бездротовий Зв'язок:

Переконайтеся, що бездротовий зв'язок з платформою працює належним чином.

Ці кроки допоможуть гарантувати правильне функціонування системи та нададуть можливість здійснити експеримент без перешкод.

5.3 Характеристика розробленого програмного забезпечення

Метод вивчення функціональної автоматичної системи керування мобільним роботом можна уявити як послідовність операцій для забезпечення ефективного проведення експерименту:

1. Початок Роботи ПК та Комп'ютерної Програми:

Запустіть комп'ютер та ініціалізуйте комп'ютерну програму, яка призначена для відображення та аналізу результатів експерименту.

2. Підключення до Системного Приймача:

Підключіться до одного з СОМ-портів ПК та перевірте належну роботу системного приймача.

3. Увімкнення Живлення та Мобільного Робота:

Увімкніть живлення всіх компонентів на платі мобільного робота, готуючи його до подальшого використання.

4. Активація Android-пристрою для Управління:

Увімкніть Android-пристрій, спеціально призначений для дистанційного керування мобільним роботом.

5. Бездротове Підключення через Bluetooth:

Здійсніть бездротове підключення між платформою та системою керування за допомогою технології Bluetooth.

6. Трансляція Керуючих Сигналів:

Передавайте керуючі сигнали на платформу, гарантуючи зв'язок та взаємодію з мобільним роботом.

7. Оцінка Положення Робота з Графічним Інтерфейсом:

Оцініть розташування мобільного робота на карті та в просторі, використовуючи графічний інтерфейс монітора комп'ютера.

Ця стратегічно побудована послідовність операцій є ключовою для успішного проведення експерименту та отримання інформації щодо роботи системи керування.

5.4 Аналіз результатів проведення експериментів і впровадження розробки

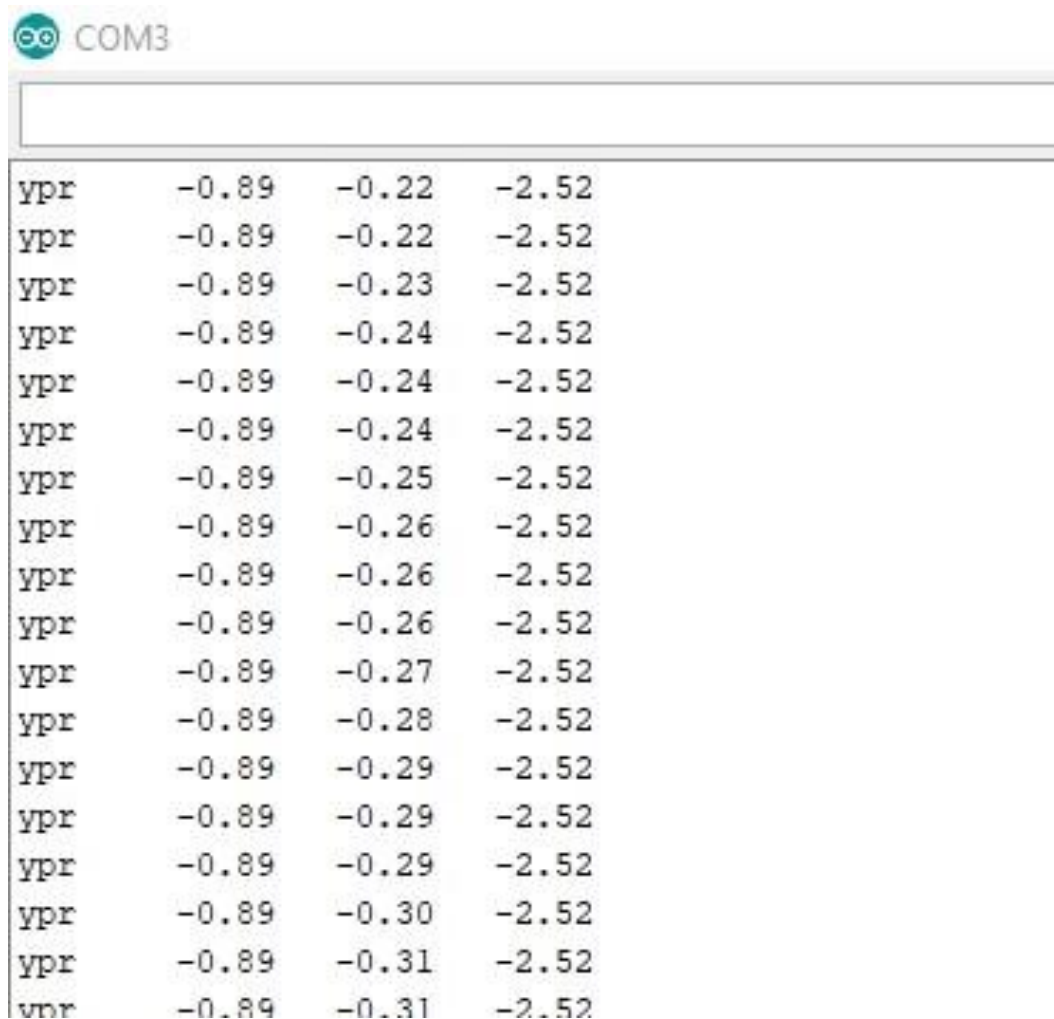
Правильна робота всієї системи відображається у графічному інтерфейсі комп'ютерної програми, де дані від периферійних датчиків використовуються

для візуальної оцінки. Для надійного відображення положення рухомого робота в місцевості та просторі необхідно провести оцінку можливих втрат даних при їх передачі від передавального контролера до приймаючого за допомогою сигналу радіозв'язку.

Цей процес вимагає уважного моніторингу ефективності передачі даних між контролерами та виявлення можливих артефактів чи втрат інформації. Забезпечуючи надійну та стабільну передачу сигналів радіозв'язку, система гарантує точність та достовірність відображених результатів у графічному інтерфейсі.

Оцінка на втрату даних допомагає виявляти можливі перешкоди чи збої в зв'язку, що можуть виникнути під час передачі інформації. Застосування надійних методів корекції та контролю можливих втрат даних забезпечує стабільну роботу системи та достовірність результатів, які відображаються у графічному інтерфейсі програми.

Монітор порту, що передає контролер, по радіо показано на рис. 5.3:



YPR			
YPR	-0.89	-0.22	-2.52
YPR	-0.89	-0.22	-2.52
YPR	-0.89	-0.23	-2.52
YPR	-0.89	-0.24	-2.52
YPR	-0.89	-0.24	-2.52
YPR	-0.89	-0.24	-2.52
YPR	-0.89	-0.25	-2.52
YPR	-0.89	-0.26	-2.52
YPR	-0.89	-0.26	-2.52
YPR	-0.89	-0.26	-2.52
YPR	-0.89	-0.27	-2.52
YPR	-0.89	-0.28	-2.52
YPR	-0.89	-0.29	-2.52
YPR	-0.89	-0.29	-2.52
YPR	-0.89	-0.29	-2.52
YPR	-0.89	-0.30	-2.52
YPR	-0.89	-0.31	-2.52
YPR	-0.89	-0.31	-2.52

Рисунок 5.3 – Монітор порту передавального контролера

Монітор порту приймаючого контролера представлений на рис. 5.4:

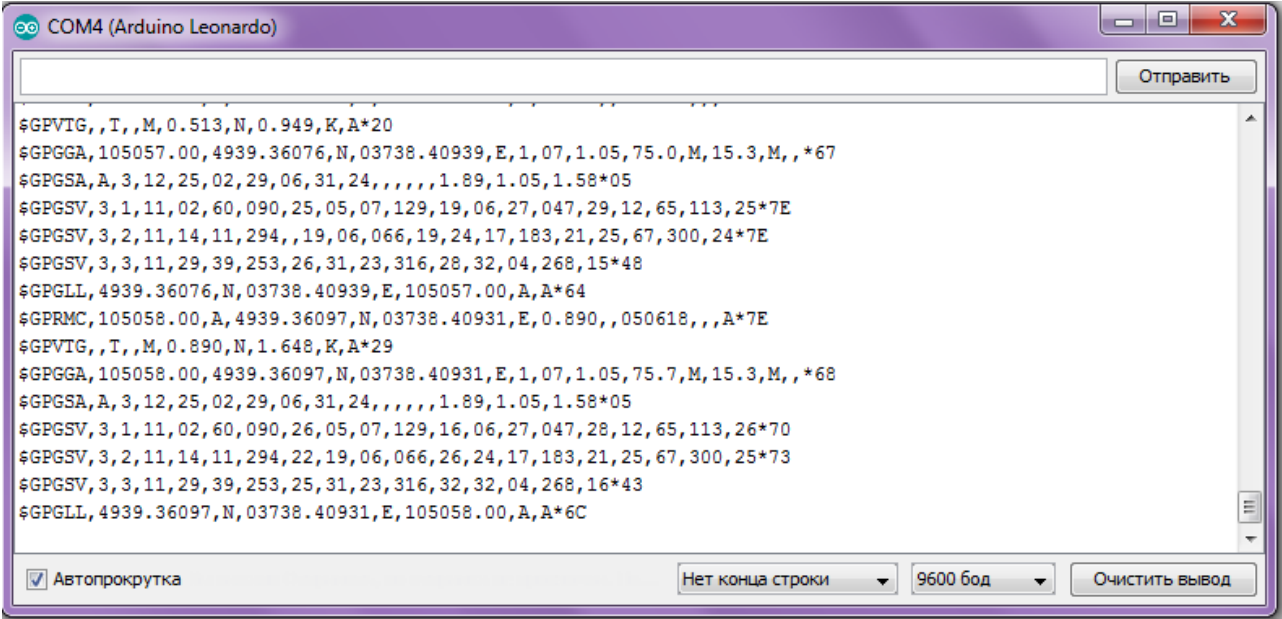


Рисунок 5.4 – Монитор порту приймаючого контролера

Вигляд виводу інформації на мобільний додаток з МСП представлений на рис. 5.5:

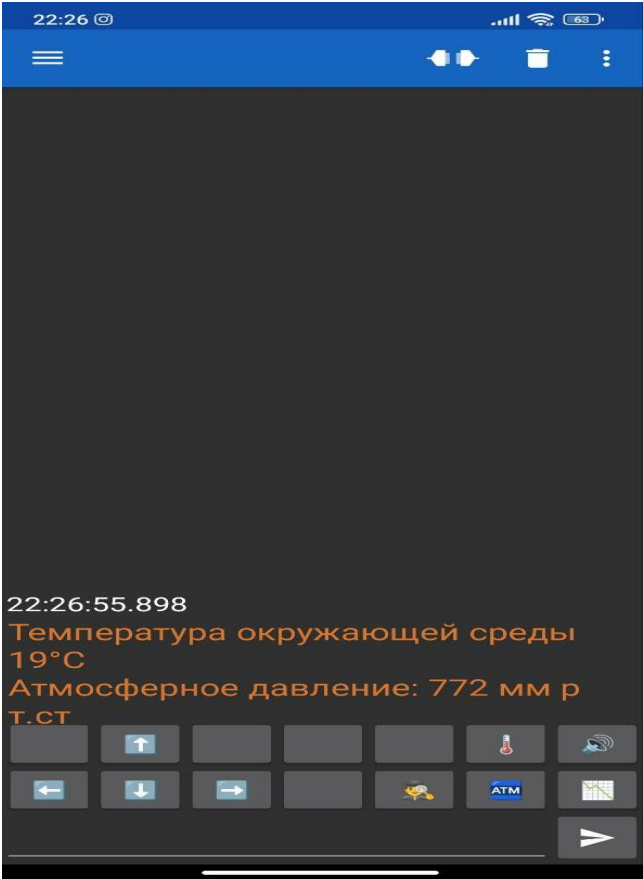


Рисунок 5.5 – Виведення інформації на мобільний додаток з МСП

5.5 Висновки за розділом

У ході виконання даного етапу проводились експериментальні та теоретичні дослідження системи автоматичного керування мобільним роботом. Після ретельної перевірки працездатності системи можна вивести висновок, що вона вірно відпрацьовує задані керуючі ефекти та передає достовірну інформацію для подальшого відображення в мобільному додатку. Отримані результати дозволяють зробити висновок, що розроблена система успішно виконує поставлені завдання.

Експериментальні дослідження надають можливість перевірити функціональність системи в реальних умовах експлуатації, враховуючи різноманітні сценарії роботи. Теоретичні дослідження в свою чергу дозволяють глибше розуміти принципи роботи системи та визначати її оптимальні параметри.

На підставі отриманих результатів можна засвідчити, що створена система відповідає поставленим завданням та ефективно функціонує у визначених умовах.

6 ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ РОЗРОБКИ

6.1 Опис об'єкта і практичних результатів дослідження

Метою даного розділу у магістерському проекті є створення економічної бази для розробки мобільних роботів. Для досягнення цієї цілі передбачено вирішення наступних завдань:

1. Аналіз ринку мобільних роботів:

- Оцінка та дослідження поточного стану ринку мобільних роботів, ідентифікація ключових тенденцій та конкурентоспроможності, а також визначення можливостей для подальшого розвитку.

2. Складність розрахункової роботи:

- Аналіз складності розрахункової роботи, враховуючи різноманіття факторів та величин, що впливають на ефективність проекту.

3. Оцінка розвитку мобільних роботів:

- Визначення перспектив та темпів розвитку мобільних роботів на основі аналізу існуючих технологічних та ринкових тенденцій.

4. Розрахунок заробітної плати, визначення вартості предметів і прогноз прибутку:

- Встановлення оптимального рівня заробітної плати, визначення собівартості предметів та прогнозування очікуваного прибутку від реалізації проекту.

5. Визначення точки беззбитковості:

- Встановлення рівня продажів, при якому витрати та доходи будуть однаковими, що дозволить визначити точку беззбитковості для проекту.

6.2 Аналіз ринку

Аналіз ринку - це систематичне дослідження та оцінка факторів, що впливають на споживачів та конкурентні умови в певній галузі чи сегменті економіки. Метою аналізу ринку є отримання глибокого розуміння ринкових умов, споживацьких потреб, конкурентних сил і можливостей для розробки та впровадження продукту чи послуги.

Основні етапи аналізу ринку включають збір та аналіз статистичних даних, вивчення тенденцій споживання, визначення конкурентів, оцінку потреб цільової аудиторії, аналіз можливостей ринкового зростання та визначення стратегій маркетингу. Результати аналізу ринку допомагають підприємствам у прийнятті обґрунтованих рішень щодо планування, розвитку та маркетингової стратегії своїх продуктів чи послуг.

Послідовність аналізу та розбиття запропонованого ринку можна подати наступним чином:

1. Визначення принципів та критеріїв сегментації продукту:

- Розробка основних параметрів для класифікації пропонованого продукту на ринку.

2. Розділення ринку на сегменти та визначення груп споживачів:

- Визначення груп споживачів продукту та поділ ринку на конкретні сегменти. Докладний аналіз кожного сегмента та розрахунок щорічної потужності для кожного сегмента та загального ринку.

3. Фіксація результатів:

- Збереження отриманих даних у вигляді структурованої форми для подальшого використання та аналізу.

4. Аналіз та вибір сегментів для докладнішого вивчення:

- Оцінка інформації про кожен сегмент та вибір тих, які потребують подальшого вивчення та розгляду.

5. Позичіонування продукту та визначення цільової виробничої потужності:

- Визначення стратегії позиціонування розробленого продукту в обраних сегментах ринку та визначення цільової виробничої потужності продукту.

Потенційними споживачами розробленої продукції для мобільних роботів можуть бути різноманітні підприємства як в Україні, так і за кордоном. Основним параметром сегмента є його місткість, що визначає кількість продуктів, які можна реалізувати протягом календарного року.

Розрахунок загальної місткості ринку можна розрахувати за такою формулою:

$$S_{\text{полн.1}} = N_1 * Q_1 * m_1,$$

де N_1 – кількість підприємств споживачів виробу в і-тому сегменті;

Q_i – середня річна програма виробів в і-тому сегменті, для яких буде поставлятися розглянутий товар;

m_i – кількість комплектуючих виробів, що йдуть в один виріб-споживач (1 шт.).

Сегментування і розрахунок місткості ринку представлені в таблиці 8.1.

Таблиця 6.1 – Сегментування і розрахунок повної ємності ринку

Сегменти ринку	Код сегменту	Кількість підприємств-споживачів	Середньо – річна програма виробів підприємств – споживачів	Повна ємність шт./рік
Підприємства споживачі України	А	3	7	21
Підприємства споживачі ближнього зарубіжжя	Б	4	6	24
Підприємства споживачі далекого зарубіжжя	В	1	5	5
Всього		8	18	50

З даних таблиці 6.1 видно, що загальна ємність ринку становить 50 одиниць / рік.

6.3 Розрахунки відповідно до завдання консультанта з економічного розділу

Витрати на виробництво включають різні складові, які враховують різні аспекти виробничого процесу. Серед цих складових входять:

1. Основні матеріали:
 - Вартість матеріалів, що використовуються в процесі виробництва.
2. Великі комплектуючі:
 - Витрати на значущі компоненти та вузли виробу.
3. Пряма та додаткова заробітна плата:
 - Витрати на оплату праці працівників, включаючи пряму та додаткову заробітну плату.
4. Витрати на технічне обслуговування та експлуатацію обладнання:**
 - Вартість утримання та регулярного технічного обслуговування виробничого обладнання.
5. Витрати на утримання транспорту:
 - Розходи, пов'язані з утриманням та експлуатацією транспортних засобів.
6. Державні податки та відрахування:
 - Вартість обов'язкових податків та соціальних відрахувань.

Для розробки інсталяції необхідно залучення різних спеціалістів, таких як розробники, колекціонери, тюнери і менеджери. Середній робочий місяць складає 22 дні. Композицію виконавців можна знайти в таблиці 6.2.

Таблиця 6.2 – Склад виконавців роботи

Посада	Посадові оклади, грн.	
	Місячні	Денні
Розробник	15000	681.81
Збирач	10000	454.55
Настроювач	12000	545.45
Керівник	20000	909.09
Всього	57000	2590.90

Таким чином, завдяки таблиці 6.4 видно, що загальний фонд заробітної плати за місяць складає 57000 грн. Щоденний посадовий оклад розробника – 2590.90 грн., збирача – 454.55 грн., настроювача – 545.45 грн., керівника – 909.09 грн.

Розрахунок трудомісткості робіт представлений в таблиці 6.3.

Таблиця 6.3 – Розрахунок трудомісткості робіт

Вид робіт	Тривалість, дні	Трудомісткість, люд./дні	Виконавці			
			Розробник	Збирач	Настроювач	Керівник
Технічне завдання (ТЗ)						
Постановка задачі	1	1	+	-	-	-
Розробка графіка ходу роботи	2	4	+	-	-	+
Визначення вимог до розробки плати	1	3	+	+	-	+
Розробка ТЗ	3	6	+	+	-	-
Погодження та затвердження ТЗ	3	9	+	+	-	+
Проектування друкованої плати						
Підготовчі роботи	1	2	-	-	+	+
Складальні роботи	3	6	-	+	-	+
Монтажні роботи	2	4	-	+	-	+
Налаштування	2	4	-	-	+	+
Приймально-здавальні	5	15	-	+	+	+

Впровадження						
Налагодження продукту	5	10	-	-	+	+
Випробування і здача продукту в експлуатацію	1	2	-	-	+	+
Всього	29	66	10	17	14	25

Отже, загальний робочий час становить 29 днів, а загальна складність – 66 осіб/добу. Трудомісткість – 10 днів для розробників, 17 днів для приймачів, 14 днів для тюнерів і 25 днів для менеджерів.

Розрахуємо базову заробітну плату розробника друкованих плат, враховуючи витрати на оплату праці, кількість керівників, середньоденну заробітну плату.

Для цього кількість днів роботи окремого виконавця множиться на його денну заробітну плату:

$$\text{ОЗП} = 681.81 \cdot 10 + 454.55 \cdot 17 + 545.45 \cdot 14 + 909.09 \cdot 25 = 44909,00 \text{ грн.}$$

У цьому випадку основна заробітна плата за місяць становить 44909,00 грн.

Додаткова заробітна плата розраховується за такою формулою:

$$\text{ДЗП} = 15\% \cdot \text{ОЗП} = 44909,00 \cdot 15\% = 6736,35 \text{ грн.}$$

Розрахувати вартість придбаної продукції та матеріалів, необхідних для виготовлення установки. Перелік придбаної продукції складається на основі переліку функціональних схем окремих плат.

Придбані вироби та матеріали наведені в таблиці 6.4. Ціни вказані в гривнях.

Таблиця 6.4 – Перелік покупних виробів і матеріалів

№	Елементи	Кількість	Ціна, грн.	Вартість
1	Мікроконтролер Arduino Uno R3	1	300,00	300,00
2	Драйвер двигунів L293D	1	58,00	58,00
3	Ультразвуковий далекомір HC-SR04	3	65,00	195,00
4	Плата MPU-6050	1	55,00	55,00
5	Bluetooth-модуль HC-05	1	161,00	161,00
6	Робо-платформа однопалубна (4 колеса)	1	166,00	166,00
7	Вимикач	1	15,00	15,00
8	З'єднувачі	20	0,80	41,6
9	Батарейки AA	4	6	24

Таким чином, вартість покупних виробів системи становить 1015,6 грн.

Внесок до єдиного соціального фонду становить 22% основної та додаткової заробітної плати:

$$Z_{\text{відр}} = \frac{\text{ОЗП} + \text{ДЗП}}{100} \cdot 22 = \frac{15454.25 + 2318.14}{100} \cdot 22 = 3909.93 \text{ грн. грн.}$$

Розрахувати річну норму амортизації за переліком обладнання, наведеним у таблиці 6.5.

Таблиця 6.5 – Вартість використаних обладнань

№	Обладнання	Кількість, шт.	Ціна, грн.
1	Стіл	4	1500
2	Стілець	4	1000
3	Лампа	4	300
4	Комп'ютер	4	30000
5	Паяльна станція	1	5000
6	Інструменти	1	500
7	Сума		38300

Річна норма амортизації (Нам) розраховується на рівні 25% вартості обладнання. Оскільки трудомісткість становить 29 днів, то амортизація розраховується так:

$$A_m = \frac{OC * 0,25 * 29}{12 * 22} = \frac{38300 * 0,25 * 21}{12 * 22} = 1051,8 \text{ грн.}$$

Орієнтовні розрахунки, витрати та ціни на продукцію, наведені в таблицях за статтями (табл. 6.6).

Таблиця 6.6 – Розрахунок собівартості і ціни виробу за статтями

№	Статті	Сума, грн.	Примітка
1	Основна заробітна плата (ОЗП)	44909,00	
2	Додаткова ЗП (ДЗП)	6736,35	20% від ОЗП
3	Єдиний соціальний фонд	11361,98	22%*(ОЗП+ДЗП)
4	Матеріали й куплені вироби (С _м)	1015,60	Табл. 8.4
5	Амортизація	1051,80	$A_m = \frac{OC * 0,25 * 29}{12 * 22}$
6	Витрати на утримання обладнання	3830,00	10% від табл.8.5
7	Додаткові витрати	17963,60	40% від ОЗП
8	Виробнича собівартість (С)	86868,33	п.1+п.2+п.3+...+п.7
9	Адміністративні витрати	20209,05	45% від ОЗП
10	Витрати на збут	21717,08	2,5% від п.8
11	Собівартість власних робіт	215662,79	п.1+п.2+п.3+...+п.10
12	Прибуток (П)	43132,56	20% від п.11
13	Ціна без ПДВ	258795,35	П+п.11
14	ПДВ	51759,07	20% від ціни без ПДВ
15	Ціна з ПДВ	310554,42	п.13+п.14

Отже, вартість робіт самого проекту з розробки лабораторного цеху становить 258795,35 грн, а ціна проекту (з ПДВ) 310554,42 грн.

Кількість замовлених примірників має бути не менше 45 примірників. Виробнича собівартість програмного продукту (БП) визначається за такою формулою:

$$BC_0 = BC / \text{кпп},$$

де BC – виробничі витрати;

$KПП$ – кількість замовлених екземплярів.

$$BC_0 = \frac{86868,33}{45} = 1930,41.$$

Загальна вартість одного примірника програмного продукту спільного підприємства включає суму виробничих витрат VSP , адміністративних витрат AB та маркетингових витрат VZ , які є копіями кожного програмного продукту:

$$СП_0 = BC_0 + AB_0 + Bz_0.$$

Адміністративні витрати AB_0 , що належать одному примірнику програмного продукту, визначаються за такою формулою:

$$AB_0 = \frac{AB}{KПП} = \frac{20209,05}{45} = 449,09.$$

Вартість реалізації Bz_0 припадає на один примірник програмного продукту і визначається за такою формулою:

$$Bz_0 = \frac{Bz}{KПП} = 482,6.$$

Таким чином:

$$СП_0 = 1930,41 + 449,09 + 482,6 = 2862,10 \text{ грн.}$$

Рентабельність продукції - це відношення загального прибутку до витрат виробництва та реалізації (відносна сума прибутку на 1 грн поточної собівартості):

$$P_{II} = \frac{Ц - BC}{BC} \cdot 100\% = \frac{310554,42 - 215662,79}{215662,79} \cdot 100\% = 44\%.$$

Тож рентабельність становить 44%. Обчислює вартість оптової ціни (без ПДВ) СРР продукту.

$$\text{ЦПП} = \text{СП} * \left(1 + \frac{P_p}{100}\right) = 2862,10 * \left(1 + \frac{44}{100}\right) = 4121,42 \text{ грн.}$$

$$\text{ДР} = \text{ЦПП} * \text{КПП} = 4121,42 * 45 = 185464,03 \text{ грн.}$$

де P_p — норма прибутку.

Обчисліть точку беззбитковості. Дохід від продажу програмних продуктів розраховується шляхом множення ціни одного програмного забезпечення на кількість замовлених копій програмного забезпечення:

Аналітичний розмір, критична програма (РКП) розраховується шляхом ділення фіксованої вартості R_{POSTV} на різницю між ціною одного програмного продукту CPP і змінною вартістю за копію програмного продукту ($З_{MB0}$), яка становить:

$$PKP = \frac{RP_{ост}B}{(ЦПП - З_{MB0})} = \frac{64771,53}{4121,42 - 1422,73} = 24шт.$$

Річні постійні витрати $RP_{ост}B$ складаються із суми наступних витрат:

$$\begin{aligned} RP_{ост}B &= BOY + A_M + ДВ + АВ + ВЗ \\ &= 3830 + 1051.8 + 17963,60 + 20209,05 + 21717,08 \\ &= 64771,53 \text{ грн} \end{aligned}$$

Річні зміни витрати $PЗ_M B$ складаються із суми наступних витрат:

$$\begin{aligned} PЗ_M B &= BM + ФОП + ЄСФОП = 1015,80 + (44909,00 + 6736,35) + \\ &11361,98 = 64022,93 \text{ грн. } PЗ_M B = BM + ФОП + ЄСФОП = 2275 + \\ &(15454.25 + 2318.14) + \text{Огрн.} \end{aligned}$$

Змінна вартість програмного продукту визначається шляхом ділення річних змінних витрат на річний план випуску продукту:

$$З_{MB0} = \frac{PЗ_M B}{45} = \frac{64022,93}{45} = 1422.73 \text{ грн.}$$

Річний дохід в точці беззбитковості:

$$Д_{БЗ} = 1422.73 * 24 = 34145,52 \text{ грн.}$$

Графічне зображення точки беззбитковості показано на рис. 6.1

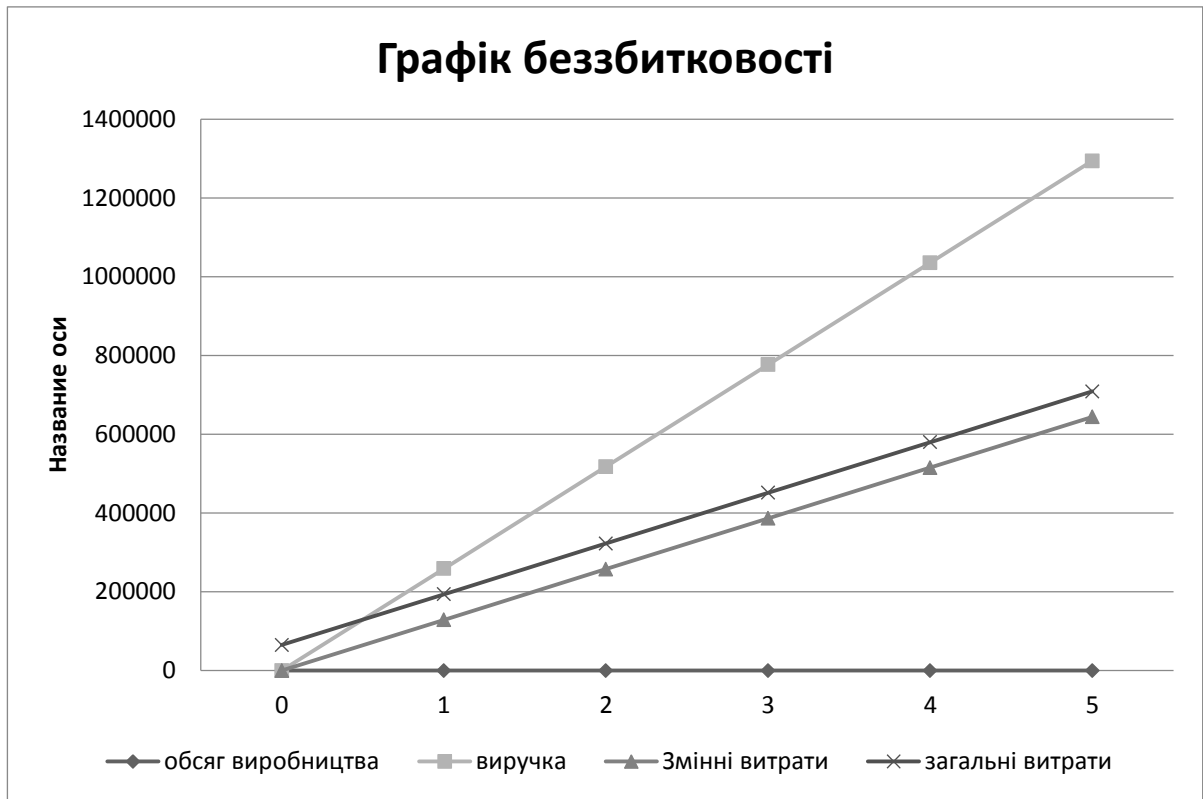


Рисунок 6.1 – Графік беззбитковості

Графік беззбитковості показує, що при 45 проданих одиницях виручка від реалізації перевищує загальну собівартість, тому на ділянці між ними ми отримуємо прибуток.

6.4 Висновки за розділом

У цьому розділі висвітлюється економічний аспект створення систем автоматичного керування мобільними роботами. Піддаються аналізу наступні етапи впровадження продукту: обсяг виробництва - 45 одиниць на рік, оцінка складності та розташування виконавців з загальною трудомісткістю 66 чол./добу, розрахунок заробітної плати з базовим окладом 6736,35 грн. Собівартість та ціна системи розраховуються окремо, без ПДВ - 258795,35 грн., включаючи ПДВ - 310554,42 грн. Для визначення точки беззбитковості розраховано річний дохід, який складає 34145,52 грн., при виробництві 45 штук на рік.

ВИСНОВКИ

1. Постановка задачі системи управління мобільною колісною платформою та дослідження режиму керованого заносу:

У цьому розділі було чітко визначено постановку завдання для розробки системи управління мобільною платформою. Проаналізовано та досліджено режим керованого заносу, що надало підґрунтя для подальшого проектування та експериментів.

2. Аналіз і синтез системи управління мобільною колісною платформою та дослідження режиму керованого заносу.

У даному розділі проведено глибокий аналіз та синтез системи управління, враховані різноманітні аспекти та вимоги. Результатом є високоякісна система, яка добре впоралася із завданням управління та демонструє стійкість у режимі керованого заносу.

3. Конструкторська частина:

В розділі конструкторської частини виконано детальний розгляд конструкційних аспектів розробки мобільної колісної платформи. Враховано ергономіку, надійність та функціональність, що сприяє створенню високоякісного продукту.

4. Дослідницька частина:

В даному розділі проведено дослідження різноманітних аспектів, пов'язаних із функціонуванням мобільної платформи. Отримані результати дозволили глибше зрозуміти та вдосконалити розроблену систему управління.

5. Експериментально-практична частина:

Розділ охоплює практичні випробування розробленої системи. Виконано експерименти з автоматичним управлінням, взаємодією з користувачем та відображенням інформації. Отримані результати свідчать про успішність системи у різних умовах.

6. Економічне обґрунтування розробки:

В розділі економічного обґрунтування викладено фінансові аспекти розробки, зокрема витрати, прибуток та точку беззбитковості. Детально розглянуто економічну сторону проекту, підкреслюючи його ефективність та рентабельність.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Перспективи застосування робототехнічних комплексів на користь забезпечення військової безпеки держави / Калач Г. П., Калач Г. Г., Травніков С. А. 2016.
2. Структурні властивості та класифікація кінематичних та динамічних моделей колісних мобільних роботів/ Г. Камп'юн, Ж. Бастен, Б. д'Андреа-Новель // Нелінійна динаміка, т. 7. №4 (Мобільні роботи), 2011.
3. Б.Б. Михайлов. Автономні мобільні роботи – Навігація та управління 2016.
4. Globally stable integrated kinematic and dynamic control for trajectory tracking of a mobile robot provided with caster wheel / Ayman A. El-Badawy, Amr Y. Mowafy // Journal of Al Azhar University Engineering Sector. Vol. 7, No. 24, July 2012, 1531-1544.
5. Момот М. В. Мобільні роботи на базі Arduino, 2017.
6. Звонарьов Д. А. Управління мобільним роботом з електричним двигуном. 2011.
7. А.С. Кулік, Конспект лекцій з теорії автоматичного керування. 89с.
8. Теорія автоматичного управління: навч. посібник / М.М. Савін, В.С. Єлсуков, О.М. П'ятина; під. ред. д.т.н., проф. В.І. Лачина. 2011.
9. Рачков, М. Ю. Технічні засоби автоматизації: підручник для академічного бакалаврату: для студентів вищих навчальних закладів, які навчаються з інженерно-технічних напрямків 2018.
10. Управління рухом мобільних колісних роботів/ Ю.Р. Мартиненко // Фундаментальна та прикладна математика, том 11, № 8, 2005. - С. 29 - 80.
11. Огляд алгоритмів локального позиціонування для мобільних пристроїв / Р. М. Мініахметов, О.О. Рогів, М. Л. Цимблер

ДОДАТОК А

Таблиця А.1 – Патентний аналіз

№	Номер патента, клас, країна, рік, автор	Назва патента	Опис винаходу	Постанова задачі	Результат	Недоліки
1	2	3	4	5	6	7
1	9866 B61F Україна 2015 Кулік Анатолій Степанович (UA); Дергачов Костянтин Юрійович (UA); Басова Анна Євгенівна (UA); Євтухов Богдан Миколайович (UA)	СИСТЕМА УПРАВЛІН НЯ КОЛІСНИМ ТРАНСПОР ТНИМ РОБОТОМ	Система управління колiсним транспортним роботом містить платформу, три колеса, два колiсні вали із змонтованими на них колесами, два електродвигуни, датчик кута повороту першого колеса, датчик швидкості обертання першого колеса,	В основу корисної моделі поставлено задачу створення засобів для планування траєкторії руху та оминання перешкод	запропонована система дозволяє планувати траєкторію руху та оминання перешкод.	

2	<p>US7319927B1</p> <p>Kelsey-Hayes Co</p> <p>Qiong Sun</p> <p>Eric Schieb</p> <p>Arnold H. Spieker</p> <p>David W. Webe</p> <p>Frank Lubischer</p>	<p>Constant speed control system</p>	<p>Винахід відноситься до систем стабілізації швидкості транспортних засобів за умов низькошвидкісної їх експлуатації з використання обертаючих моментів двигуна та гальм</p>	<p>Розраховується цільова швидкість транспортного засобу від положення педалі «газ». Визначається геометрія повороту транспортного засобу в залежності від положення руля. Розраховується цільова колісна швидкість для кожного з коліс як функція від цільової швидкості транспортного засобу та геометрії повороту. Вимірюються фактичні швидкості кожного з коліс. Визначаються вихідні обертаючий момент двигуна та гальм у відповідь на цільову швидкість коліс і фактичну швидкість коліс.</p>	<p>Технічним результатом є постійне підтримування заданої швидкості транспортним засобом.</p>	<p>Недолік зазначеної системи полягає в тому, що розроблена система призначена лише для експлуатації транспортного засобу лише на низьких швидкостях.</p>
---	--	--------------------------------------	---	--	---	---

3	<p>US3288232A</p> <p>SHEPHERD MACHINERY Co</p> <p>Willard W Shepherd</p>	<p>Wheel speed control</p>	<p>Винахід відноситься до регулювання швидкості обертання коліс і, зокрема, до системи управління для обмеження прослизання та перевищення швидкості колесами транспортного засобу (прокручування).</p>	<p>Вимірюються швидкості всіх коліс, ці сигнали подаються до електричної схеми, яка виробляє сигнали для управління гальмами для того, щоб припинити прослизання чи прокручування коліс. Електрична схема виконана повністю на аналогових елементах. Гальма є пневматичними.</p>	<p>Технічним результатом є значне зменшення прослизання та прокручування коліс транспортного засобу.</p>	<p>Складно конфігурувати через відсутність цифрового контролера в системі управління (відсутність цифрової техніки також є і перевагою через підвищення надійності системи).</p>
---	--	--------------------------------	---	--	--	--

ДОДАТОК Б

Лістинг коду роботи мобільного додатку та малого самохідного пристрою

Код роботи мобільного додатку:

```
import android.app.Activity;
import android.app.AlertDialog;
import android.bluetooth.BluetoothAdapter;
import android.bluetooth.BluetoothDevice;
import android.content.ComponentName;
import android.content.Context;
import android.content.Intent;
import android.content.ServiceConnection;
import android.os.Bundle;
import android.os.IBinder;
import android.text.Editable;
import android.text.Spannable;
import android.text.SpannableStringBuilder;
import android.text.method.ScrollingMovementMethod;
import android.text.style.ForegroundColorSpan;
import android.view.LayoutInflater;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.view.View;
import android.view.ViewGroup;
import android.widget.TextView;
import android.widget.Toast;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.fragment.app.Fragment;

public class TerminalFragment extends Fragment implements ServiceConnection,
SerialListener {

    private enum Connected { False, Pending, True }

    private String deviceAddress;
    private SerialService service;

    private TextView receiveText;
    private TextView sendText;
    private TextUtil.HexWatcher hexWatcher;

    private Connected connected = Connected.False;
    private boolean initialStart = true;
    private boolean hexEnabled = false;
    private boolean pendingNewline = false;
    private String newline = TextUtil.newline_crlf;

    /*
     * Lifecycle
     */
    @Override
    public void onCreate(@Nullable Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
    }
}
```

```

        setHasOptionsMenu(true);
        setRetainInstance(true);
        deviceAddress = getArguments().getString("device");
    }

    @Override
    public void onDestroy() {
        if (connected != Connected.False)
            disconnect();
        getActivity().stopService(new Intent(getActivity(),
SerialService.class));
        super.onDestroy();
    }

    @Override
    public void onStart() {
        super.onStart();
        if(service != null)
            service.attach(this);
        else
            getActivity().startService(new Intent(getActivity(),
SerialService.class)); // prevents service destroy on unbind from recreated activity
caused by orientation change
    }

    @Override
    public void onStop() {
        if(service != null && !getActivity().isChangingConfigurations())
            service.detach();
        super.onStop();
    }

    @SuppressWarnings("deprecation") // onAttach(context) was added with API 23.
    onAttach(activity) works for all API versions
    @Override
    public void onAttach(@NonNull Activity activity) {
        super.onAttach(activity);
        getActivity().bindService(new Intent(getActivity(),
SerialService.class), this, Context.BIND_AUTO_CREATE);
    }

    @Override
    public void onDetach() {
        try { getActivity().unbindService(this); } catch(Exception ignored) {}
        super.onDetach();
    }

    @Override
    public void onResume() {
        super.onResume();
        if(initialStart && service != null) {
            initialStart = false;
            getActivity().runOnUiThread(this::connect);
        }
    }

    @Override
    public void onServiceConnected(ComponentName name, IBinder binder) {
        service = ((SerialService.SerialBinder) binder).getService();
        service.attach(this);
        if(initialStart && isResumed()) {
            initialStart = false;
            getActivity().runOnUiThread(this::connect);
        }
    }

```



```

    }
}

@Override
public void onServiceDisconnected(ComponentName name) {
    service = null;
}

/*
 * UI
 */
@Override
public View onCreateView(@NonNull LayoutInflater inflater, ViewGroup
container, Bundle savedInstanceState) {
    View view = inflater.inflate(R.layout.fragment_terminal, container,
false);
    receiveText = view.findViewById(R.id.receive_text);
// TextView performance decreases with number of spans

receiveText.setTextColor(getResources().getColor(R.color.colorRecieveText)); // set
as default color to reduce number of spans
    receiveText.setMovementMethod(ScrollingMovementMethod.getInstance());

    sendText = view.findViewById(R.id.send_text);
    hexWatcher = new TextUtil.HexWatcher(sendText);
    hexWatcher.enable(hexEnabled);
    sendText.addTextChangedListener(hexWatcher);
    sendText.setHint(hexEnabled ? "HEX mode" : "");

    View sendBtn = view.findViewById(R.id.send_btn);
    sendBtn.setOnClickListener(v -> send(sendText.getText().toString()));
    return view;
}

@Override
public void onCreateOptionsMenu(@NonNull Menu menu, MenuInflater inflater) {
    inflater.inflate(R.menu.menu_terminal, menu);
    menu.findItem(R.id.hex).setChecked(hexEnabled);
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    int id = item.getItemId();
    if (id == R.id.clear) {
        receiveText.setText("");
        return true;
    } else if (id == R.id.newline) {
        String[] newlineNames =
getResources().getStringArray(R.array.newline_names);
        String[] newlineValues =
getResources().getStringArray(R.array.newline_values);
        int pos = java.util.Arrays.asList(newlineValues).indexOf(newline);
        AlertDialog.Builder builder = new
AlertDialog.Builder(getActivity());
        builder.setTitle("Newline");
        builder.setSingleChoiceItems(newlineNames, pos, (dialog, item1) -> {
            newline = newlineValues[item1];
            dialog.dismiss();
        });
        builder.create().show();
        return true;
    } else if (id == R.id.hex) {
        hexEnabled = !hexEnabled;
    }
}

```

```

        sendText.setText("");
        hexWatcher.enable(hexEnabled);
        sendText.setHint(hexEnabled ? "HEX mode" : "");
        item.setChecked(hexEnabled);
        return true;
    } else {
        return super.onOptionsItemSelected(item);
    }
}

/*
 * Serial + UI
 */
private void connect() {
    try {
        BluetoothAdapter bluetoothAdapter =
BluetoothAdapter.getDefaultAdapter();
        BluetoothDevice device =
bluetoothAdapter.getRemoteDevice(deviceAddress);
        status("connecting...");
        connected = Connected.Pending;
        SerialSocket socket = new
SerialSocket(getActivity().getApplicationContext(), device);
        service.connect(socket);
    } catch (Exception e) {
        onSerialConnectError(e);
    }
}

private void disconnect() {
    connected = Connected.False;
    service.disconnect();
}

private void send(String str) {
    if(connected != Connected.True) {
        Toast.makeText(getActivity(), "not connected",
Toast.LENGTH_SHORT).show();
        return;
    }
    try {
        String msg;
        byte[] data;
        if(hexEnabled) {
            StringBuilder sb = new StringBuilder();
            TextUtil.toHexString(sb, TextUtil.fromHexString(str));
            TextUtil.toHexString(sb, newline.getBytes());
            msg = sb.toString();
            data = TextUtil.fromHexString(msg);
        } else {
            msg = str;
            data = (str + newline).getBytes();
        }
        SpannableStringBuilder spn = new SpannableStringBuilder(msg + '\n');
        spn.setSpan(new
ForegroundColorSpan(getResources().getColor(R.color.colorSendText)), 0, spn.length(),
Spannable.SPAN_EXCLUSIVE_EXCLUSIVE);
        receiveText.append(spn);
        service.write(data);
    } catch (Exception e) {
        onSerialIoError(e);
    }
}
}

```

```

private void receive(byte[] data) {
    if(hexEnabled) {
        receiveText.append(TextUtil.toHexString(data) + '\n');
    } else {
        String msg = new String(data);
        if(newline.equals(TextUtil.newline_crlf) && msg.length() > 0) {
            // don't show CR as ^M if directly before LF
            msg = msg.replace(TextUtil.newline_crlf, TextUtil.newline_lf);
            // special handling if CR and LF come in separate fragments
            if (pendingNewline && msg.charAt(0) == '\n') {
                Editable edt = receiveText.getEditableText();
                if (edt != null && edt.length() > 1)
                    edt.replace(edt.length() - 2, edt.length(), "");
            }
            pendingNewline = msg.charAt(msg.length() - 1) == '\r';
        }
        receiveText.append(TextUtil.toCaretString(msg, newline.length() !=
0));
    }
}

private void status(String str) {
    SpannableStringBuilder spn = new SpannableStringBuilder(str + '\n');
    spn.setSpan(new
ForegroundColorSpan(getResources().getColor(R.color.colorStatusText)), 0,
spn.length(), Spannable.SPAN_EXCLUSIVE_EXCLUSIVE);
    receiveText.append(spn);
}

/*
 * SerialListener
 */
@Override
public void onSerialConnect() {
    status("connected");
    connected = Connected.True;
}

@Override
public void onSerialConnectError(Exception e) {
    status("connection failed: " + e.getMessage());
    disconnect();
}

@Override
public void onSerialRead(byte[] data) {
    receive(data);
}

@Override
public void onSerialIoError(Exception e) {
    status("connection lost: " + e.getMessage());
    disconnect();
}
}

```

Код роботи мобільного робота:

```

#include <AFMotor.h> //библиотека драйвера мотора
#include <NewPing.h> //библиотека датчика расстояния
#include <Servo.h> //библиотека сервы
#include "Parser.h"
#include <SFE_BMP180.h>
#include <Wire.h>

#include <DFPlayer_Mini_Mp3.h> //библиотека mp3 модуля
#include <SoftwareSerial.h> //библиотека сериалпорта

#define TRIG_PIN A0 // пин trig датчика расстояния
#define ECHO_PIN A1 // пин echo датчика расстояния
#define MAX_DISTANCE 100 // максимальное расстояние датчика расстояния
#define MAX_SPEED 150 // максимальная скорость двигателей
#define MAX_SPEED_OFFSET 20

#define CONST_TEMP 200
#define CONST_DISTANCE 400

int16_t ax, ay, az, gx, gy, gz;

#define BLUETOOTH_RX A3
#define BLUETOOTH_TX A2
#define CONTROLL_DELAY 1000

SoftwareSerial bluetooth(BLUETOOTH_RX, BLUETOOTH_TX);

NewPing sonar(TRIG_PIN, ECHO_PIN, MAX_DISTANCE); //создание объекта датчика
расстояния

AF_DCMotor motor1(1, MOTOR12_64KHZ); //подключение мотора 1
AF_DCMotor motor2(2, MOTOR12_64KHZ); //подключение мотора 2
AF_DCMotor motor3(3, MOTOR34_64KHZ); //подключение мотора 3
AF_DCMotor motor4(4, MOTOR34_64KHZ); //подключение мотора 4

#include "I2Cdev.h" //библиотека i2c - шини
#include "MPU6050.h" //библиотека акселерометра
#define TIME_OUT 20
MPU6050 accgyro; // создание объекта акселерометра
Servo myservo; // создание объекта серво

boolean goesForward = false;
int distance = 100;
int speedSet = 0; //скорость мотора по-умолчанию

int cm, t, p;
SFE_BMP180 bmp;
unsigned long lastTimeRead;
boolean autoPilot = false;

void setup() {
  accgyro.initialize(); //инициализация акселерометра

  bluetooth.begin(9600);
  Serial.begin(9600); //открытия порта для передачи данных
  mp3_set_serial(Serial); //создание канала связи между контролером и модулем
  mp3
  mp3_set_volume(100); //настройка громкости
  mp3_play(1); //воспроизведение первого трека

```

```

        if(bmp.begin()) // Инициализация датчика
        {
            Serial.println("BMP180 init success"); // Печать текста "BMP180
подключен"
        }
        else{ // В противном случае, датчик не
подключен
            Serial.println("BMP180 init fail\n\n"); // Печать текста "BMP180 не
подключен"
        }
        myservo.attach(10); //подключение сервы к 10 пину
        myservo.write(75); // установка начального угла
        readSensor();
    }
    void loop()
    {
        accgyro.getMotion6(&ax, &ay, &az, &gx, &gy, &gz); //считывание положения
акселерометра
        if (bluetooth.available()) {
            char buf[ 50];
            bluetooth.readBytesUntil(";", buf, 50);
            Parser data(buf, ',');
            int ints[ 10];
            data.parseInts(ints);

            switch (ints[0]) {
                case 0:
                    if(!autoPilot){
                        controllCar(ints[1]);
                        break;
                    }
                case 1:
                    sayDistance(distance);
                    break;
                case 2:
                    sayTemp(t);
                    break;
                case 3:
                    switchAuto(); moveStop();
                    break;
            }
        }
        // Serial.println(ax); //отправка данных в ком порт с значение оси x
        if (autoPilot) {
            if (distance <= 50) // проверка если дистанция меньше 50 сантиметров
            {
                takeLook(); //отработка функции предупреждение о дистанции
                reaction(); //отработка функции на предупреждение столкновения
            }
            if (ax > 6500 || ax < -6500) //проверка на столкновение
            {
                takeDamage();//отработка функции предупреждение о ударе
                reaction();//отработка функции на предупреждение столкновения
            }
            else { //если все нормально
                moveForward(); //отработка функции езды вперед
            }
        }
        readSensor();
    }
    void switchAuto() {
        autoPilot = !autoPilot;
        if(autoPilot){
            myservo.attach(10);
        }else{

```

```

        myservo.detach();
    }
}

void sayDistance(int distance){
    mp3_play (CONST_DISTANCE + distance);
    safeDelay()
}

void sayTemp(int temperature){
    mp3_play(CONST_TEMP + temperature)
}

void controllCar(int data){
    switch (data) {
        case 1:
            moveForward();
            safeDelay(CONTROLL_DELAY);
            moveStop(); break;
        case 2:
            moveBackward();
            safeDelay(CONTROLL_DELAY);
            moveStop();
            break;
        case 3:
            turnLeft();
            safeDelay(700);
            moveStop();
            break;
        case 4:
            turnRight();
            safeDelay(700);
            moveStop();
            break;
    }
}

void readSensor(){
    if(millis() - lastTimeRead > 100){
        cm = sonar.ping_cm();//считывание данных и преобразование в см
        if(bmp.startTemperature()){
            bmp.getTemperature(t);
        }
        if(bmp.startPressure(3)){
            double pTemp;
            bmp.getTemperature(pTemp, t);
            p = pTemp * 0.7500637554192
        }
        lastTimeRead = millis()
    }
}

void reaction() { //функция реакции на столкновение
    int distanceR = 0; //дистанция с право
    int distanceL = 0; //дистанция с лево
    safeDelay(40); //задержка
    moveStop(); // остановка
    safeDelay(200); //задержка
    distanceR = lookRight(); //считывание расстояния с правой стороны
    safeDelay(200); //задержка
    distanceL = lookLeft(); //считывание расстояния с левой стороны
    safeDelay(200); //задержка
    if (distanceR >= distanceL) //сравнение с какой стороны расстояние больше
    {
        turnRight(); // поворот в право
        moveStop(); // остановка
    } else

```

```

    {
        turnLeft();// поворот в лево
        moveStop();// остановка
    }
}
void control(int forward, int backward, int left, int right) {
    if (forward > 0) {
        moveForward();
        safeDelay(CONTROLL_DELAY);
        moveStop();
    }
    if (backward > 0) {
        moveBackward();
        safeDelay(CONTROLL_DELAY);
        moveStop();
    }
    if (left > 0) {
        turnLeft();
        moveStop();
    }
    if (right > 0) {
        turnRight();
        moveStop();
    }
}
void takeLook() { //функция предупреждения о расстоянии
    mp3_play (2); // воспроизведение 2 трека
    moveStop(); //остановка движения
    delay(100); //задержка
    moveBackward();//езда назад
    delay(200);//задержка
    moveStop();//остановка
    delay(200);//задержка
}
void takeDamage() { //функция предупреждения о ударе
    mp3_play (3); //воспроизведение 3 трека
    moveStop();//остановка движения
    delay(700);//задержка
    moveBackward();//езда назад
    delay(200);//задержка
    moveStop();//остановка
    delay(200);//задержка
    mp3_play (4); //воспроизведение 4 трека
}
boolean safeDelay(int pause) { //функция задержки без затормаживания работы
системы
    unsigned long timer = millis(); //получение времени с момента включения
контролера в миллисекундах
    while (millis() - timer <= pause) { //цикл
    }
}
int lookRight() // функция просмотра в право
{
    myservo.write(30);
    safeDelay(500);
    int distance = readPing();
    safeDelay(100);
    myservo.write(75);
    return distance;
}
int lookLeft()// функция просмотра в лево
{
    myservo.write(135);
}

```

```

    safeDelay(500);
    int distance = readPing();
    safeDelay(100);
    myservo.write(75);
    return distance;
    delay(100);
}
void moveStop() { //функция остановки
    motor1.run(RELEASE);
    motor2.run(RELEASE);
    motor3.run(RELEASE);
    motor4.run(RELEASE);
}
void moveForward() { // функция движения вперед

    if (!goesForward)
    {
        goesForward = true;
        motor1.run(FORWARD);
        motor2.run(FORWARD);
        motor3.run(FORWARD);
        motor4.run(FORWARD);
        for (speedSet = 0; speedSet < MAX_SPEED; speedSet += 2) // slowly bring the
speed up to avoid loading down the batteries too quickly
        {
            motor1.setSpeed(speedSet);
            motor2.setSpeed(speedSet);
            motor3.setSpeed(speedSet);
            motor4.setSpeed(speedSet);
            delay(5);
        }
    }
}
void moveBackward() { // функция движения назад
    goesForward = false;
    motor1.run(BACKWARD);
    motor2.run(BACKWARD);
    motor3.run(BACKWARD);
    motor4.run(BACKWARD);
    for (speedSet = 0; speedSet < MAX_SPEED; speedSet += 2) // slowly bring the
speed up to avoid loading down the batteries too quickly
    {
        motor1.setSpeed(speedSet);
        motor2.setSpeed(speedSet);
        motor3.setSpeed(speedSet);
        motor4.setSpeed(speedSet);
        delay(5);
    }
}
void turnRight() { // функция поворота в право
    motor1.run(BACKWARD);
    motor2.run(FORWARD);
    motor3.run(FORWARD);
    motor4.run(BACKWARD);
    delay(700);
}
void turnLeft() { // функция поворота в лево
    motor1.run(FORWARD);
    motor2.run(BACKWARD);
    motor3.run(BACKWARD);
    motor4.run(FORWARD);
    delay(700);
}
}

```


ДОДАТОК В

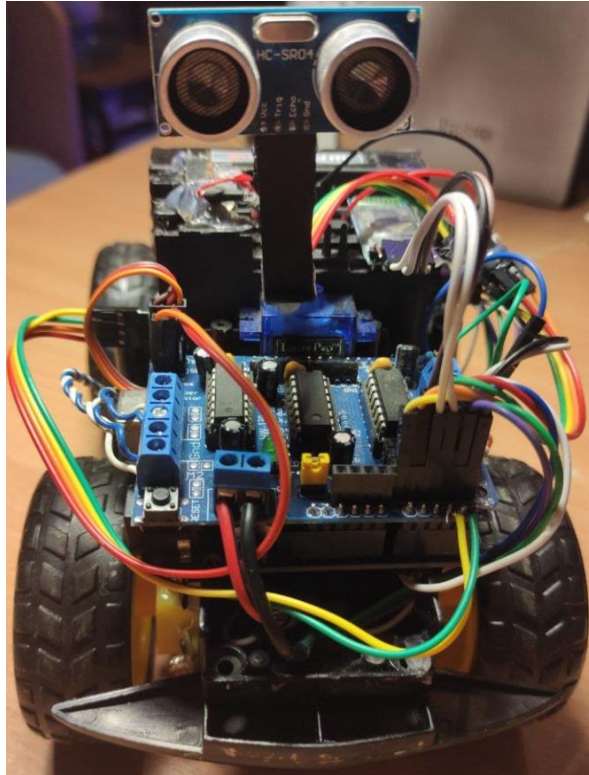


Рисунок В.1 – Самохідний пристрій вид зпереду

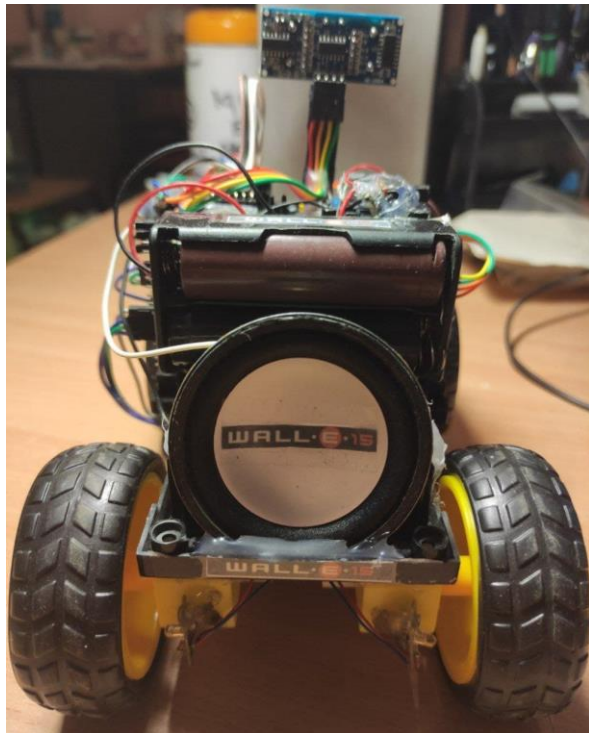


Рисунок В.2 – Самохідний пристрій вид ззаду

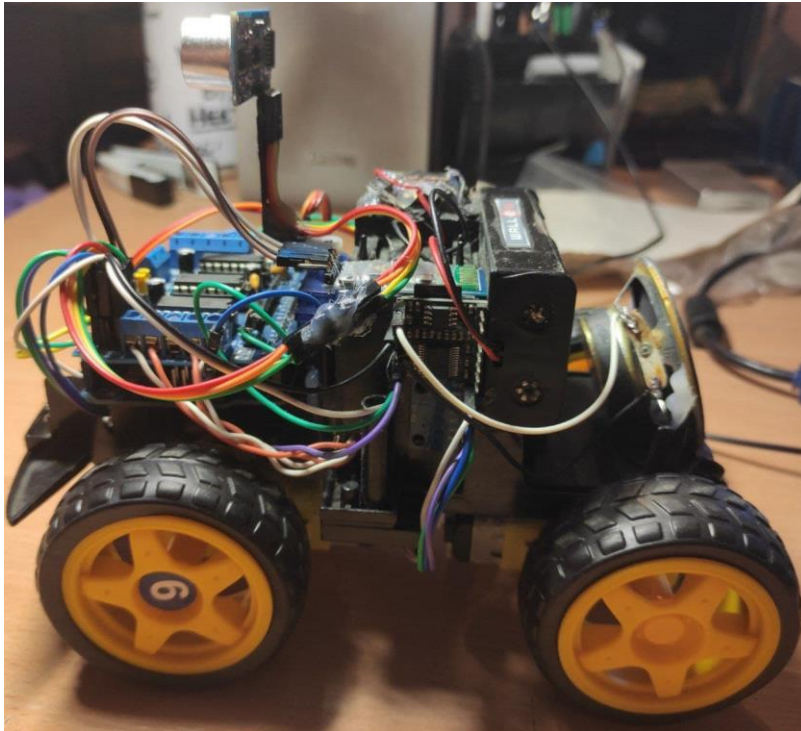


Рисунок В.3 – Самохідний пристрій вид зліва

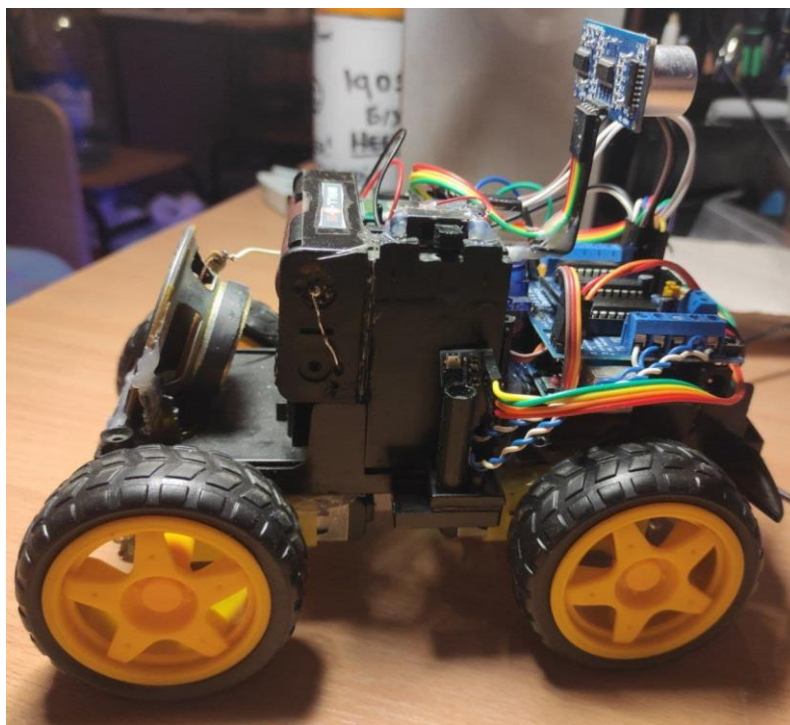


Рисунок В.4 – Самохідний пристрій вид справа

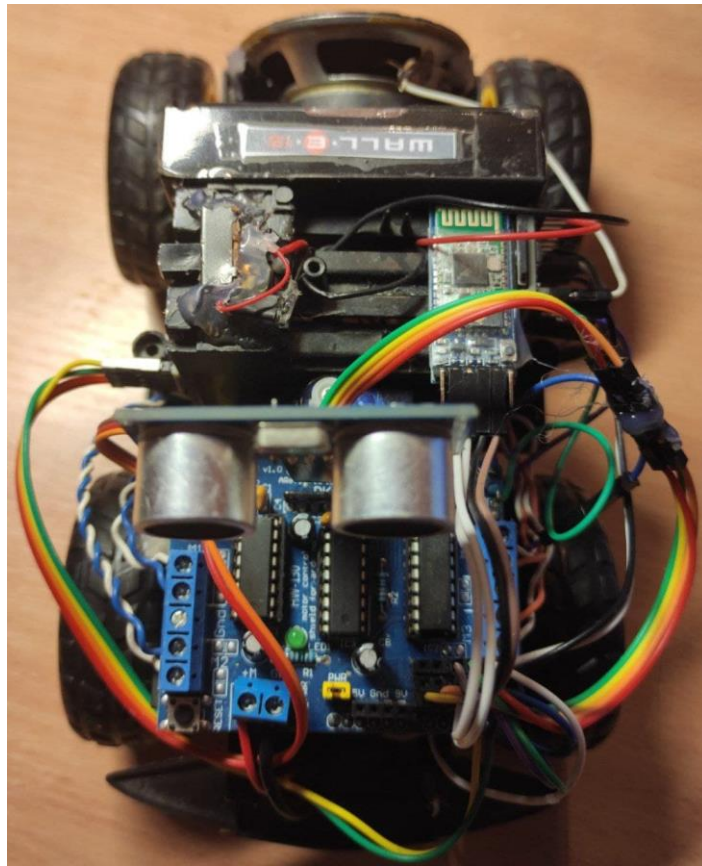


Рисунок В.5 – Самохідний пристрій вид зверху

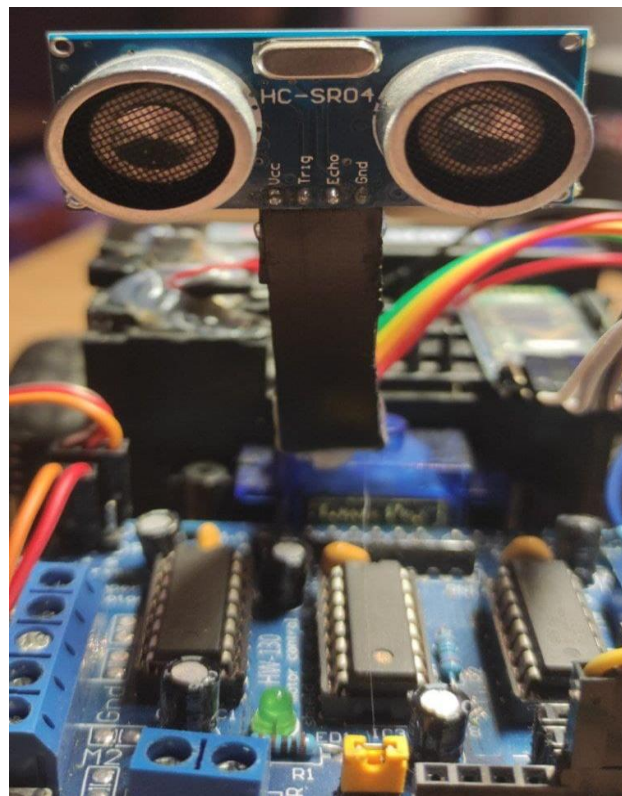


Рисунок В.6 – ультразвуковий датчик соннар

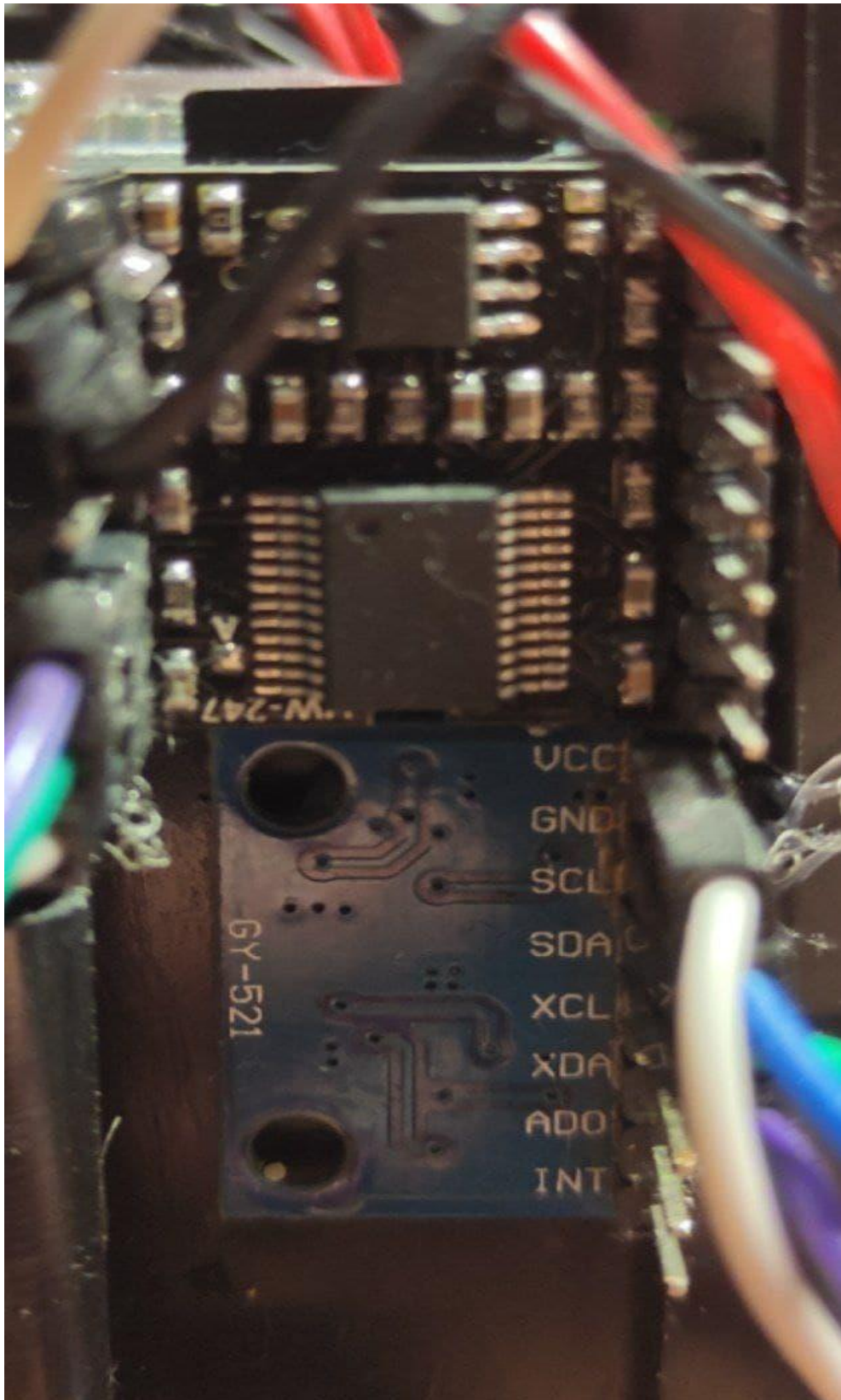


Рисунок В.7 – mp3 програвач зверху та датчик температури і тиску знизу



Рисунок В.8 – Датчик гіроскопу і акселерометру



Рисунок В.9 – Блютуз модуль для з'єднання з мобільним додатком

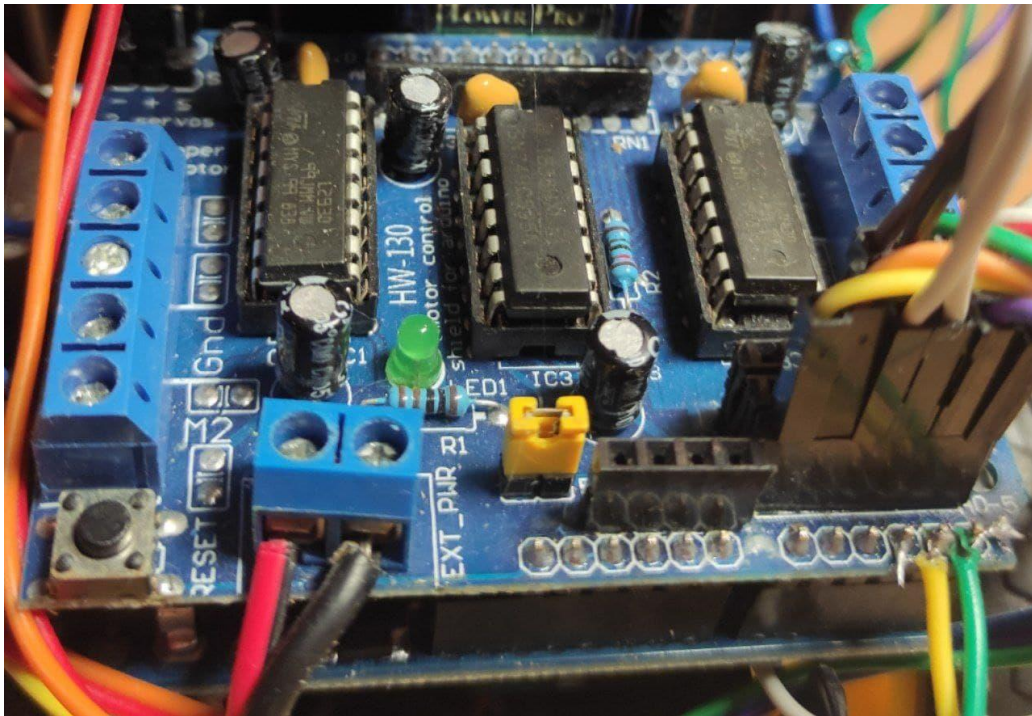


Рисунок В.10 – Драйвер моторів I293d який має вигляд розширювальної плати, яка підключається зверху на Arduino UNO



Рисунок В.11 – Акумуляторний хаб для підключення акумуляторів 18650

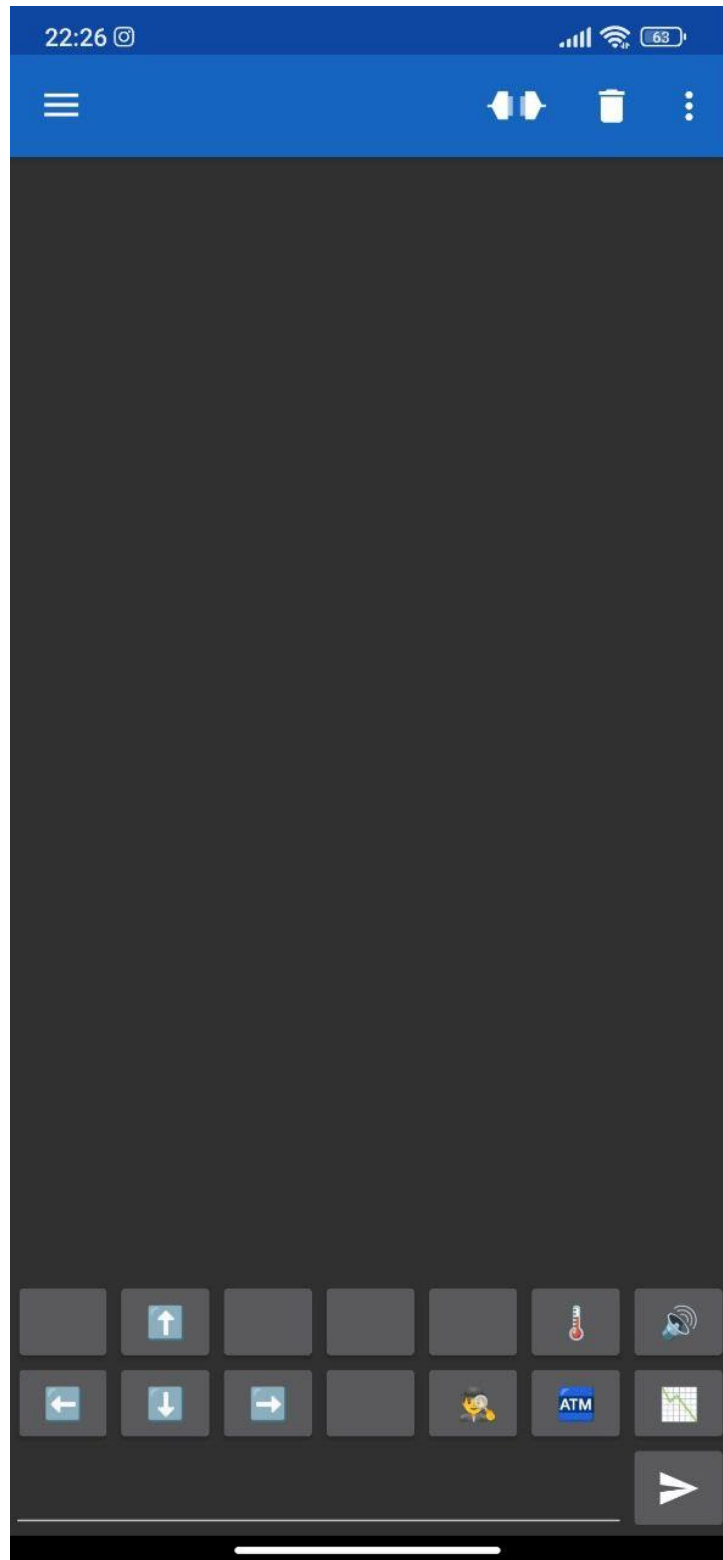


Рисунок В.12 – Інтерфейс мобільного додатку

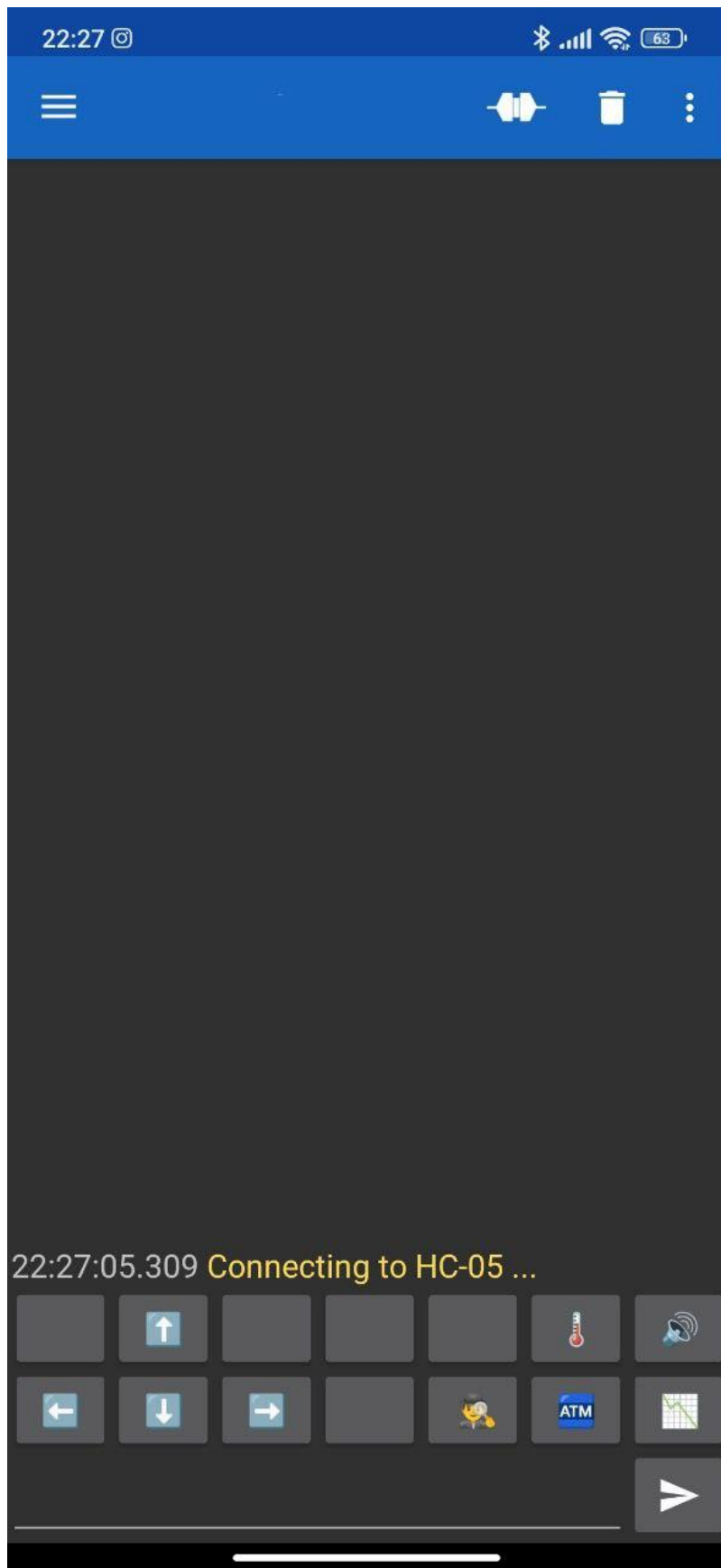


Рисунок В.13 – Підключення до Bluetooth модулю МСП

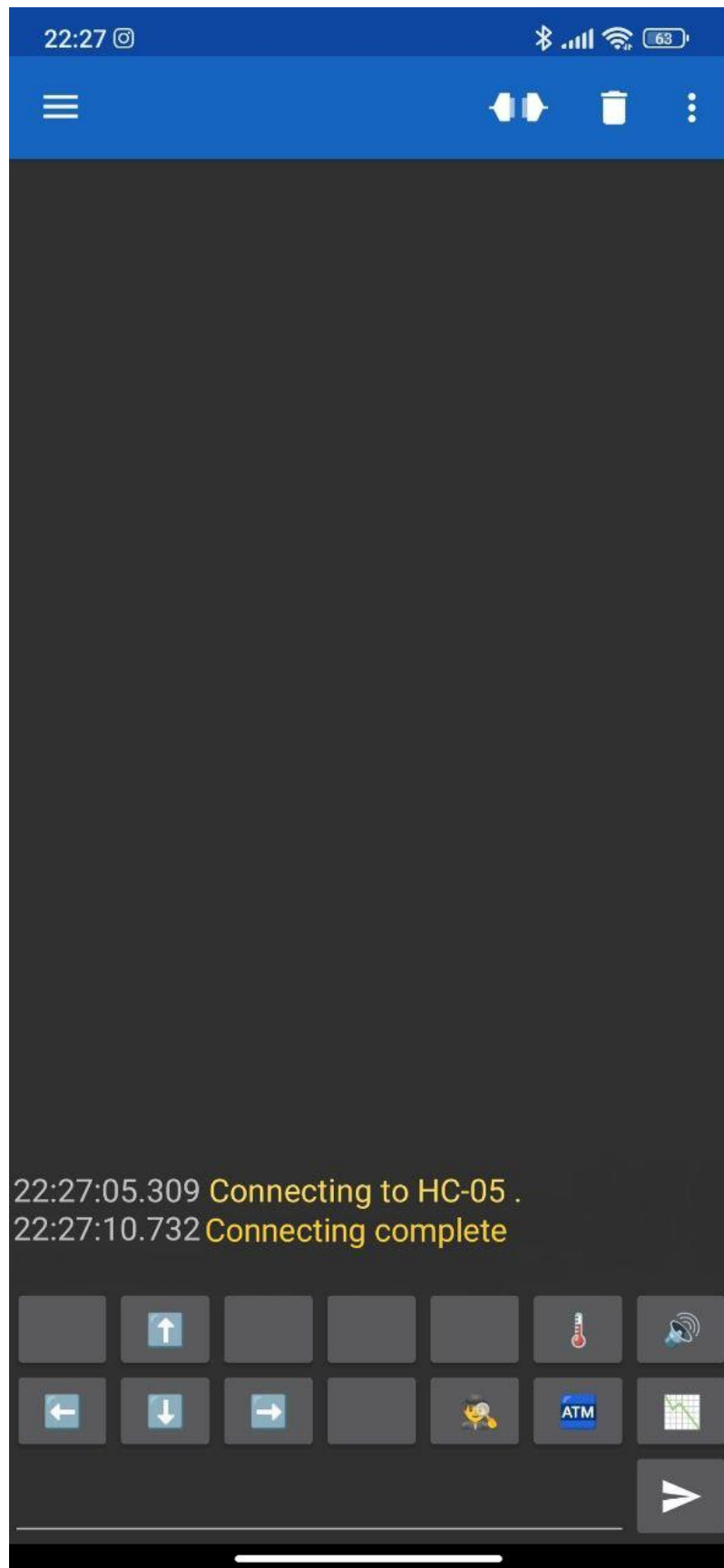


Рисунок В.14 – Успішне підключення до Bluetooth модулю МСП

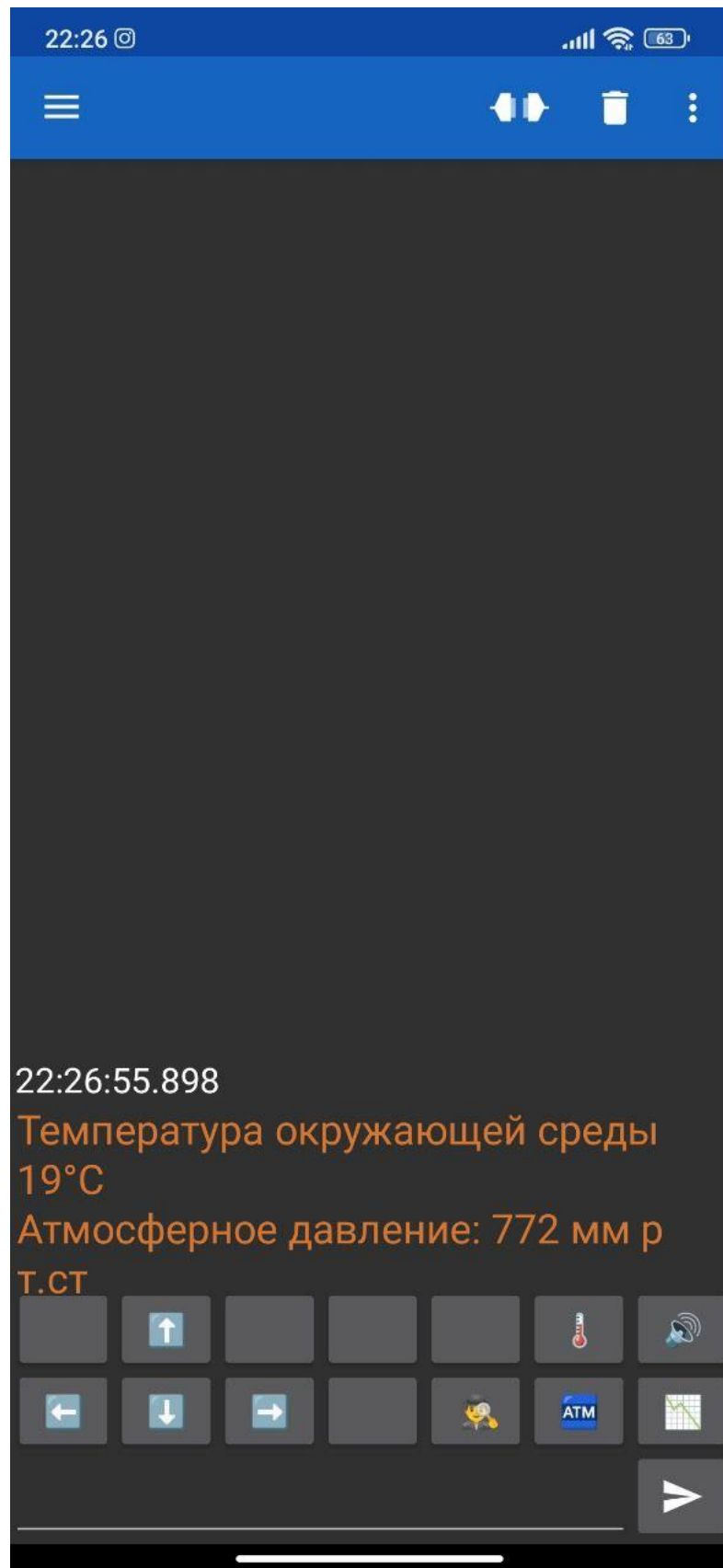


Рисунок В.15 – Выведення інформації на мобільний додаток з МСП