

Національний університет кораблебудування
імені адмірала Макарова

Навчально-науковий інститут автоматики та електротехніки

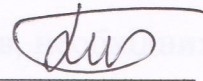
(повна назва інституту/факультету)

Кафедра автоматики

(повна назва кафедри)

«До захисту допущено»

Завідувач кафедри



(підпис)

I. С. Білюк

«20»

12

2024 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття ступеня магістра

зі спеціальності 141 Електроенергетика, електротехніка та електромеханіка

(код та назва спеціальності)

освітня програма Електромеханічні системи автоматизації та електропривод

(назва освітньої програми)

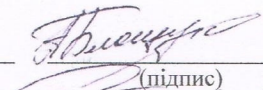
на тему: Система керування робота маніпулятора

на базі Arduino uno r3

Виконав: студент групи 6372м

Блащук Олександр Олександрович

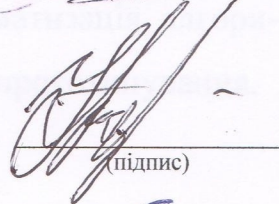
(прізвище, ім'я, по батькові)



(підпис)

Керівник к.т.н., асистент кафедри автоматики
Іванов А. В.

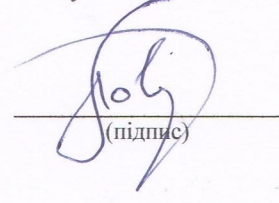
(посада, науковий ступінь, вчене звання, прізвище та ініціали)



(підпис)

Рецензент доцент кафедри СЕЕС,
к.т.н., доцент С. М. Новогрецький

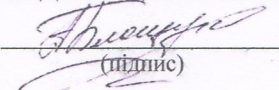
(посада, науковий ступінь, вчене звання, прізвище та ініціали)



(підпис)

Засвідчую, що у цій кваліфікаційній
роботі немає запозичень з праць інших
авторів без відповідних посилань.

Студент



(підпис)

м. Миколаїв – 2024 року

АНОТАЦІЯ

У даній магістерській роботі розглядається процес розробки системи керування роботом-маніпулятором на базі платформи Arduino Uno R3. Метою роботи є створення ефективного та надійного автоматизаційного механізму, здатного виконувати різноманітні завдання у галузі робототехніки. Робота починається з аналізу існуючих систем керування маніпуляторами та огляду сучасних підходів і технологій, що використовуються в цій сфері.

Далі надається детальний опис компонентів, необхідних для створення системи керування, включаючи апаратні (Arduino Uno R3, серводвигуни, датчики) та програмні засоби. Описано вибір робота-маніпулятора, його технічні характеристики та можливості взаємодії з Arduino. Велику увагу приділено методології розробки системи керування, включаючи визначення вимог до системи, створення блок-схеми процесів та розробку алгоритмів керування.

Основну частину роботи займає розробка алгоритмів керування, які включають алгоритми захоплення і переміщення об'єктів, точного позиціонування та використання зворотного зв'язку для корекції рухів у реальному часі. Результати роботи демонструють високу точність, надійність та ефективність розробленої системи керування роботом-маніпулятором.

Ключові слова: робот-маніпулятор, Arduino Uno R3, автоматизація, алгоритми керування, зворотний зв'язок, позиціонування, тестування, прототипування.

ABSTRACT

This master's thesis examines the development of a control system for a robot manipulator based on the Arduino Uno R3 platform. The goal of this work is to create an efficient and reliable automation mechanism capable of performing various tasks in the field of robotics. The study begins with an analysis of existing control systems for manipulators and a review of modern approaches and technologies used in this domain.

A detailed description of the components necessary for creating the control system, including hardware (Arduino Uno R3, servomotors, sensors) and software tools, is provided. The selection of the robot manipulator, its technical characteristics, and interaction capabilities with Arduino are discussed. Significant attention is paid to the methodology of developing the control system, including defining system requirements, creating process flowcharts, and developing control algorithms.

The main part of the thesis focuses on the development of control algorithms, which include object gripping and movement algorithms, precise positioning, and the use of feedback for real-time motion correction. The results of the study demonstrate the high accuracy, reliability, and efficiency of the developed control system for the robot manipulator.

Keywords: robot manipulator, Arduino Uno R3, automation, control algorithms, feedback, positioning, testing, prototyping.

ЗМІСТ

ВСТУП.....	6
РОЗДІЛ 1. ПРИЗНАЧЕННЯ ТА ОПИС ТЕХНОЛОГІЧНОГО ОБЛАДНАННЯ	7
1.1. Існуючі моделі роботів-маніпуляторів	7
1.1.1 Промислові роботи	7
1.1.2 Медичні роботи	10
1.1.3 Роботи для сільського господарства	12
1.1.4 Логістичні роботи	16
1.2. Основні підходи та технології систем керування.....	19
1.3. Використання Arduino в робототехніці	22
1.4. Програмне забезпечення для керування роботами-маніпуляторами	24
Висновки за розділом	26
РОЗДІЛ 2. МЕТОДОЛОГІЯ МАГІСТЕРСЬКОГО ДОСЛІДЖЕННЯ.....	27
2.1. Побудова структурно-логічної схеми магістерського дослідження.....	27
2.2. Структурно-логічна схема магістерського дослідження	33
2.3. Вибір методу вирішення поставленої задачі дослідження	34
РОЗДІЛ 3. ВИБІР ЕЛЕМЕНТІВ СХЕМИ СИСТЕМИ КЕРУВАННЯ	39
3.1. Опис Arduino Uno R3 та його можливостей.....	39
3.2. Опис маніпулятора.....	41
3.3. Підбір компонентів та системи електроприводу	43
3.4. Датчики для вимірювання позиції та зворотного зв'язку	45
3.5. Вибір додаткових компонентів.....	49
3.6. Опис процесу розробки та методології.....	54
Висновки за розділом	56

РОЗДІЛ 4. РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	58
4.1. Створення блок-схеми процесів	58
4.2. Розробка схеми системи керування на базі ARDUINO UNO3.....	64
4.3. Розробка програми системи керування роботом маніпулятором	71
Висновки за розділом	86
РОЗДІЛ 5. ОХОРОНА ПРАЦІ	88
5.1 Безпека використання роботів	88
5.2 Захист від дотику до об'єктів під напругою.....	90
5.3 Забезпечення безпеки виробничого обладнання.....	95
5.4 Пожежна безпека.....	95
5.5 Віброзахист.....	96
5.6 Виробничий пил як виробничий фактор шкідливості	98
5.7 Виробничий пил як виробничий фактор шкідливості	101
Висновки за розділом	102
ВИСНОВКИ.....	103
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	105
ДОДАТОК А МУЛЬТИМЕДІЙНА ПРЕЗЕНТАЦІЯ.....	107

ВСТУП

Сучасні технології автоматизації та роботизації активно розвиваються і застосовуються в різних галузях промисловості, медицини, сільського господарства тощо. Розробка системи керування робота-маніпулятора на базі Arduino Uno 3 є актуальною, оскільки дозволяє створювати недорогі, але ефективні рішення для автоматизації виробничих процесів і навчальних цілей. Використання Arduino Uno 3 надає можливість швидкого прототипування і тестування різноманітних підходів, що робить його ідеальним вибором як для професійних розробників, так і для студентів.

Мета і завдання роботи: Метою цієї роботи є розробка системи керування робота-маніпулятора на базі Arduino Uno 3, яка забезпечить точне і ефективне виконання робочих завдань.

Завданнями роботи: вивчення принципів роботи систем керування роботами-маніпуляторами; провести аналіз різних мікроконтролерів, датчиків, сервоприводів та інших компонентів, що використовуються в системах керування маніпуляторами, і вибрати оптимальні для реалізації на базі Arduino Uno R3; розробити алгоритми, які забезпечать точне і плавне виконання рухів маніпулятора, зокрема алгоритми для керування обертами сервоприводів, корекції траєкторії та управління швидкістю рухів; тестування та оцінка ефективності системи.

Об'єкт дослідження: об'єктом даного дослідження є система керування роботом-маніпулятором, що базується на платформі Arduino Uno R3.

Предмет дослідження: предметом дослідження є розробка та реалізація програмного забезпечення та апаратної частини системи керування маніпулятором на базі мікроконтролера Arduino Uno R3, зокрема, методи керування рухами маніпулятора, використання датчиків та сервоприводів.

РОЗДІЛ 1

ПРИЗНАЧЕННЯ ТА ОПИС ТЕХНОЛОГІЧНОГО ОБЛАДНАННЯ

1.1 Існуючі моделі роботів-маніпуляторів

Робот-маніпулятор – це автоматизований пристрій, що імітує рухи руки людини для виконання широкого спектру завдань, які потребують високої точності, швидкості та повторюваності. Основними компонентами такого робота є ланки (механічні сегменти), шарніри (з'єднання, що забезпечують рух), сенсори (для отримання інформації про навколишнє середовище) та приводи (елементи, що генерують рух).

Роботи-маніпулятори мають від 3 до 7 ступенів свободи, які дозволяють виконувати різноманітні завдання у просторі, включаючи переміщення об'єктів, складання деталей або проведення операцій.

Метою використання роботів-маніпуляторів є автоматизація процесів, які потребують високої точності, швидкості, повторюваності чи здатності працювати в умовах, небезпечних або недоступних для людини. Їх застосування значно знижує витрати часу та ресурсів, підвищуючи ефективність і якість виконуваних завдань.

1.1.1 Промислові роботи

Промислові роботи широко використовуються у виробничих процесах для автоматизації задач, що вимагають високої точності, надійності, повторюваності, швидкості та можливості працювати у складних умовах. Основними задачами є зварювання, фарбування, обробка матеріалів та сортування продукції.

Одними з найбільш відомих моделей є роботи компаній KUKA, ABB, FANUC і Yaskawa. Серія KUKA KR є однією з найпопулярніших серед промислових роботів. Вона охоплює широкий спектр моделей, які можуть виконувати різноманітні завдання від легких до важких умов роботи. Роботи ABB IRB відомі своєю універсальністю, енергоефективністю та високою точністю виконання завдань. Ця серія використовується в багатьох галузях, від складання до пакування. Роботи FANUC спеціалізуються на точних і швидких операціях. Моделі серії M-20iA вирізняються високою маневреністю і міцністю. Японська компанія Yaskawa спеціалізується на розробці роботів для зварювання, обробки матеріалів і палетування[1].

Ці системи можуть включати такі елементи:

Програмовані логічні контролери (ПЛК): Вони використовуються для автоматизації виробничих ліній та керування різними пристроями, включаючи роботів-маніпуляторів. ПЛК забезпечують високу швидкість обробки даних та можуть керувати складними операціями.

Промислові роботи з числовим програмним керуванням (ЧПК): Вони використовуються в металообробці, деревообробці та інших виробничих процесах. ЧПК дозволяють точно керувати рухом маніпуляторів, забезпечуючи високу точність виготовлення деталей.

Роботи для автоматизації складання

Ці роботи використовуються на виробничих лініях для складання компонентів та деталей. Роботи для автоматизації складання є важливим елементом сучасного виробництва. Вони призначені для виконання завдань зі збирання деталей і компонентів з високою точністю, швидкістю та повторюваністю. Використання таких роботів дозволяє зменшити виробничі витрати, підвищити якість продукції та знизити вплив людського фактора. Вони часто використовують технології машинного зору для точного розпізнавання об'єктів і їх розташування.

Основні елементи включають:

Системи машинного зору: Ці системи використовують камери та алгоритми обробки зображень для розпізнавання об'єктів і визначення їх положення у просторі. Це дозволяє роботам точно захоплювати і маніпулювати об'єктами.

Системи зі зворотним зв'язком: Вони використовують датчики для вимірювання положення, швидкості та сили, які використовуються маніпулятором. Це дозволяє коригувати рухи робота в режимі реального часу для досягнення високої точності.

Гнучкі захвати: Спеціальні інструменти для роботи з різними типами деталей: вакуумні присоски, магнітні захвати, механічні пальці. Такі захвати адаптуються під форму та матеріал компонентів, що підвищує ефективність.

Контроль моменту сили: Роботи оснащуються датчиками, які контролюють силу затиску під час складання, що зменшує ризик пошкодження крихких деталей.

Програмне забезпечення для симуляції: Робочі завдання робітв моделюються у віртуальному середовищі перед впровадженням на виробництві, що дозволяє оптимізувати процеси.

Прикладом промислових робітв для складання є:

FANUC LR Mate 200iD – це компактний 6-осьовий робот-маніпулятор, який оптимально підходить для завдань автоматизації складання (рис.1.1). До сфери застосування можна віднести: складання електронних компонентів; збирання дрібних механічних деталей; маніпуляції з крихкими елементами, що вимагають обережності. Особливості даного робота це: інтеграція з системою машинного зору FANUC iRVision для точного визначення положення деталей; компактність дозволяє встановлення навіть на невеликих виробничих ділянках; висока енергоефективність і низькі витрати на обслуговування.



Рисунок 1.1 Промисловий робот FANUC: lr mate 200id 7h

KUKA KR AGILUS – швидкий, компактний робот для складання, який відзначається високою точністю та гнучкістю (рис.1.2). Застосовується для монтажу компонентів у автомобільній промисловості та роботі з пластиковими, металевими або електронними деталями. Відрізняється наступними особливостями: інтеграція із системами автоматизації через програмне забезпечення KUKA.WorkVisual; висока продуктивність завдяки швидкій динаміці руху; можливість роботи в складних умовах (запиленість, висока вологість).



Рисунок 1.2 Робот KUKA KR 3 AGILUS

1.1.2 Медичні роботи

Медичні роботи-маніпулятори розроблені для виконання завдань, які потребують максимальної точності, стабільності та чіткого виконання інструкцій. Вони стали незамінними в хірургії, діагностиці, реабілітації та інших медичних процедурах. Їх застосування дозволяє зменшити ризик для пацієнтів, підвищити якість лікування та знизити фізичне навантаження на медичних працівників. Вони забезпечують високий рівень точності та мінімізацію інвазивності.

Одними з представників даного типу роботів-маніпуляторів є:

Da Vinci – це роботизована хірургічна система, створена компанією Intuitive Surgical. Вона є одним із найпоширеніших і найрозвинутіших роботів для мінімально інвазивних операцій (рис.1.3). Система дозволяє хірургу керувати роботизованими інструментами через консоль, забезпечуючи високу точність і маневровість[2]. Сфера застосування: хірургія простати – радикальна простатектомія є однією з найчастіших операцій, виконуваних за допомогою Da Vinci. Робот дозволяє зберігати нервові закінчення, що зменшує ризик побічних ефектів; кардіохірургія – використовується для реконструкції клапанів серця, що вимагає високої точності та мінімізації травм; гінекологія – лапароскопічні операції на матці, включаючи видалення пухлин; гастроентерологія – операції на шлунково-кишковому тракті, такі як резекція пухлин шлунка або товстої кишки.



Рисунок 1.3 Робот-хірург da Vinci

Університетська клініка Мейо (США) активно використовує Da Vinci для виконання операцій у сфері урології. Завдяки використанню роботів час перебування пацієнтів у лікарні зменшився на 30%, а ускладнення скоротилися на 25%.

ROSA – це робот для нейрохірургії, який допомагає виконувати складні операції на мозку. Він використовується для планування і виконання стереотаксичних процедур, таких як встановлення електродів для лікування епілепсії чи хвороби Паркінсона (рис.1.4).

Основні елементи включають:

Хірургічні роботи: Вони дозволяють хірургам виконувати операції з мінімальними розрізами, що скорочує час відновлення пацієнтів. Вони часто використовують систему керування з високою точністю для виконання складних маніпуляцій.

Роботи для реабілітації: Вони використовуються для допомоги пацієнтам у відновленні після травм чи операцій. Ці роботи можуть включати екзоскелети, які

допомагають пацієнтам рухатися, або роботизовані системи, що надають підтримку під час фізіотерапії.



Рисунок 1.4 Операційний робот напрямок ROSA ONE

1.1.3 Роботи для сільського господарства

У сільському господарстві роботи використовуються для автоматизації задач, таких як збирання врожаю, обробка ґрунту та догляд за рослинами. Вони сприяють автоматизації трудомістких процесів, підвищенню врожайності та зниженню витрат на людську працю.

Приклад використання: робот Tertill для догляду за рослинами

Tertill – це робот-маніпулятор, створений для догляду за рослинами, зокрема для знищення бур'янів (рис.1.5). Він розроблений із врахуванням потреб дрібних фермерських господарств і теплиць. Перевагами даного робота є: автономність – робот працює на сонячних батареях, що забезпечує безперервну роботу протягом

дня без підзарядки; оснащений сенсорами для навігації серед грядок і уникнення перешкод. Функціональність – робот розпізнає бур'яни за розміром рослин і видаляє їх механічними інструментами; підходить для роботи на грядках із низькорослими культурами, такими як салат, полуниця або зелень. Можливість роботи у відкритому ґрунті – захищений корпус дозволяє працювати в умовах підвищеної вологості або запиленості.



Рисунок 1.5 Робот маніпулятор Tertill

Приклад використання: робот-маніпулятор для збирання плодів – Agrobot E-Series – робот-маніпулятор, розроблений для автоматизованого збирання ягід і фруктів, наприклад, полуниці (рис.1.6).

Основні характеристики:

Високоточна система збору: робот оснащений маніпулятором з м'якими захватами, що дозволяють обережно збирати плоди, не пошкоджуючи їх.

Машинне бачення: камери та сенсори аналізують зрілість плодів за кольором, розміром і формою, що дозволяє збирати тільки стиглі фрукти.

Мобільність: робот переміщається по полю завдяки гусеничному або колісному шасі.

Продуктивність: може збирати до 2-3 тонн полуниці за день, що перевищує продуктивність людських робітників.

Автономність: може працювати безперервно, використовуючи акумулятори або генератори.



Рисунок 1.6 AGROBOT E-Series Robotic Strawberry Harvesters

Роботи для посіву та обробки ґрунту – робот FarmBot

FarmBot – це автоматизований модульний робот, призначений для точного висіву насіння, догляду за рослинами та поливу (рис.1.7).

У фермерському господарстві в Нідерландах робот Agrobot E-Series був використаний для збирання полуниці. Завдяки автоматизації продуктивність ферми зросла на 40%, а витрати на людську працю зменшилися на 30%. Крім того, якість ягід значно покращилася завдяки дбайливому збиранню плодів.

Основні характеристики:

Модульна конструкція: легко налаштовується під різні розміри грядок.

Точний посів: система використовує координати для висіву насіння з точністю до міліметра, що знижує втрати насіння.

Моніторинг рослин: оснащений камерами для аналізу стану ґрунту, вологості та росту культур.

Автоматизація догляду: вбудовані інструменти дозволяють видаляти бур'яни, розпушувати ґрунт і забезпечувати полив у визначених зонах.

Інтеграція з ПЗ: підключається до смартфона або комп'ютера, дозволяючи фермеру програмувати завдання.

Основні елементи включають:

Автономні трактори: Вони можуть самостійно виконувати задачі на полях, такі як оранка, посів і збирання врожаю. Вони часто використовують GPS-навігацію для точного визначення свого положення на полі.

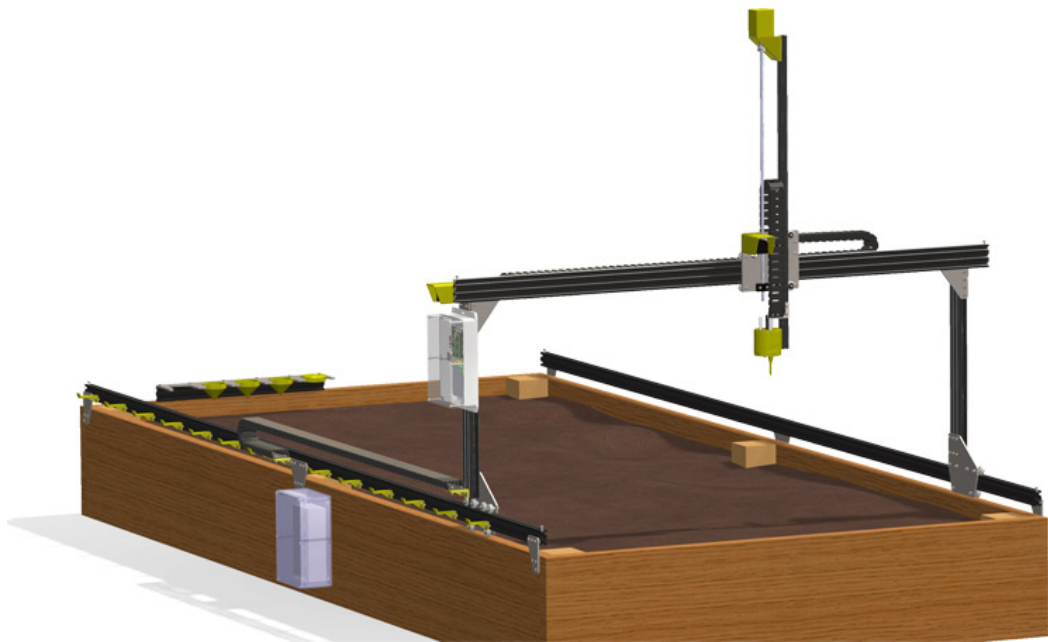


Рисунок 1.7 Farmbot: автономний робот

Роботи для збирання врожаю: Вони використовують технології машинного зору та маніпулятори для автоматичного збирання фруктів, овочів та інших культур. Це дозволяє підвищити ефективність збирання врожаю і зменшити втрати.

Машинне бачення: використовується для розпізнавання рослин, бур'янів і оцінки стану культур.

Датчики вологості та температури: дозволяють оцінювати стан ґрунту і приймати рішення про полив.

Розумні алгоритми: забезпечують оптимізацію роботи, визначають оптимальні маршрути руху та порядок виконання завдань.

Механічні захвати: спеціально розроблені для роботи з рослинами, які вимагають делікатного поводження.

1.1.4 Логістичні роботи

Логістичні роботи використовуються для автоматизації складських і транспортних операцій. Вони дозволяють підвищити ефективність управління запасами, зменшити витрати на ручну працю, забезпечити швидкість і точність у процесах переміщення, сортування та складування вантажів.

Universal Robots (UR) – це серія універсальних роботів-маніпуляторів (так звані колаборативні роботи або коботи), які можуть бути інтегровані в різні логістичні та виробничі процеси. UR-роботи розроблені компанією Universal Robots із Данії. Вони призначені для автоматизації рутинних завдань без необхідності кардинальної зміни інфраструктури (рис.1.8).

Основні особливості

Універсальність: роботи підходять для виконання широкого спектра завдань, включаючи сортування, пакування, переміщення вантажів і складування; можуть бути адаптовані до різних умов завдяки програмуванню.

Легка інтеграція: UR-роботи легко інтегруються у вже існуючі складські системи, включаючи конвеєрні лінії та автоматизовані склади; інтуїтивний

інтерфейс програмування дозволяє швидко налаштувати робота під конкретні завдання без необхідності залучення програмістів.

Колаборативність: безпечна робота поруч із людьми завдяки сенсорам для запобігання зіткненням; можливість функціонування у змішаних командах (людина + робот).

Гнучкість: оснащені різними типами захватів для маніпуляції з різними об'єктами: коробками, палетами, контейнерами.

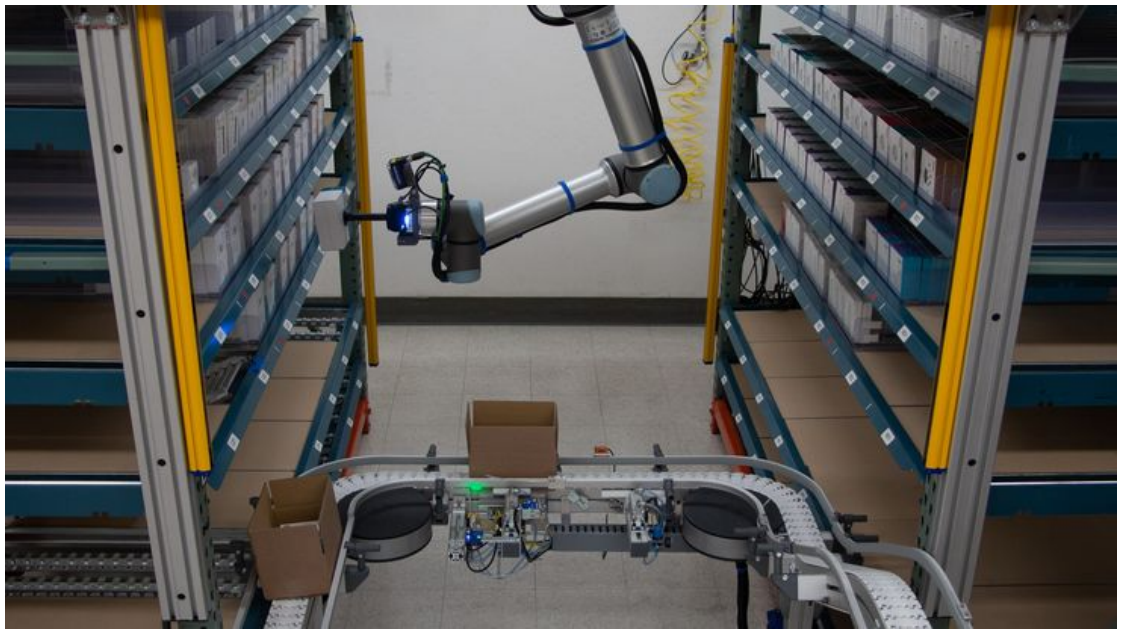


Рисунок 1.8 Universal Robots в логістиці

Сфери застосування в логістиці

Сортування товарів: UR-роботи використовуються для сортування вантажів на основі їхнього розміру, ваги чи інших характеристик.

Автоматизоване пакування: роботи здійснюють пакування товарів, оптимізуючи використання простору.

Переміщення коробок: роботи здатні переносити об'єкти між різними точками складу або виробничої лінії.

Складування: UR-роботи забезпечують точне складування товарів на полиці, включаючи маніпуляції у важкодоступних місцях[3].

Boston Dynamics Stretch

Stretch – це спеціалізований логістичний робот, розроблений компанією Boston Dynamics для автоматизації операцій у складських приміщеннях. Він оптимізований для роботи з коробками і палетами, що робить його ідеальним для компаній із великими обсягами вантажних перевезень (рис1.9).

Основні характеристики

Переміщення коробок: Stretch оснащений роботизованою "рукою" з високочутливим захватом, який може працювати з коробками різних розмірів і ваги до 23 кг; використовується для розвантаження та завантаження вантажівок, переміщення коробок між палетами або стелажми.

Мобільність: робот здатний вільно переміщатися по складу завдяки базі з колесами, оснащений сенсорами для навігації; інтегрована система стабілізації дозволяє працювати на нерівних поверхнях.

Швидкість і ефективність: Stretch може обробляти до 800 коробок на годину, що значно перевищує продуктивність людської праці; оптимізація алгоритмів дозволяє знижувати витрати часу на пошук коробок і виконання повторюваних рухів.

Інтелектуальна система навігації: оснащений системами машинного зору, які дозволяють розпізнавати об'єкти, аналізувати їхній розмір, вагу та позицію; легко інтегрується у вже існуючі складські системи управління.



Рисунок 1.9 Stretch — це багатоцільовий мобільний робот

Сфери застосування в логістиці

Розвантаження вантажівок: робот Stretch здатний швидко і точно розвантажувати коробки з транспортних засобів.

Комплектація замовлень: робот автоматично сортує товари для підготовки до відправлення клієнтам.

Оптимізація складських операцій: використовується для переміщення вантажів між різними зонами складу без залучення людської праці.

Приклад впровадження роботів у логістиці

Universal Robots (UR): використовується у компанії Amazon для автоматизації пакування та складування товарів у центрах виконання замовлень. Завдяки UR-роботам компанія зменшила витрати на обробку замовлень на 20%.

Boston Dynamics Stretch: впроваджений у логістичних центрах DHL для автоматизації розвантаження вантажівок, що дозволило скоротити час обробки вантажів на 50%.

Автоматизація логістики за допомогою роботів значно покращує ефективність і точність процесів, роблячи галузь більш стійкою до сучасних викликів.

1.2 Основні підходи та технології систем керування

Інверсна кінематика

Інверсна кінематика допомагає у визначенні необхідних кутів обертання суглобів для досягнення цільової позиції та орієнтації робочого інструменту. Це вирішується через математичні моделі, які враховують геометрію маніпулятора та різні обмеження. Використання алгоритмів інверсної кінематики дозволяє створювати точні та плавні траєкторії руху, що є критично важливим для таких застосувань, як збирання і монтаж деталей.

Пропорційно-інтегрально-диференційне (ПІД) регулювання

ПІД-регулятори застосовуються для підтримання заданих параметрів системи, наприклад, положення та швидкості маніпулятора. Регулятор обчислює помилку між заданим значенням та фактичним значенням і коригує керуючий сигнал таким чином, щоб мінімізувати цю помилку. Це дозволяє забезпечити стабільну та точну роботу маніпулятора навіть в умовах непередбачених зовнішніх впливів.

Використання датчиків

Датчики є невід'ємною частиною систем керування роботами-маніпуляторами. Вони забезпечують систему інформацією про стан робота та його оточення. Використання таких датчиків як енкодери, потенціометри, лідарами та ультразвуковими датчиками дозволяє забезпечити високу точність та надійність роботи. Інформація від датчиків використовується для коригування рухів маніпулятора в режимі реального часу.

Машинне навчання та штучний інтелект

Застосування алгоритмів машинного навчання та штучного інтелекту дозволяє роботам адаптуватися до змін у навколишньому середовищі та виконувати складні завдання. Нейронні мережі можуть використовуватися для розпізнавання об'єктів, планування траєкторій та оптимізації процесів керування. Це забезпечує роботам-маніпуляторам можливість навчатися на досвіді та постійно покращувати свою продуктивність. Особливості: використання алгоритмів машинного навчання для аналізу та адаптації; здатність до самооптимізації та навчання на основі минулого досвіду; складність і висока вартість впровадження. Приклади: роботи, які аналізують дефекти виробів під час контролю якості; логістичні роботи, які оптимізують маршрути на складі[4].

Ручне управління (без автоматизації)

Роботом управляє оператор, виконуючи всі дії вручну за допомогою пульта або інших пристроїв керування. Особливості: висока гнучкість у виконанні нестандартних завдань; низька продуктивність через залежність від оператора; не потребує складних алгоритмів. Приклади: ручне керування роботом-маніпулятором у рятувальних операціях; маніпуляції з роботизованим захватом під час хірургічних операцій.

Автоматичне управління

Робот виконує завдання без участі людини після початкового налаштування. Особливості: підвищена продуктивність і точність; необхідність попереднього програмування; обмежена адаптивність у разі непередбачених ситуацій. Приклади:

робот для автоматичного складання електронних компонентів; автономні роботи для збирання врожаю.

Напіваавтоматичне управління

Поєднує ручне управління з певним рівнем автоматизації. Оператор задає напрямки роботи, а робот виконує окремі операції автоматично. Особливості: гнучкість у виконанні завдань; використовується в складних умовах, де потрібен контроль людини; часто вимагає менших витрат на налаштування. Приклади: роботизовані візки на складах, які рухаються за маршрутом, заданим оператором; хірургічні роботи, що виконують дії під наглядом лікаря.

Відкриті системи (без зворотного зв'язку)

Управління відбувається без оцінки результату виконаної дії. Робот працює строго за заданим алгоритмом, не отримуючи інформацію про стан об'єкта чи навколишнього середовища. Особливості: простота конструкції та програмування; відсутність можливості коригувати дії у разі помилок; застосовуються у стандартних умовах. Приклади: друкарські роботи, які виконують шаблонні операції; роботи для різання металу за визначеними координатами.

Замкнуті системи (зі зворотним зв'язком)

Система отримує дані про стан об'єкта чи середовища після виконання кожної дії. Ця інформація використовується для корекції подальших дій. Особливості: висока точність і адаптивність; використання сенсорів і систем зчитування даних; складність і висока вартість реалізації. Приклади: роботи, які регулюють силу захвату залежно від ваги об'єкта; логістичні роботи, що коригують свої маршрути залежно від змін у складських умовах.

Мережеві протоколи та комунікації

Мережеві протоколи, такі як CAN, Ethernet та Modbus, забезпечують надійну передачу даних між різними компонентами системи. Це дозволяє інтегрувати роботів-маніпуляторів у складні виробничі системи та забезпечувати координацію їхньої роботи з іншими пристроями та машинами.

Системи візуального спостереження

Машинний зір є важливим інструментом для виконання точних маніпуляцій та розпізнавання об'єктів у реальному часі. Камери та алгоритми обробки зображень дозволяють роботам бачити навколишнє середовище, визначати положення та орієнтацію об'єктів, а також виконувати складні завдання з високою точністю.

Інтернет речей (IoT) і хмарні технології

Інтеграція роботів-маніпуляторів у мережу IoT дозволяє здійснювати віддалений моніторинг та керування їхньою роботою. Хмарні технології забезпечують зберігання та обробку великих обсягів даних, що дозволяє підвищити ефективність роботи роботів та забезпечити можливість їхнього навчання на основі зібраних даних[5].

Адаптивне управління (реакція на зміну умов)

Система може реагувати на зміни у зовнішньому середовищі, коригуючи свою поведінку в реальному часі. Для цього використовуються сенсори, які передають інформацію про поточний стан об'єктів або середовища. Особливості: використовуються датчики для збору даних про середовище; можливість роботи в нестандартних ситуаціях; складність налаштування системи. Приклади: робот-маніпулятор, який регулює силу захвату залежно від форми та матеріалу об'єкта; роботи для сортування вантажів, які адаптуються до різних розмірів коробок.

1.3 Використання Arduino у робототехніці

Простота використання. Arduino відомий своєю простотою використання. Середовище розробки (IDE) має інтуїтивно зрозумілий інтерфейс, який дозволяє початківцям швидко розпочати розробку. Базові проекти можуть бути створені за

лічені години, що робить Arduino ідеальним для навчання і швидкого прототипування.

Гнучкість і модульність. Платформа Arduino є модульною, що дозволяє легко додавати різні компоненти. Існують численні щити (shields) для розширення функціональності, такі як Wi-Fi, Bluetooth, дисплеї, двигуни і датчики. Це робить Arduino універсальною платформою для створення роботів різної складності та призначення.

Спільнота та ресурси. Arduino має величезну спільноту користувачів по всьому світу. Це означає, що можна легко знайти багато корисних матеріалів, таких як інструкції, відеоуроки, форуми та блоги. Це значно спрощує процес розробки, оскільки завжди можна знайти відповіді на свої питання або попросити про допомогу.

Вартість. Arduino є доступним за ціною, що робить його привабливим для студентів, хобістів та невеликих компаній. Крім того, існує велика кількість клонів та сумісних плат, які також є недорогими. Це дозволяє створювати масштабовані проекти з мінімальними витратами.

Програмування. Arduino використовує мову програмування, що базується на C/C++. Це дозволяє розробникам швидко оволодіти основами та зосередитися на логіці проекту. Крім того, Arduino IDE надає безліч бібліотек, які спрощують роботу з різними компонентами, такими як датчики, двигуни та дисплеї[6].

Приклади використання. Arduino широко використовується для створення різних типів роботів:

Мобільні роботи: Автономні роботи, які можуть рухатися у просторі, обминати перешкоди та виконувати різноманітні завдання.

Маніпулятори: Роботи, які можуть виконувати точні маніпуляції з об'єктами, такі як збирання та розбирання деталей.

Роботи-дрони: Літаючі роботи, що можуть виконувати задачі з спостереження, доставки і багато іншого.

Освітні роботи: Проекти для навчання студентів основам робототехніки та програмування.

Проекти з відкритим кодом. Існує безліч проектів з відкритим кодом на базі Arduino, які можна використовувати як основу для власних розробок. Ці проекти охоплюють широкий спектр застосувань, від простих сенсорних модулів до складних автономних систем. Використання відкритого коду дозволяє значно скоротити час на розробку та зосередитися на впровадженні інноваційних ідей.

1.4 Програмне забезпечення для керування роботами-маніпуляторами

Arduino IDE (Інтегроване середовище розробки) – це офіційне середовище програмування для Arduino, яке дозволяє писати, компілювати і завантажувати код безпосередньо на плату. Це просте у використанні середовище, яке підтримує безліч бібліотек і прикладів коду, що значно спрощує розробку.

Бібліотеки для роботи з маніпуляторами. Arduino має безліч бібліотек, які спрощують роботу з різними компонентами та датчиками. Деякі з них особливо корисні для керування роботами-маніпуляторами: Servo.h: Бібліотека для керування сервоприводами, що часто використовуються в маніпуляторах для точного позиціонування; AccelStepper.h: Бібліотека для керування кроковими двигунами, які також можуть бути використані у маніпуляторах для забезпечення плавного та точного руху.

Середовище обробки даних. Для обробки даних від датчиків та керування виконавчими механізмами можна використовувати різні програмні засоби, наприклад: Processing: Мова програмування та середовище розробки, яка часто використовується разом з Arduino для створення візуальних інтерфейсів; MATLAB: Використовується для аналізу даних та моделювання систем. MATLAB може взаємодіяти з Arduino для збору та обробки даних.

Інтеграція з іншими платформами. Arduino може інтегруватися з іншими платформами та мікроконтролерами для створення більш складних систем.

Наприклад: Raspberry Pi: Використання Raspberry Pi для обробки високорівневих задач, таких як обробка зображень або машинне навчання, у поєднанні з Arduino для керування низькорівневими виконавчими механізмами; ROS (Robot Operating System): Використовується для розробки програмного забезпечення для роботів. ROS підтримує Arduino, що дозволяє використовувати його як частину більш складних робототехнічних систем[7].

Програмні модулі для симуляції. Для тестування та моделювання роботи робота-маніпулятора можна використовувати спеціальні програмні середовища для симуляції: Gazebo: Платформа для моделювання роботів, що дозволяє створювати віртуальні середовища та тестувати алгоритми без необхідності фізичного прототипу; V-REP (CoppeliaSim): Програмне забезпечення для симуляції роботів, яке підтримує інтеграцію з Arduino та іншими платформами.

Інтернет речей (IoT) та віддалене керування. Arduino може бути використаний для створення проектів з підтримкою Інтернету речей (IoT), що дозволяє віддалено контролювати та моніторити робота-маніпулятора. Для цього можна використовувати такі бібліотеки та сервіси: ESP8266/ESP32: Мікроконтролери з вбудованою підтримкою Wi-Fi, які дозволяють підключити Arduino до Інтернету; Blynk: Платформа для створення мобільних додатків для віддаленого керування Arduino; MQTT: Протокол для обміну повідомленнями між пристроями IoT. Ці програмні засоби та підходи дозволяють створювати ефективні та гнучкі системи керування роботами-маніпуляторами на базі Arduino.

Висновок за розділом

У цьому розділі роботи був проведений детальний огляд існуючих систем керування роботами-маніпуляторами, основних підходів та технологій, що використовуються у цій галузі. Розглянуто промислові роботи, роботи для автоматизації складання, медичні роботи та роботи для сільського господарства, їхні особливості та застосування.

Також були проаналізовані основні методи керування, такі як інверсна кінематика, ПД-регулятори, використання датчиків, машинне навчання та штучний інтелект, мережеві протоколи та комунікації, системи візуального спостереження та Інтернет речей (IoT). Кожен з цих методів має свої переваги та недоліки і забезпечує різні аспекти керування роботами-маніпуляторами.

Крім того, було розглянуто використання платформи Arduino у робототехніці. Завдяки своїй простоті, доступності та модульності, Arduino є ідеальним вибором для створення різних роботизованих систем, як для початківців, так і для професіоналів. Платформа Arduino надає широкий спектр бібліотек та підтримується величезною спільнотою користувачів, що значно спрощує процес розробки та впровадження робототехнічних проектів.

Отже, основні висновки першого розділу полягають у тому, що для ефективного керування роботами-маніпуляторами існує велика кількість методів та технологій, які можна комбінувати для досягнення найкращих результатів. Використання платформи Arduino є обґрунтованим вибором завдяки її широким можливостям та підтримці спільноти. Усі розглянуті підходи та технології створюють надійний фундамент для подальшої розробки та реалізації системи керування роботом-маніпулятором.

РОЗДІЛ 2

МЕТОДОЛОГІЯ МАГІСТЕРСЬКОГО ДОСЛІДЖЕННЯ

2.1 Побудова структурно-логічної схеми магістерського дослідження

Метод (від грецького *methodos* – шлях до чого-небудь) – в найбільш загальному випадку означає засіб досягнення мети, спосіб дослідження явища, який визначає планомірний підхід до їх наукового пізнання та встановлення істини. Науковий метод – це спосіб пізнання явищ дійсності в їх взаємозв'язку та розвитку, спосіб досягнення поставленої мети і завдань дослідження і відповідає на запитання: «Як пізнавати?». Методика дослідження – це система правил використання методів, прийомів та способів для проведення будь-якого дослідження. Свідоме застосування науково обґрунтованих методів слід розглядати як найсуттєвішу умову отримання нових знань. Дослідник, який добре знає методи дослідження і можливості їх застосування, витрачає менше зусиль і працює успішніше, ніж той, хто у своєму дослідженні спирається лише на інтуїцію або діє за принципом «спроб і помилок». Звісно, що точні і правильні методи – не єдині компоненти, що забезпечують успішність наукового дослідження. Методи не можуть, наприклад, замінити творчу думку дослідника, його здібність аналізувати, робити висновки і передбачення. Але застосування правильних методів спрямовує хід думок дослідника, відкриває перед ним найкоротший шлях для досягнення мети і забезпечує таким чином можливість раціонально витратити енергію і час науковця. Кожний метод наукового пізнання слід розглядати як систему регулятивних принципів практичної і теоретичної діяльності людини. Методів пізнання об'єктивної дійсності відомо дуже багато. Правильний вибір методів дослідження потребує знання їх класифікації.

Розрізняють методи теоретичних та емпіричних досліджень. Такий розподіл методів завжди умовний, оскільки з розвитком пізнання один науковий метод може переходити з однієї категорії в іншу.

Методи наукових досліджень теоретичного рівня. На теоретичному рівні дослідження одним з найпоширеніших наукових методів є абстрагування 26 (таблиця 2.1). В цілому дієслово «абстрагувати» означає відокремлювати в думках одні властивості, зв'язки та відношення явища або об'єкта від інших. Відповідно зміст даного методу полягає у відверненні від неістотних ознак предмета дослідження й одночасному виділенні тих, які мають найбільший інтерес. Використання методу абстрагування на початковому етапі дослідження дає змогу превентивно вирішити наукову проблему, акцентуючи увагу на її основних аспектах [6].

Таблиця 2.1 – Наукові методи теоретичних досліджень

Методи	Характеристики
Абстрагування	Уявне відвернення від неістотних, другорядних ознак явища або об'єкта, зв'язків і відношень між ними та виокремлення кількох сторін, які цікавлять дослідника
Аналіз	Метод пізнання, який дає змогу гіпотетично поділити явище або об'єкт на окремі складові частини та досліджувати їх відокремлено одна від одної
Синтез	Метод, який є результатом об'єднання окремих складових частин явища або об'єкта в єдине ціле, досліджуючи загальні ознаки і властивості
Моделювання	Опосередкований метод дослідження явища або об'єкта, який ґрунтується на застосуванні моделі як засобу дослідження, коли вивчити явище або об'єкт не можливо
Аксіоматичний метод	Метод побудови наукової теорії, за яким деякі твердження приймаються без доведень, а всі інші знання виводяться з них відповідно до певних логічних правил
Ідеалізація	Гіпотетичне конструювання явища або об'єкта, яких немає в дійсності або які за суб'єктивних причин практично нездійсненні

Методи	Характеристики
Формалізація	Метод вивчення явища або об'єкта через відображення їх структури в знаковій формі за допомогою штучних мов, наприклад, математики, фізики, хімії тощо
Дедукція	Умовивід, у якому висновок про той чи той елемент множини робиться на підставі знання про загальні властивості усієї множини
Індукція	Умовивід від часткового до загального, від окремих фактів до узагальнень, коли на основі знань про частини предметів певного класу робиться висновок про клас загалом

Вибір конкретних методів дослідження диктується характером фактичного матеріалу, умовами і метою конкретного дослідження.

В науковій сфері методи аналізу і синтезу зіставляються між собою.

Якщо вивчити явище або об'єкт складно та недоцільно, використовується метод моделювання. Незважаючи на те, що даний метод пов'язаний з ризиковістю, він дає змогу ґрунтовно дослідити закономірності виникнення й існування того чи іншого явища або об'єкта, реально оцінити їх взаємозалежність. При порівнянні альтернативних моделей кращою вважають ту з них, якій відповідає мінімальне значення середнього ризику. Величина похибки відображає залежність між предметом дослідження та впливом низки екзогенних чинників.

Емпіричні методи наукових досліджень

На емпіричному рівні використовується широкий спектр методів наукового дослідження, які можна об'єднати в наступну групу (таблиця 2.2).

Таблиця 2.2 - Емпіричні методи наукових досліджень

Методи	Характеристики
Вимірювання	Визначення кількісних характеристик явища або об'єкта за допомогою спеціальних технічних пристроїв та одиниць виміру

Методи	Характеристики
Порівняння	Процес зіставлення явища або об'єкта з метою з'ясування подібності та відмінності між ними, а також знаходження загального, що може бути властивим двом чи кільком явищам або об'єктам дослідження
Спостереження	Системне, цілеспрямоване, спеціально організоване візуальне сприймання явища або об'єкта, опосередковане раціональним знанням
Експеримент	Метод вивчення явища або об'єкта, коли дослідник активно та цілеспрямовано впливає на нього для створення штучних чи використання природних умов, необхідних для виявлення його ознак і властивостей

Найскладнішою є методика експериментальних досліджень, як лабораторних, так і польових.

Натомість на теоретичному рівні використовується пізнавальна діяльність, в якій структура реального експерименту відтворюється в уяві, тобто розумовий експеримент.

Модель (лат. *modus* — міра, франц. — *modele* — зразок) — штучно створений об'єкт у вигляді схеми, креслення, логіко-математичних знакових формул, фізичної конструкції й т.п., що, будучи аналогічним (подібний, подібний) досліджуваному об'єкту, процесу відображає й відтворює в більше простому, зменшеному виді структуру, властивості, взаємозв'язки й відносини між елементами досліджуваного об'єкта, процесу, безпосереднє вивчення якого зв'язано якими-небудь труднощами, більшими витратами коштів або просто не-доступно, і тим самим полегшує процес одержання інформації про предмет, що цікавить нас, явищі, процеси

Пізніше були осмислені модельні властивості креслень, малюнків, карт — реальних об'єктів штучного походження, які дозволили втілити абстракцію більш високого рівня. Тобто наступний крок заключався в визнанні того, що моделями

можуть служити не тільки реальні об'єкти, а і абстрактні, ідеальні побудови, типовим прикладом яких є математичні моделі.

Коротко кажучи – Модель це системне відображення оригіналу.

Термін "модель" використовується в різних областях діяльності людини і як правило означає спрощену картину реальності або прообраз майбутнього.

У дослідженнях модель заміняє оригінал так, що її вивчення дає нові знання про об'єкт дослідження.

Термін "модель" в науці являє собою штучно створену систему, яка відображає або здатна відтворити основні сторони реальної системи, яка називається оригіналом.

Моделювання – це дослідження оригіналу не безпосередньо, а непрямим шляхом з допомогою штучно побудованої системи, яка називається моделлю.

Моделювання включає процес побудови, вивчення та примінення моделей.

Виникає запитання : для чого необхідне моделювання , у яких випадках необхідно розробляти модель, а коли проводити пряме дослідження оригіналу? При цьому слід відмітити, що оригінал може бути недоступним для безпосереднього дослідження з причини відсутності часу і засобів, або відсутності самого оригіналу (розробка проекту, тощо). Крім того, модель завжди простіша оригіналу і дослідження на ній обходиться дешевше і потребує менше часу. Модель дозволяє оперативно міняти умови досліду (що не завжди можливо із оригіналом), дозволяє багаторазово повторити експеримент до пізнання суті.

При цьому не передбачається повна ідентичність оригіналу оскільки в такому разі втрачаються переваги моделі і всяка суть моделювання, та і вона перестає бути моделлю, а це вже буде копія-двійник, копія-дублер. Модель повинна замінити оригінал лише в принципі, по суті.

В моделі оригінал можна уявити в чистому вигляді без сторонніх впливів і непотрібних деталей.

Серед відомих методів дослідження складних електромеханічних систем можна виділити фізичне, імітаційне, аналогове та математичне моделювання.

Метод фізичного моделювання передбачає дослідження на моделі, яка відтворює основні геометричні, фізичні, динамічні та функціональні характеристики та властивості оригіналу (але не обов'язково всі властивості) та має

великі можливості з визначення різноманітних якостей електромеханічних систем ті їх елементів. Він базується на трьох теоремах аналізу подібності.

Практично необмежені можливості з виявлення ефективного функціонування складної електромеханічної системи має метод математичного моделювання. Він ґрунтується на здатності математично описувати рівняннями різні за природою явища (процеси) та виявляти різні функціональні зв'язки між елементами системи.

Аналогове моделювання базується на однаковому для моделі та натури математичному описі та використовується для імітації на основі аналогової фізичної системи по її елементам. При цьому кожному з фізичних елементів натури в моделі відповідає певний еквівалент. У якості моделей використовуються електричні кола, складені з пасивних та активних елементів.

Імітаційне моделювання використовується частіше за всього при дослідженні складних електромеханічних систем та представляють собою процес формування моделі реальної системи та проведення над нею численних експериментів з метою виявлення шляхів її створення, вдосконалення та (або) ефективного використання.

При математичному моделюванні застосовуються аналітичні та статистичні моделі. Аналітичні моделі більш грубі, враховують менше число факторів, завжди потребують деяких припущень та спрощень. Проте вони більше пристосовані для пошуку оптимальних рішень.

Статистичні моделі у порівнянні з аналітичними більш точні, але громіздкі та погано доступні для огляду.

При математичному моделюванні взаємозв'язки і закономірності оригіналу відображаються в математичній формі.

Моделювання починається із всестороннього вивчення об'єкту дослідження і постановки задач (проблеми). Проблема і задачі повинні бути чітко сформульовані. Далі слід знайти в реальній формі чи розробити модель оригіналу, яка б відображала чи могла відобразити основні сторони оригіналу.

По розробленій моделі проводять власне експеримент. Змінюють в певних межах вхідні параметри і досліджують вихідні. В результаті таких експериментів

проверяють модель на адекватність відображення оригіналу, отримують нові дані про оригінал, визначають числові параметри позитивних і негативних змін.

Комп'ютерна модель (або чисельна)— комп'ютерна програма, яка реалізує абстрактну модель деякої системи. Комп'ютерні моделі використовуються для отримання нових знань про об'єкт, що моделюється або для приблизної оцінки поведінки математичних систем, які досить складні для аналітичного дослідження. Комп'ютерне моделювання є одним з ефективних методів вивчення складних систем. Комп'ютерні моделі простіші та їх зручніше досліджувати в силу можливості проводити обчислювальні експерименти в тих випадках, коли реальні експерименти немає можливості проводити через фінансові труднощі або фізичні перешкоди. Для підтримки математичного моделювання розроблені системи комп'ютерної математики, наприклад, *Maple*, *Mathematica*, *Mathcad*, *MATLAB*, *VisSim* та інші. В даній роботі використовується програма *Simulink* пакету *MATLAB*.

2.2 Структурно-логічна схема магістерського дослідження

Методологія - це система базисних принципів, методів, методик, способів і засобів їх реалізації в організації та побудові науково-практичної діяльності. Вона включає в себе методи теоретичного та емпіричного пізнання і являє собою алгоритм пошуку і досягнення мети.

Основне призначення методики дослідження полягає у тому, щоб на основі відповідних принципів (вимог, умов, обмежень тощо) забезпечити успішне вирішення визначених мети і завдань наукового дослідження.

Структурно-логічна схема є однією з форм планування роботи і візуалізації процесу, яка дозволяє з одного боку описувати процедуру (алгоритм) дослідження і досягнення мети тобто сформулювати робочу гіпотезу дослідження, а з другого — установити логічну послідовність виконання структурних розділів і підрозділів магістерської роботи.

Спочатку вибирається тема дослідження та обґрунтовується її актуальність та складається індивідуальний календарний план роботи.

Ведеться пошук літератури за темою, її аналіз, складання бібліографії.

Розробляється перший варіант теоретичної частини:

- постановка мети і конкретних завдань дослідження;
- визначення об'єкту й предмета дослідження;
- вибір методів (методики) проведення дослідження.

Виконується опис процесу дослідження згідно з обраним методом. Моделювання процесів у електроприводі гвинтового конвеєра проводиться шляхом математичного моделювання з використанням пакету Simulink програми MATLAB.

Наводиться інтерпретація отриманих результатів дослідження.

На підставі виконаних розрахунків та отриманих характеристик проводиться узагальнення отриманих теоретичних результатів.

На заключному етапі роботи проводиться розробка заходів охорони праці, впровадження результатів магістерського дослідження у навчальний процес.

2.3 Вибір методу вирішення поставленої задачі дослідження

Дана магістерська робота не передбачає проведення експериментальних досліджень, має теоретичний характер, тому основним методом для відображення отриманих результатів є моделювання.

Моделювання як метод наукового пізнання виникло в зв'язку з необхідністю вирішення завдань, які з тих чи інших причин не можуть бути вирішені безпосередньо. Вони виникають у випадках, коли об'єкт або недосяжний для дослідника, або він ще не існує і потрібно обрати оптимальний варіант його створення, або дослідження реального об'єкта вимагає багато часу та економічно не вигідне тощо. При моделюванні використовується принцип аналогії, ґрунтуючись на якому в процесі наукового дослідження висуваються гіпотези, тобто передбачення, що будуються на невеликій кількості дослідних даних, спостережень, інтуїтивних припущень, перевірка правильності яких здійснюється шляхом експерименту [17].

Модель будується для того, щоб відобразити характеристики об'єкта (елементи, взаємозв'язки, структурні та функціональні властивості), суттєві з точки зору мети дослідження. Отже, моделювання пов'язане зі спрощенням, огрубленням прототипу, абстрагуванням від ряду його властивостей, ознак, сторін.

Процес дослідження реальних систем, що охоплює побудову моделі, дослідження її властивостей і перенесення одержаних відомостей на реальну систему, називають моделюванням.

Моделі поділяються на такі види:

- інтуїтивні (виражені на папері);
- фізичні (макети, муляжі, що дозволяють наочно ілюструвати, саме ті процеси, які мають місце у природі. За допомогою таких моделей можна вивчати вплив окремих параметрів на певний фізичний процес);
- математичні (дозволяють кількісно досліджувати явища, які важко піддаються вивченню за допомогою фізичних моделей);
- знакові (вираженні мовою, алгоритмами, графічно, математично);
- функціональні (описують залежності між входом і виходом системи);
- структурні (описують оргструктури систем).

Послідовність робіт при моделюванні виробничої системи.

- дослідження модельованої системи і постановка задачі - вивчення літературних джерел, проведення досліджень, тощо;
- формалізація задачі - виявлення особливостей системи, виділення головного, основного. На цьому етапі визначають, які параметри системи, яку ми моделюємо необхідно відобразити в моделі;
- розробка математичної моделі задачі і її запис в структурній формі.

Розробити математичну модель задачі це означає побудувати спеціальну таблицю чи схему, у якій усі економічні, технологічні та інші виробничі умови і вимоги виражені у вигляді нерівностей і рівнянь і об'єднані цільовою функцією. Математична модель може бути подана у структурній, загальній символічній чи числовій формі. Послідовність розробки моделі може бути різною, але зручніше спочатку скласти систему змінних величин, записати систему обмежень, а потім

цільову функцію. В моделі виділяють змінні величини, константи і коефіцієнти.

Змінні в моделі відображають невідомі величини;

- аналіз кількісних залежностей параметрів задачі;
- збір інформації і її обробка;
- побудова числової моделі;
- вибір математичного методу вирішення задачі;
- вирішення задачі на ЕОМ;
- аналіз результатів вирішення і коригування моделі;
- вирішення задачі на ЕОМ по скоригованій моделі.

Головними рівнями дослідження та моделювання систем є мікро- та макрорівень. Мікрорівневе моделювання системи пов'язане з детальним описом кожного компонента системи, дослідженням її структури, функцій, взаємозв'язків, тощо. Практична реалізація найважливішого етапу мікромодельювання – виявлення елементів системи та взаємозв'язків між ними – пов'язана з необхідністю подолання суперечності між бажанням повного дослідження кожної з підсистем та елементів системи, реальною можливістю дослідити при цьому структуру системи загалом і принципи її функціонування.

Макрорівневе моделювання полягає в ігноруванні детальної структури системи та вивченні лише загальної поведінки системи як єдиного цілого.

У процесі дослідження реальних систем і побудови їх моделей використовуються різні методи моделювання, що залежать від характеристик об'єкта, рівня знань про нього, мети дослідження та вимог до моделі.

Аксіоматичне моделювання полягає у відповідній інтерпретації та переведенні змістовного опису системи на мову чітких математичних термінів і відношень, у процесі чого усуваються неясності, суперечності, неповнота або надлишковість, які властиві вербальному описові системи.

Емпірико-статистичне моделювання використовує широко відомий кібернетичний принцип “чорної скрині”, що не дозволяє отримати модуль структури системи, причинно-наслідкових зв'язків і механізмів її функціонування.

В результаті моделювання отримують моделі типу “вхід - вихід”, які базуються на теоретичних гіпотезах про форми взаємозв’язку між входами і виходами системи.

Оптимізаційне моделювання передбачає включення у модель як взаємозв’язків між змінними та параметрами, так і критерії якості функціонування системи.

Імітаційні моделі складних систем надзвичайно поширені внаслідок своєї універсальності, можливості проведення чисельних експериментів, передбачення різноманітних змін.

За мірою повноти опису моделювання поділяють на повне, неповне та наближене. Повне моделювання передбачає побудову моделі, адекватної об’єкту дослідження у просторі та часі. Для неповного моделювання ця адекватність не зберігається. При наближеному моделюванні беруться до уваги лише найважливіші аспекти системи.

Дослідження математичної моделі дає змогу отримати характеристики реального об’єкта чи системи. Вигляд математичної моделі залежить як від природи системи, так і від завдань дослідження.

Математична модель системи містить, як правило, опис множини можливих станів системи та закон переходу з одного стану в інший. Математичне моделювання охоплює імітаційне, інформаційне, структурне, ситуаційне тощо.

Розрахунок задач магістерського дослідження проводиться за допомогою електронної обчислювальної техніки за допомогою програми Mathcad. На базі складених диференціальних рівнянь проводиться математичне моделювання системи керування електропривода. За допомогою програми Simulink досліджуються перехідні характеристики системи при зміні задаючих параметрів.

Слід зазначити, що метод моделювання – вивчення явищ за допомогою моделей – сьогодні має особливе значення для вирішення складних фізичних завдань, чому значною мірою сприяє зростанню продуктивності комп’ютерної техніки. Саме тому в цьому магістерському дослідженні як основа використаний метод моделювання.

РОЗДІЛ 3

ВИБІР ЕЛЕМЕНТІВ СХЕМИ СИСТЕМИ КЕРУВАННЯ

3.1 Опис Arduino Uno R3 та його можливостей

Arduino Uno R3 - це одна з найпопулярніших мікроконтролерних плат, яка базується на мікроконтролері ATmega328P (рис.3.1). Вона має такі характеристики та можливості:

Технічні характеристики: мікроконтролер: ATmega328P; робоча напруга: 5 В; вхідна напруга (рекомендована): 7-12 В; вхідна напруга (межі): 6-20 В; цифрові входи/виходи: 14 (з яких 6 можуть використовуватися як ШІМ-виходи); аналогові входи: 6; максимальний струм для одного входу/виходу: 40 мА; флеш-пам'ять: 32 КВ (ATmega328P) з яких 0,5 КВ використовуються для завантажувача; SRAM: 2 КВ (ATmega328P); EEPROM: 1 КВ (ATmega328P); частота: 16 МГц

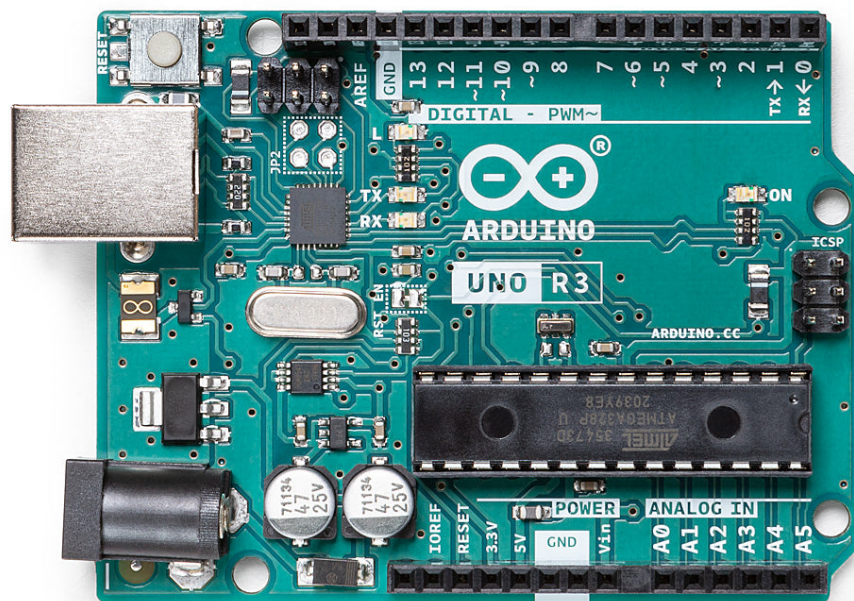


Рисунок 3.1 Arduino Uno R3

Можливості: Програмування через USB: Arduino Uno R3 має USB-порт, який дозволяє легко завантажувати програми на плату через Arduino IDE. Він також може бути живлений через цей USB-порт; Підтримка щитів (shields): Uno підтримує різноманітні додаткові плати розширення, які дозволяють додати різні функції, такі як Wi-Fi, Bluetooth, різні сенсори і модулі; Робота з аналоговими та цифровими сигналами: Плата може зчитувати та генерувати як аналогові, так і цифрові сигнали, що робить її універсальною для різних завдань; Інтерфейси зв'язку: Uno підтримує UART (послідовний зв'язок), I2C та SPI, що дозволяє підключати різні периферійні пристрої та модулі.

ШИМ (PWM): Підтримка широтно-імпульсної модуляції дозволяє регулювати потужність двигунів, яскравість світлодіодів та інші параметри.

Зв'язок з комп'ютером: Arduino Uno може бути використаний для створення різних інтерфейсів з комп'ютером, таких як серійний монітор для зчитування та відправки даних.

Простота використання: Завдяки великій кількості онлайн-ресурсів, прикладів, бібліотек та форумів, розробка з Arduino Uno є досить простою навіть для початківців.

Можливості програмування:

Програмування через IDE Arduino: Це середовище розробки надає простий інтерфейс для написання, компіляції та завантаження коду в Arduino Uno. Воно підтримує мову програмування, засновану на C/C++.

Бібліотеки: Arduino IDE має доступ до великої кількості бібліотек, які дозволяють розширити можливості плати. Наприклад, бібліотеки для роботи з датчиками, двигунами, Wi-Fi та іншими модулями.

Sketchbook: Інтерфейс, який дозволяє зберігати та організовувати всі ваші проекти (скетчі) в одному місці для легкого доступу та управління.

Можливості розширення: Щитки (shields) – це додаткові плати, які можна встановлювати на Arduino для додавання нових функцій. Наприклад, Ethernet

Shield для мережевих підключень, Motor Shield для керування двигунами або Sensor Shield для підключення різних датчиків.

Сумісність з іншими модулями: Arduino Uno може працювати з різними модулями, такими як RFID, GPS, Bluetooth, Wi-Fi, OLED-дисплеї, і багато інших. Це дозволяє створювати комплексні проекти, які включають бездротовий зв'язок, відстеження місця розташування, відображення даних та багато іншого.

Енергоспоживання: Ефективність енергоспоживання: Плата Arduino Uno може працювати від різних джерел живлення, включаючи USB, батареї та адаптери змінного струму. Це робить її універсальною для використання як у стаціонарних, так і в мобільних проектах[8].

Захист від перенапруги: Вбудовані захисні схеми допомагають запобігти пошкодженню плати від перенапруги або неправильної полярності підключення живлення.

Оновлення та підтримка: Arduino постійно оновлюється та підтримується розробниками, що забезпечує стабільність та нові можливості для плати.

3.2 Опис маніпулятора

Маніпулятор – це робототехнічний пристрій, що складається з кількох рухомих з'єднань і призначений для виконання різних маніпуляцій з об'єктами. Для реалізації сконструйовано маніпулятор, деталі якого були роздруковані на 3D принтері (рис.3.2). Він має наступні характеристики:

Кількість ступенів свободи

Чотири ступенів свободи:

Обертання основи: Поворот всього маніпулятора навколо вертикальної осі. Це дозволяє маніпулятору обертатися вліво і вправо.

Підняття та опускання плеча: Рух сегмента "плече" вгору і вниз, що дозволяє маніпулятору змінювати висоту захоплюючого механізму.

Підняття та опускання ліктя: Рух сегмента "лікоть" вгору і вниз, що дозволяє змінювати кут нахилу маніпулятора та його досяжність.

Обертання грипера: Поворот захоплюючого механізму (грипера), що дозволяє маніпулятору захоплювати об'єкти під різними кутами.



Рисунок 3.2 Сконструйований маніпулятор

Робочий діапазон і можливості

Діапазон руху: Основа: 0-180 градусів; Плече: 0-90 градусів; Лікоть: 0-90 градусів; Зап'ястя поворот: 0-180 градусів.

Точність позиціонування: завдяки використанню високоточних сервомоторів MG996R, маніпулятор забезпечує високу точність позиціонування з мінімальними похибками.

Вантажопідйомність: маніпулятор може піднімати і переміщувати об'єкти вагою до 1 кг, що робить його підходящим для різних навчальних і дослідницьких завдань.

Додаткові функції

Плати розширення: Використання шпідів для підключення додаткових датчиків і модулів, забезпечуючи гнучкість і функціональність системи.

Програмне забезпечення: Написання коду в Arduino IDE для керування рухами маніпулятора. Програма дозволяє налаштовувати та координувати рухи всіх ступенів свободи маніпулятора[9].

3.3 Підбір компонентів та системи електроприводу

Сервомотори – це двигуни, які можуть обертатися на певний кут за допомогою зворотного зв'язку. Вони зазвичай використовуються для точного позиціонування. В ході аналізу обрано наступні сервомотори:

SG90 Micro Servo: Маленький і доступний сервомотор, популярний для невеликих проєктів. Володіє крутним моментом близько 1,8 кг/см при 4,8 В і може обертатися на 180 градусів.

MG996R: Потужніший сервомотор з металевими шестернями, здатний обертатися на 180 градусів з крутним моментом до 9,4 кг/см при 6 В (рис.3.3). Ідеальний для більш важких застосувань, де потрібна висока потужність і точність.



Рисунок 3.3 Сервомотор MG996R 180 градусів

Крокові двигуни розділяють оберт на рівні кроки. Вони можуть точно переміщатися з одного положення в інше без необхідності зворотного зв'язку. Для реалізації надійності та точності обрано двигуни:

28BYJ-48: Популярний кроковий двигун з вбудованим редуктором, що забезпечує точність і велику силу. Має 64 кроки на оберт, але з редуктором отримуємо 2048 кроків на оберт.

NEMA 17: Стандартний кроковий двигун, широко використовується в 3D-принтерах та інших проектах (рис.3.4). Має 200 кроків на оберт (1,8 градуса на крок) і може розвивати великий крутний момент до 4,2 кг/см.



Рисунок 3.4 Кроковий двигун NEMA 17 17HS4401

Платформа для маніпулятора – це основна конструкція, яка підтримує всі інші компоненти та забезпечує їх стабільність і функціональність. Для реалізації нашого проекту ми вибрали конструкцію з алюмінієвих профілів. Вони мають кілька переваг, які роблять їх ідеальним вибором для створення надійного та гнучкого маніпулятора. Для реалізації всіх вимог для конструкції маніпулятора було використано алюмінієвий профілів типу 2020, що забезпечить міцність, стабільність і гнучкість у налаштуванні. Це дозволить створити надійну платформу для встановлення всіх необхідних компонентів і легко модифікувати конструкцію. Дана платформа має наступні переваги:

Міцність і легкість: Алюмінієві профілі забезпечують високу міцність при малій вазі, що важливо для портативності маніпулятора.

Легкість у збірці: Профілі типу 2020 легко збираються за допомогою стандартних кріплень і дозволяють швидко налаштовувати конструкцію.

Можливість модифікації: Пазова структура профілів дозволяє додавати нові елементи, датчики і двигуни без значних змін конструкції[10].

3.4 Датчики для вимірювання позиції та зворотного зв'язку

Датчики для вимірювання позиції та зворотного зв'язку є важливими компонентами для точного керування роботом маніпулятором. Вони дозволяють вимірювати положення, швидкість і напрямок руху кожного елемента маніпулятора, що необхідно для коректної роботи системи.

Енкодери

Оптичні енкодери: використовують оптичні датчики для визначення позиції валу або осі. Вони можуть бути з відкритим або закритим контуром. Енкодери з відкритим контуром не мають зворотного зв'язку і часто використовуються для простих задач, тоді як енкодери з закритим контуром забезпечують точніше відслідковування позиції.

Магнітні енкодери: використовують магнітні поля для визначення позиції. Вони менш чутливі до пилу і бруду, тому можуть бути кращими для використання в умовах, де є можливість забруднення.

KY-040 Rotary Encoder – популярний енкодер, який може бути використаний для вимірювання кутового положення валів маніпулятора (рис.3.5). Цей енкодер має два виходи, які можна підключити до Arduino для відслідковування кута обертання.

Потенціометри (аналогові датчики позиції)

Потенціометри є одним з найпростіших способів вимірювання кута обертання в сервомоторах або ланках маніпулятора. Вони перетворюють механічний рух у змінний електричний сигнал (напругу), пропорційний зміні кута повороту. Переваги: легкість у використанні та інтеграції з Arduino; дешеві та доступні; висока точність вимірювання кута. Приклад використання: Потенціометри можуть бути встановлені на валах сервомоторів або на ланках маніпулятора для вимірювання їх кута. Програмне забезпечення Arduino може використовувати ці значення для корекції руху.



Рисунок 3.5 Полярний енкодер KY-040 Rotary Encoder

KY-040 – це цифровий потенціометр, який добре підходить для використання на поворотних осях.

Bourns 10k Ohm – класичний потенціометр, який може бути використаний для точного вимірювання позицій (рис.3.6).



Рисунок 3.6 Класичний потенціометр Bourns 10k Ohm

Датчики сили/моменти (датчики тиску, тензометричні датчики).

Датчики сили використовуються для вимірювання сили, яку маніпулятор прикладає до об'єктів. Це дозволяє визначити, чи надто сильно маніпулятор стискає об'єкт або, навпаки, чи не достатньо сильно для захоплення. Переваги: допомагають коригувати рухи маніпулятора, щоб уникнути пошкоджень об'єктів; можуть бути використані для захоплення об'єктів з точним контролем сили. Приклад використання: Датчики сили можуть бути встановлені на захоплюючому механізмі маніпулятора для контролю сили захоплення або сили, яку маніпулятор прикладає до об'єкта.

FSR (Force Sensitive Resistor) – датчики тиску, які можна використовувати для визначення сили захоплення.

HX711 + Tension Sensors – тензометричні датчики, які використовуються для вимірювання сили або крутного моменту в роботизованих системах(рис.3.7).

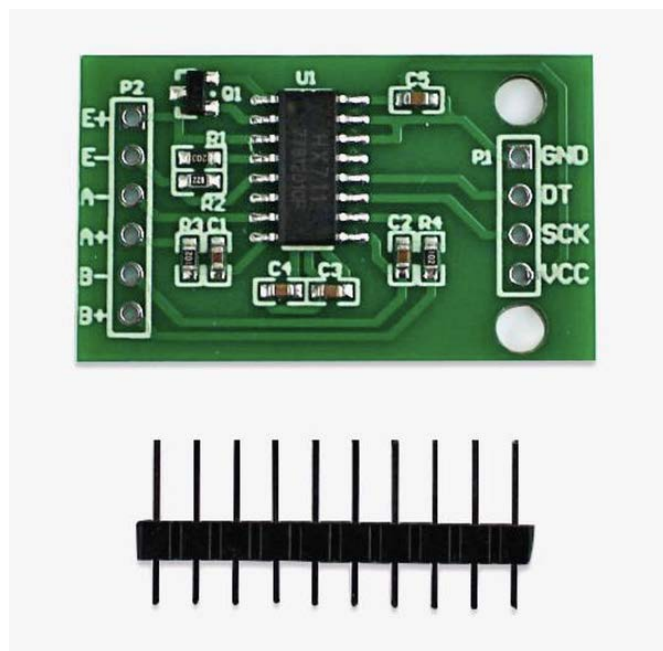


Рисунок 3.7 Двухканальний модуль датчиків ваги (тензодатчиків) HX711

Ультразвукові датчики відстані

Ультразвукові датчики вимірюють відстань до об'єкта, використовуючи принцип відбиття ультразвукових хвиль. Це дозволяє маніпулятору уникати

перешкод під час руху та маніпулювання об'єктами. Переваги: простота використання та інтеграції з Arduino; можуть бути використані для вимірювання відстані до об'єктів у безпечній зоні. Приклад використання: ультразвукові датчики можуть бути встановлені на кінці маніпулятора або на базовій платформі для відслідковування об'єктів і перешкод у радіусі дії маніпулятора.

HC-SR04 – найпоширеніший ультразвуковий датчик для вимірювання відстані.

MaxBotix MB1000 – більш точний ультразвуковий датчик для довших відстаней.

Інфрачервоні датчики відстані

Інфрачервоні датчики вимірюють відстань до об'єктів на основі відбиття інфрачервоних хвиль. Це дозволяє маніпулятору точніше визначати положення об'єктів в обмеженому просторі. Переваги: швидка реакція та невисока вартість; можна використовувати для виявлення об'єктів у близькому радіусі. Приклад використання: ІЧ-датчики можуть бути використані для виявлення об'єктів на кінці маніпулятора або на робочій поверхні, щоб маніпулятор не зіткнувся з ними.

Sharp GP2Y0A21YK0F – ІЧ датчик для вимірювання відстані на кілька сантиметрів (рис.3.8).

TCS3200 – ІЧ датчик для точного визначення відстані і кольору об'єктів.

Лінійні датчики (потенціометри або оптичні датчики)

Лінійні датчики використовуються для вимірювання переміщення по лінії, що може бути корисно для точного позиціонування частин маніпулятора в напрямку переміщення. Переваги: висока точність вимірювань; легкість у реалізації на механізмах з лінійним рухом.

LVDT (Linear Variable Differential Transformer) – датчик для точного вимірювання лінійних переміщень.

ALPS RKJXV 10K – потенціометр для лінійного вимірювання.



Рисунок 3.8 Sharp GP2Y0A21YK0F

3.5 Вибір додаткових компонентів

Зовнішнє джерело живлення для Arduino. Arduino Uno отримує живлення через USB або через зовнішнє джерело живлення через роз'єм DC IN (наприклад, через адаптер 9V). Однак живлення через USB зазвичай недостатньо для забезпечення стабільної роботи зовнішніх компонентів, таких як сервомотори, тому для проекту потрібно підключити зовнішнє джерело живлення.

Рекомендоване джерело живлення для Arduino:

Адаптер 9V: Зазвичай для Arduino Uno використовують адаптери на 9V, які підключаються через стандартний роз'єм DC IN. Це дозволяє забезпечити стабільне живлення для самої плати Arduino.

Моделі адаптерів:

MeanWell HDR-15-9 — компактний адаптер, що забезпечує стабільне живлення для Arduino.

KEDU 9V 1A — адаптер на 9V з вихідним струмом 1A, що добре підходить для Arduino.

Акумуляторна батарея 9V: Для можливого варіанту забезпечення мобільності системи, можна використовувати акумулятор 9V (наприклад, літієвий) з відповідним зарядним пристроєм: Li-ion 18650 3.7V з використанням модуля для підвищення напруги до 9V або 12V; акумулятори Li-Po 9V з регулюванням напруги через перетворювач.

Живлення для сервомоторів. Сервомотори мають більш високі вимоги до живлення, ніж сама плата Arduino, і потребують окремого джерела живлення, оскільки піні Arduino можуть не забезпечити достатній струм для коректної роботи сервомоторів. Типова напруга для сервомоторів варіюється від 4.8V до 6V залежно від моделі мотора.

Адаптер 5V або 6V: Оскільки більшість сервомоторів працюють при напрузі 5V або 6V, для них потрібен окремий блок живлення, що забезпечить необхідну потужність: MeanWell HDR-15-5 — стабільне джерело живлення на 5V, що підходить для живлення кількох сервомоторів; KEDU 6V 2A — адаптер на 6V, що забезпечує достатній струм для живлення кількох сервомоторів одночасно.

Акумуляторні батареї: В залежності від вимог системи, можна використовувати акумуляторні батареї, що забезпечують необхідну напругу: Li-ion 18650 3.7V: Через модуль перетворення напруги можна отримати 5V або 6V для сервомоторів; Li-Po 6V: Підходить для живлення декількох сервомоторів.

Реле та транзистори для управління живленням сервомоторів. Реле та транзистори використовуються для керування подачею живлення на сервомотори маніпулятора через Arduino. Це дозволяє знизити навантаження на піні Arduino, оскільки сервомотори можуть споживати значний струм, а піні Arduino не здатні забезпечити таку кількість струму без ризику пошкодження.

Реле електричний компонент, який використовує малий струм для управління більшим струмом. За допомогою реле можна підключати та відключати живлення сервомоторів через піні Arduino, не перевантажуючи його.

Транзистори використовуються для комутації струму між джерелом живлення та сервомоторами. Вони виконують функцію посилення сигналу від Arduino для управління високими струмами. Для управління сервомоторами зазвичай використовуються NPN транзистори або MOSFET транзистори.

Обрані моделі транзисторів: 2N2222 — популярний NPN транзистор для малих навантажень; IRLZ44N — MOSFET транзистор, який може працювати при низьких напругах і підходить для сервомоторів.

Приклад підключення: Arduino управляє транзистором через базовий піні, який, у свою чергу, керує подачею живлення на сервомотори через колектор (для NPN транзистора) або через сток (для MOSFET). Це дозволяє ардуїно не перевантажувати свої піні.

Резистори та конденсатори для стабілізації сигналів і захисту від перешкод. Резистори та конденсатори використовуються для фільтрації сигналів, зниження електричних шумів та захисту від перенапруг. Вони допомагають стабілізувати роботу всієї системи, особливо коли в роботі використовуються високочутливі компоненти, як серводвигуни, датчики та Arduino.

Резистори використовуються для обмеження струму, який протікає через компоненти, що захищає їх від пошкодження. Резистори можуть також використовуватися для розподілу сигналів у колі та стабілізації вихідних параметрів.

Фільтраційні конденсатори: Встановлюються на живленнях сервомоторів або Arduino для зменшення високочастотних перешкод, які можуть виникати під час роботи двигунів.

Конденсатори на вході/виході: Використовуються для згладжування пульсацій напруги, що може покращити стабільність роботи системи.

Типові значення ємностей для конденсаторів — 100nF для фільтрації і 100uF для стабілізації напруги.

Приклад використання: сервомотор, що підключений до джерела живлення, потребує встановлення конденсатор 100uF для зменшення імпульсних перешкод. Резистори використовуються для обмеження струму у ланцюгах живлення.

Модулі зв'язку для віддаленого керування. Модулі зв'язку дозволяють здійснювати віддалене керування маніпулятором через бездротові мережі. Це дає змогу інтегрувати ваш маніпулятор у більші системи автоматизації, а також контролювати роботу маніпулятора на відстані.

Bluetooth (HC-05, HC-06): HC-05 — один із найбільш популярних модулів Bluetooth для Arduino(рис.3.9). Він дозволяє забезпечити з'єднання між Arduino і мобільним телефоном або комп'ютером. Модуль підтримує серійний протокол (UART), що дозволяє швидко передавати дані і команди для управління маніпулятором.

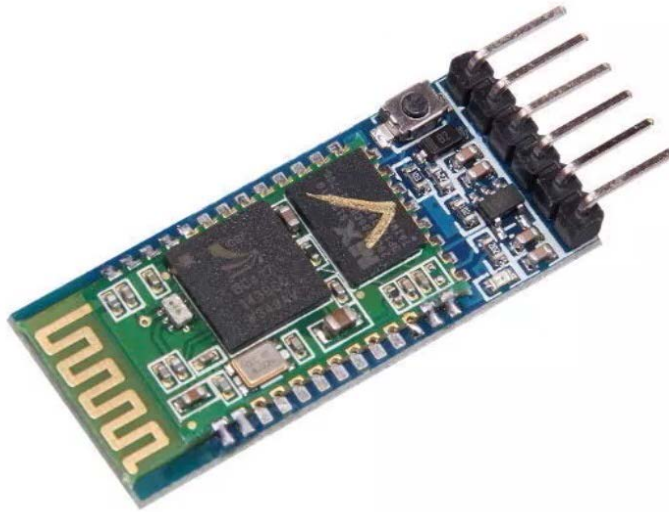


Рисунок 3.9 Bluetooth HC-05/06

Wi-Fi (ESP8266, ESP32): ESP8266 — популярний Wi-Fi модуль для Arduino, що дозволяє підключити систему до мережі Wi-Fi і керувати маніпулятором через веб-інтерфейс або мобільний додаток(рис.3.10); ESP32 — більш потужний варіант, який також включає Bluetooth, що дозволяє працювати одночасно з двома типами бездротових з'єднань. Обидва ці модулі підтримують MQTT, HTTP або WebSocket для віддаленого керування через інтернет або локальну мережу.

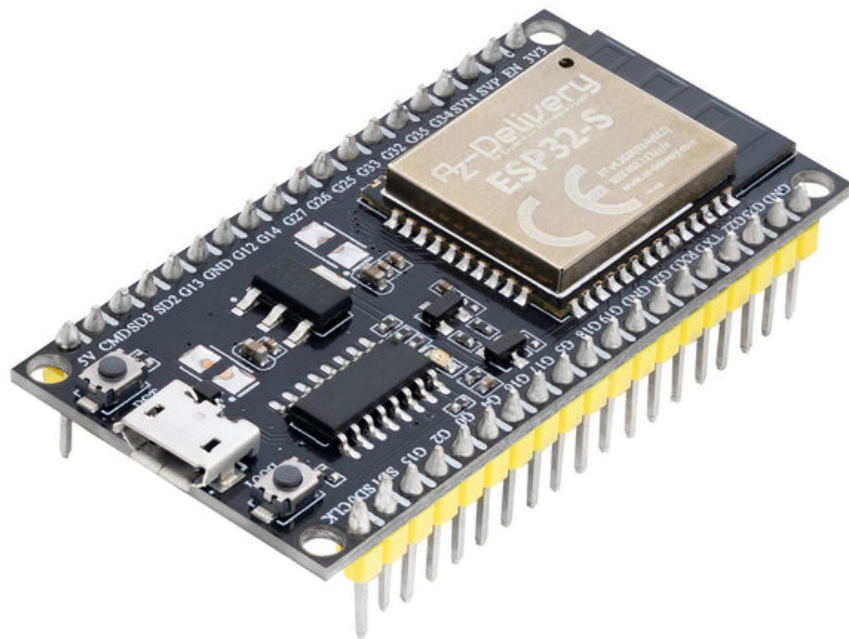


Рисунок 3.10 Wi-Fi модуль ESP8266

Для керування маніпулятором через мобільний додаток, можна використовувати Bluetooth HC-05. Користувач може надсилати команди на Arduino через мобільний додаток для управління сервомоторами та датчиками.

Для інтеграції в IoT систему або для управління через веб-інтерфейс можна використовувати ESP8266 або ESP32. Веб-додаток або мобільний додаток може відправляти HTTP запити до Arduino, а також отримувати дані з датчиків для моніторингу стану маніпулятора.

Діоди для захисту від перенапруг: Використовуються для захисту електронних компонентів від можливих перенапруг, які можуть виникати при вмиканні та вимиканні двигунів або при іншому стрибку напруги.

Мікросхеми для стабілізації напруги: Використовуються для забезпечення стабільного живлення для Arduino та інших компонентів, щоб уникнути падінь напруги, що можуть спричинити помилки в роботі[11].

3.6 Опис процесу розробки та методології

Визначення вимог до системи

На початковому етапі розробки системи керування роботом-маніпулятором необхідно визначити основні завдання, які система має виконувати. Це дозволяє чітко сформулювати вимоги до системи та забезпечити її ефективну реалізацію. Для нашої роботи можна виділити наступні завдання:

Система повинна бути здатна захоплювати об'єкти різної форми і розміру за допомогою маніпулятора. Визначення точного місця захоплення об'єктів за допомогою датчиків та обробки сигналів. Після захоплення об'єктів маніпулятор має забезпечити їхнє переміщення до заданої точки. Забезпечення плавного і точного руху маніпулятора з урахуванням траєкторії.

Точне позиціонування об'єктів у просторі з урахуванням заданих координат. Використання алгоритмів зворотного зв'язку для корекції позиції у реальному часі.

Визначення критеріїв ефективності. Для оцінки успішності реалізації проекту необхідно визначити конкретні критерії ефективності системи керування. Це дозволить об'єктивно оцінити роботу системи та внести необхідні корективи в процесі розробки.

Точність. Точність позиціонування маніпулятора має бути не меншою за ± 0.1 мм, що дозволить забезпечити високу точність обробки та маніпуляцій. Система повинна забезпечувати захоплення об'єктів з високою точністю, враховуючи їх форму та розмір.

Швидкість. Швидкість виконання основних операцій (захоплення, переміщення, позиціонування) повинна бути оптимальною для забезпечення ефективної роботи системи без втрати точності. Час реакції системи на команди користувача або зовнішні події має бути мінімальним для забезпечення швидкої та надійної роботи.

Надійність. Система повинна бути здатною працювати безперервно протягом тривалого часу без збоїв і поломок. Система повинна мати механізми виявлення та усунення несправностей для забезпечення безперебійної роботи.

Енергоспоживання. Система має споживати мінімальну кількість енергії для забезпечення економічної роботи. Визначення джерел живлення (блоки живлення, акумулятори), які забезпечують стабільну роботу всіх компонентів системи[12].

Апаратні обмеження. Маніпулятор UR3 від Universal Robots має максимальну вантажопідйомність 3 кг, що слід врахувати при виборі об'єктів для маніпуляцій. Робочий об'єм маніпулятора визначає зону, в межах якої він може виконувати операції. Необхідно врахувати цей параметр при проектуванні траєкторій руху.

Висновки за розділом

У розділі 3 було виконано підбір і аналіз основних компонентів для створення роботизованого маніпулятора з використанням мікроконтролера Arduino Uno R3. Зокрема, розглянуто технічні характеристики Arduino Uno R3, його функціональні можливості та гнучкість у роботі з периферійними пристроями, що забезпечує ефективне управління маніпулятором.

Сконструйовано маніпулятор за допомогою 3D принтеру з чотирма ступенями свободи, наведено його основні функціональні можливості, діапазони руху та характеристики точності й вантажопідйомності. У результаті дослідження було визначено, що даний маніпулятор є придатним для створення навчальних і дослідницьких проектів завдяки своїй конструкції та доступним можливостям налаштування.

У процесі вибору компонентів було визначено основні типи електроприводів для реалізації маніпулятора. До їх числа увійшли сервомотори SG90 і MG996R, а також крокові двигуни 28BYJ-48 і NEMA 17. Проведено порівняння їх характеристик і особливостей застосування, що дало змогу підібрати відповідні рішення для різних ступенів свободи маніпулятора. Крім цього, використання алюмінієвих профілів типу 2020 для створення базової конструкції забезпечило стабільність і гнучкість у налаштуванні.

Для забезпечення точності управління і зворотного зв'язку були обрані різні типи датчиків, серед яких енкодери, потенціометри, датчики сили, ультразвукові й інфрачервоні датчики. Їх інтеграція забезпечує не лише контроль положення й рухів маніпулятора, але й його адаптацію до різноманітних умов виконання задач.

Окрім того, обрано додаткові компоненти для живлення системи, які гарантують стабільну роботу як Arduino Uno R3, так і підключених сервомоторів і периферійних пристроїв. Підбір джерел живлення враховував потребу у стабільності напруги та можливість використання мобільних рішень, таких як акумулятори.

Таким чином, у розділі було систематизовано інформацію щодо комплектації, підбору електроприводів, датчиків та системи живлення, що створює основу для успішної реалізації системи керування роботизованим маніпулятором на базі Arduino Uno R3.

РОЗДІЛ 4

РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

4.1 Створення блок-схеми процесів

Створення блок-схеми процесів допомагає візуалізувати основні функції системи та їх взаємодію, що дозволяє краще розуміти структуру та логіку системи.

Візуалізація процесів є ключовим етапом, що допомагає визначити та проілюструвати всі основні компоненти системи та їх взаємодію (рис.4.1). Це дозволяє створити чітке уявлення про функціональність системи та забезпечити належне планування подальших етапів розробки.

Етапи візуалізації процесів можна розділити на наступні етапи:

Ідентифікація компонентів – контролери: центральний блок, який забезпечує управління всією системою; датчики: пристрої, що зчитують інформацію про зовнішнє середовище (наприклад, датчики позиції, тиску, температури); виконавчі механізми: пристрої, що виконують команди контролера (наприклад, серводвигуни).

Побудова схеми – компоненти: включення всіх основних компонентів на блок-схему; взаємодія: визначення зв'язків та взаємодії між компонентами, таких як обмін даними між контролером та датчиками, сигналами управління для виконавчих механізмів.

Визначення точок входу та виходу даних – точки входу: визначення точок, через які система отримує інформацію від зовнішнього середовища або від користувача; точки виходу: визначення точок, через які система передає команди виконавчим механізмам або інформацію користувачеві.



Рисунок 4.1 Блок схема загального алгоритму роботи

Визначення послідовності дій є наступним кроком після візуалізації процесів. Цей етап включає розробку послідовності операцій, які система має виконувати для досягнення поставлених завдань, а також визначення логіки управління та переходів між різними станами системи. Етапами наступного кроку є:

Аналіз завдань: Визначення конкретних завдань, які система повинна виконувати (наприклад, захоплення об'єкта, переміщення, позиціонування).

Планування операцій: Розробка детального плану виконання кожної операції з урахуванням необхідних етапів.

Наступним кроком є визначення логіки управління. Цей крок передбачає собою виконання наступних етапів: визначення умов, за яких повинні виконуватися певні дії (наприклад, якщо датчик фіксує об'єкт, активується захоплення); використання логічних операторів (if, else, while, for) для розробки

алгоритмів управління; визначення основних станів системи (наприклад, готовність до роботи, захоплення об'єкта, переміщення); визначення умов переходу між різними станами системи (рис.4.2).

Роглянемо приклад послідовності дій:



Рисунок 4.2 Блок схема алгоритм послідовності дій системи

Ідентифікація ключових компонентів є важливим етапом для визначення основних модулів системи, їх ролей та необхідних з'єднань між компонентами.

Етапи ідентифікації ключових компонентів включають в себе наступні кроки:

Визначення основних модулів системи – контролери: центральний блок управління, що обробляє сигнали з датчиків і передає команди виконавчим механізмам (наприклад, Arduino Uno R3); датчики: пристрої, що зчитують інформацію про зовнішнє середовище (наприклад, датчики положення, тиску, температури); виконавчі механізми: пристрої, що виконують команди контролера (наприклад, серводвигуни).

Визначення ролей компонентів – контролери: обробка вхідних даних, виконання алгоритмів керування, передача команд виконавчим механізмам; датчики: збір інформації про стан системи та зовнішнє середовище, передача даних на контролер; виконавчі механізми: виконання команд контролера для реалізації фізичних дій (захоплення, переміщення, позиціонування).

Визначення необхідних з'єднань та взаємодії між компонентами – електричні з'єднання: підключення датчиків та виконавчих механізмів до контролера за допомогою проводів та роз'ємів; сигнальні з'єднання: обмін сигналами між контролером та датчиками для отримання інформації про стан системи; команди управління: передача команд від контролера до виконавчих механізмів для виконання необхідних дій.

Приклад ідентифікації ключових компонентів:

Контролер – Arduino Uno R3: Виконує роль центрального блоку управління, обробляє сигнали з датчиків, виконує алгоритми керування, передає команди виконавчим механізмам.

Датчики – датчики положення: зчитують інформацію про поточне положення маніпулятора; датчики тиску: визначають силу захоплення об'єкта.

Виконавчі механізми – серводвигуни: виконують команди контролера для переміщення маніпулятора, забезпечують точне позиціонування.

Ці деталі дозволяють створити цілісне уявлення про структуру та логіку роботи системи керування роботом-маніпулятором, що забезпечує ефективне планування та реалізацію проекту.

Розглянемо більш детальний алгоритм роботи робото-маніпулятора зображений на рис.4.3.

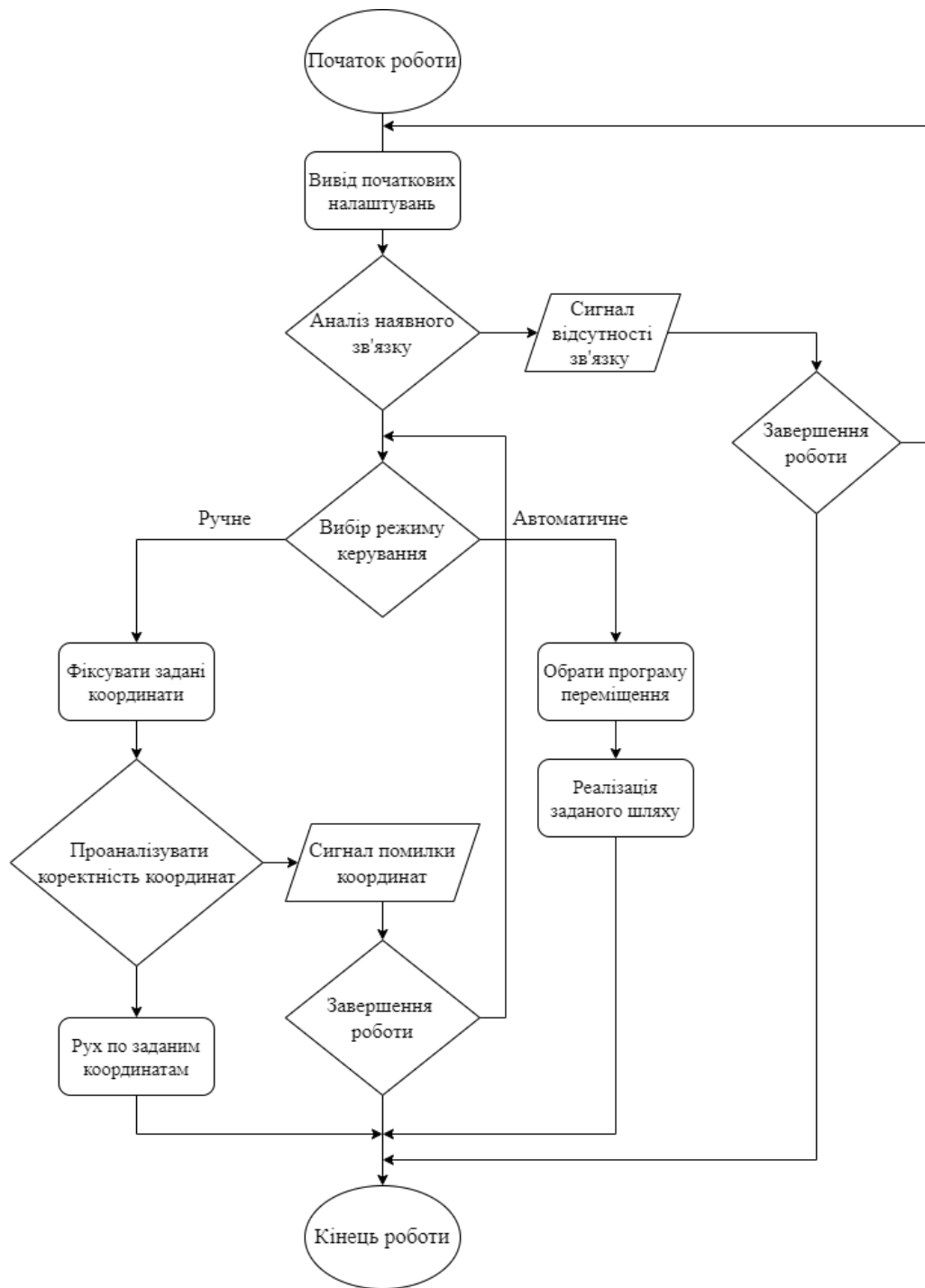


Рисунок 4.3 Детальний алгоритм роботи робота-маніпулятора

Алгоритм починається з ініціалізації системи, де виводяться початкові налаштування, необхідні для подальшої роботи. На цьому етапі система готується до взаємодії та перевіряє наявність з'єднання.

Аналіз наявності зв'язку

Перевіряється, чи є активне з'єднання з усіма необхідними компонентами. Якщо зв'язок встановлено, система переходить до вибору режиму керування. Якщо

зв'язок відсутній, виводиться повідомлення про помилку зв'язку, і система пропонує завершити роботу. Якщо користувач не бажає завершувати роботу, алгоритм намагається знову встановити зв'язок.

Вибір режиму керування. Система надає два варіанти роботи:

Ручний режим:

У ручному режимі оператор самостійно задає координати об'єкта або цільової точки для маніпулятора. Після цього координати фіксуються, і система перевіряє їх коректність.

Якщо координати правильні, маніпулятор здійснює рух до заданих точок. У разі некоректних координат система виводить повідомлення про помилку та пропонує виправити введені значення.

Після виконання руху або коригування координат система пропонує завершити роботу або продовжити введення нових координат.

Автоматичний режим:

У автоматичному режимі користувач обирає програму переміщення, яку система повинна виконати. Це може бути заздалегідь заданий шлях або послідовність дій.

Після вибору програми маніпулятор реалізує заданий шлях, виконуючи рух за заданою траєкторією.

Після завершення переміщення алгоритм запитує користувача, чи потрібно завершити роботу або повторити переміщення.

У будь-якому з режимів після виконання завдання (або на будь-якому етапі, якщо це вимагається) система пропонує завершити роботу. Якщо користувач вибирає завершення роботи, алгоритм завершує своє виконання, і система повертається до початкового стану.

Розробка та візуалізація алгоритмів для управління роботом-маніпулятором є критично важливими етапами в проектуванні складних автоматизованих систем.

Блок-схеми, що ілюструють процеси роботи системи, допомагають чітко уявити всі етапи її функціонування та взаємодію між основними компонентами. Це забезпечує кращу розумність і простоту в подальшій розробці, тестуванні та впровадженні системи.

У процесі створення таких схем визначаються основні компоненти системи, їхні ролі та необхідні з'єднання, що дозволяє отримати чітке уявлення про структуру і взаємодію різних елементів. Визначення точок входу та виходу даних, а також детальне планування операцій і визначення логіки управління дозволяють мінімізувати помилки на етапі впровадження і підвищити ефективність роботи маніпулятора.

Вибір між ручним і автоматичним режимами керування забезпечує гнучкість у роботі системи, даючи можливість оператору здійснювати точне управління або довірити виконання завдань алгоритму. Важливою складовою є також перевірка коректності введених даних, що дозволяє уникнути помилок в роботі системи.

Загалом, розробка і візуалізація алгоритмів є основою для ефективного проектування роботизованих систем, що дає змогу досягти точності, надійності та зручності в експлуатації маніпулятора в реальних умовах.

4.2 Розробка схеми системи керування на базі ARDUINO UNO3

Для початку роботи розглянемо плату керування ARDUINO UNO, а саме основні входи та виходи (рис. 4.4).

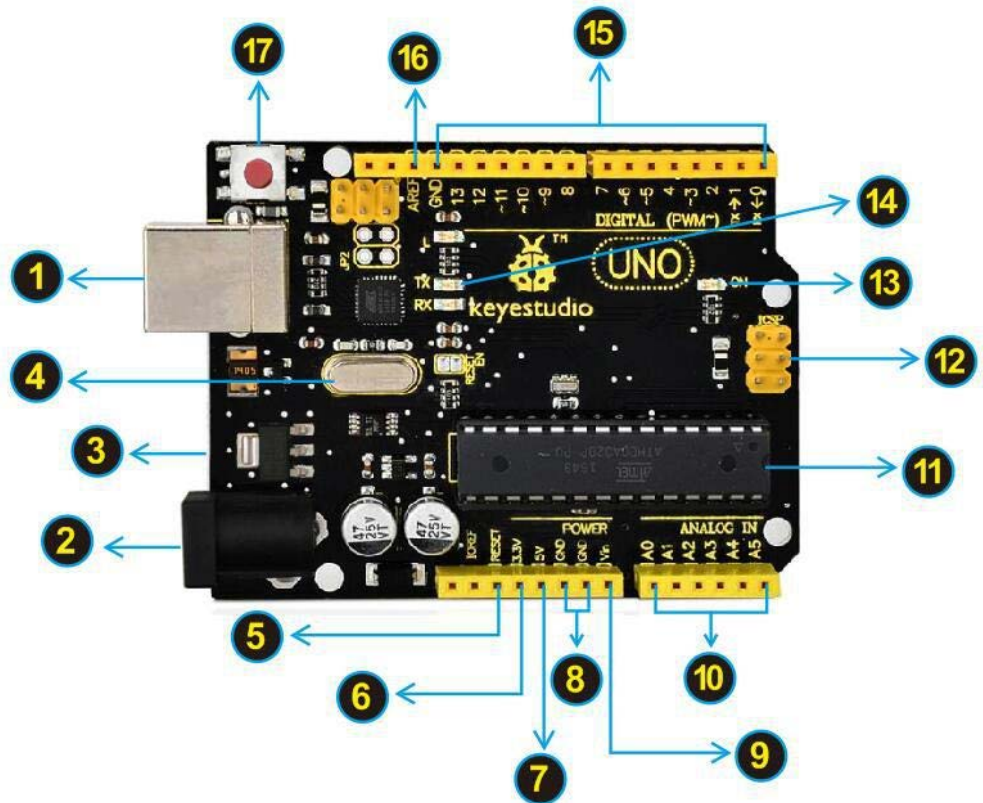


Рисунок 4.4 Плата керування ARDUINO UNO з позначеними входами та виходами

На рисунку 4.4 прийняті наступні позначення:

- 1) USB-підключення. Плату Arduino можна живити через USB-роз'єм. Або ви можете запрограмувати плату через USB-порт.
- 2) Гніздо живлення постійного струму. Плата Arduino може отримувати живлення від гнізда живлення постійного струму.
- 3) Регулятор напруги. Для контролю напруги, що подається на плату Arduino, а також для стабілізації напруги постійного струму, що використовується процесором та іншими компонентами.
- 4) Кристалічний генератор. Arduino обчислює час за допомогою кристалічного генератора. Число, надруковане на верхній частині кристала Arduino, — 16.000Н9Н. Він говорить нам, що частота становить 16 000 000 Герц або 16 МГц.

5, 17) Arduino RESET. Дає змогу скинути плату Arduino, наприклад, запустити програму з самого початку.

6, 7, 8, 9) Штифтовий роз'єм (3,3 В, 5 В, GND, Vin)

6) 3,3 В – забезпечує вихідну напругу 3,3 В

7) 5 В – забезпечує вихідну напругу 5 В

8) GND (Контакти заземлення) - два роз'єми GND на Arduino, кожен з яких можна використовувати для заземлення схеми.

9) Vin - Ви можете подавати зовнішнє живлення (наприклад, джерело змінного струму) через цей контакт до плати Arduino.

10) Аналогові контакти. Плата Arduino UNO має 6 аналогових входів, позначених від A0 до A5. Ці контакти можуть зчитувати сигнал від аналогових датчиків (таких як датчик вологості або датчик температури) і перетворювати його в цифрове значення, яке може читати мікроконтролер.

11) Мікроконтролер. Кожна плата Arduino має власний мікроконтролер.

12) ICSP (In-Circuit Serial Programming). У більшості випадків ICSP — це AVR, мікросхемний роз'єм Arduino, що складається з MOSI, MISO, SCK, RESET, VCC і GND.

13) Світлодіодний індикатор живлення. Під час живлення Arduino світлодіод горить означає, що друковану плату правильно увімкнено. Якщо світлодіодний індикатор не світиться, підключення неправильне.

14) Світлодіодний індикатор TX і RX. На борту можна знайти дві мітки: RX (отримання) і TX (передавання). Спочатку з'являються на цифрових контактах 0 і 1 для послідовного зв'язку; Крім того, світлодіод RX на платі буде блимати з різною швидкістю під час передачі послідовних даних. Швидкість спалаху залежить від швидкості передачі даних. Індикатор RX також блимає під час процесу отримання.

15) Цифровий вхід/вивід. Arduino UNO має 14 цифрових входів/виходів (з яких 6 можна використовувати як ШІМ-виходи). Ці контакти можуть бути налаштовані як контакти цифрового входу для зчитування логічного значення (0 або 1). Або використовується як цифровий вихідний штифт для управління різними модулями, такими як світлодіод, реле тощо. Штифт з позначкою « $\sim$ » можна використовувати для генерації ШІМ.

16) AREF. Опорна напруга (0-5 В) для аналогових входів. Використовується з analogReference.

Для можливості керуванням електродвигунами, сенсорами та пристроями вводу та виводу використовується плата розширення двигуна/сервоприводу keyestudio TB6612FNG(рис.4.5). Ця плата є багатофункціональним модулем, який використовується для управління електродвигунами та сенсорами в автоматизованих системах. Вона призначена для інтеграції з іншими платами, такими як Arduino, і дозволяє легко підключати периферійні пристрої, такі як джойстики, сенсори, двигуни та Bluetooth-модулі[13].

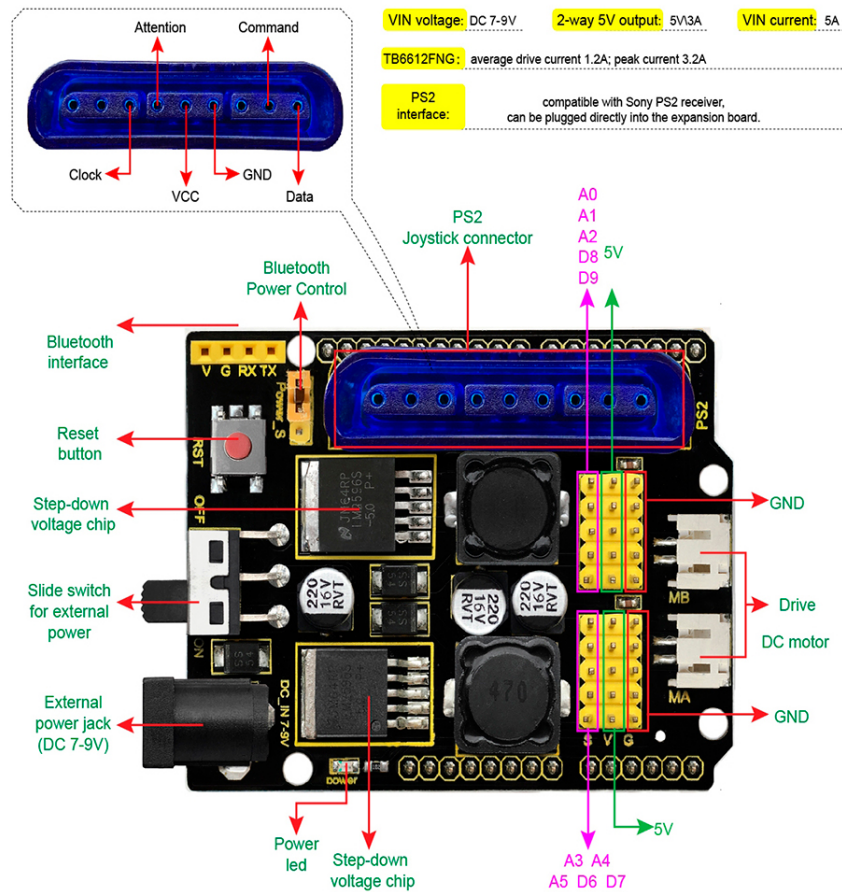


Рисунок 4.5 Плата розширення двигуна/сервоприводу keyestudio
TB6612FNG

Плата легко інтегрується з Arduino по принципу накладання одне на одного підключаючись у відповідні піни на основній платі.

Основні компоненти що зображені на рис. 4.5 мають на наступні значення:

PS2-роз'єм (Joystick connector): використовується для підключення контролера PS2 (джойстика) до плати.

Підтримує передачу сигналів для управління рухом, кнопками та іншими функціями.

Bluetooth інтерфейс: використовується для бездротового з'єднання з іншими пристроями, наприклад, смартфонами або комп'ютерами. Реалізує передачу команд та отримання даних.

Роз'єм для підключення двигунів: МА та МВ: виходи для керування двигуном постійного струму. Забезпечує подачу живлення та сигналів управління для двигуна.

Штекер для зовнішнього живлення: підтримує підключення зовнішнього джерела живлення в діапазоні 7-9V. Використовується для забезпечення стабільної роботи плати та підключених пристроїв.

Step-down перетворювач напруги: знижує вхідну напругу до рівня, необхідного для роботи внутрішніх компонентів та виходів (5V). Запобігає перевантаженню плати.

Слайдер (перемикач живлення): вмикає та вимикає живлення плати. Може використовуватися для перевірки роботи системи.

Кнопка "Reset": Використовується для перезапуску плати. Корисно при відладці або налаштуванні системи.

Індикатори живлення: Показують наявність живлення та статус роботи плати.

Аналогові та цифрові контакти (A0-A5, D8-D7): аналогові входи (A0-A5): використовуються для підключення сенсорів і пристроїв, що передають аналогові сигнали (наприклад, датчики температури).

Цифрові входи/виходи (D8-D7): для обробки цифрових сигналів, підключення кнопок, світлодіодів чи інших цифрових компонентів.

Підключення до плати серводвигунів наведено в таблиці 4.1.

Таблиця 4.1 Підключення до плати серводвигунів

Ім'я	IO Pin
Серво 1 (базова плата)	A1
Серво 2 (права сторона)	A0

Серво 3 (ліва сторона)	A6
Сервопривід 4 (затискач)	A9
Правий джойстик X	A2
Правий джойстик Y	A5
Правий джойстик Z (клавіша)	A7
Лівий джойстик X	A3
Лівий джойстик Y	A4
Лівий джойстик Z	A8
D1/DAT PS2	12
D0/CMD PS2	11
CE/SEL PS2	10
CLK PS2	13

Загальний вигляд підключених модулів наведено на рис.4.6, схема включає в себе наступні елементи:

Блок живлення: підключений до плати для забезпечення живленням всієї системи. Це джерело живлення, що забезпечує стабільну напругу для всіх компонентів.

Bluetooth-модуль: підключений до основної плати для бездротового зв'язку, для управління системою через Bluetooth з мобільного пристрою або іншого зовнішнього контролера. Підключення здійснюється через пін (Rx, Tx), що відповідає за передачу даних.

Сервоприводи (4 шт.): підключені до основної плати для керування різними рухами та механізмами робото-маніпулятора. Підключення здійснюється через спеціальні пін-контакти на платі, які передають керуючі сигнали до сервоприводів.

Пристрої вводу – джойстики (ліворуч і праворуч): підключені до плати для збору даних або моніторингу певних параметрів, таких як відстань, напрямок руху або інші показники. Вони використовуються для додаткових функцій пристрою, наприклад, для управління рухом у просторі чи збору інформації.

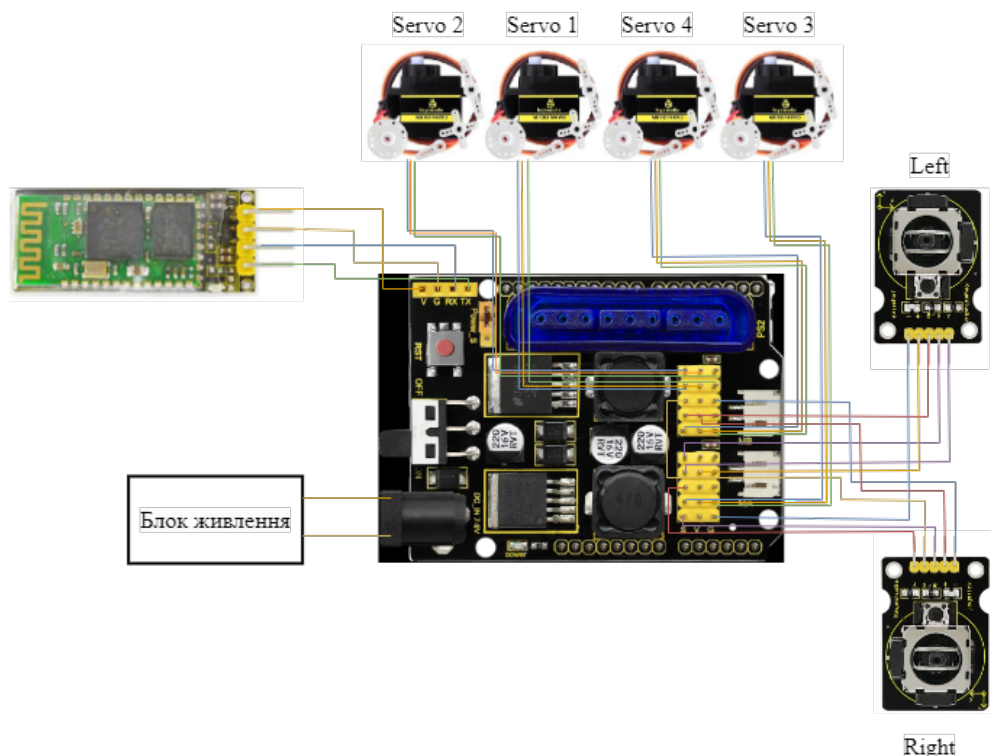


Рисунок 4.6 Загальний вигляд підключення елементів керування

4.3 Розробка програми системи керування роботом маніпулятором

```
#include <Servo.h> // Підключення бібліотеки для серво-моторів
```

```
// Створення об'єктів для серво-моторів
```

```
Servo baseServo; // Серво для основи маніпулятора
```

```
Servo shoulderServo; // Серво для плеча
```

```
Servo elbowServo; // Серво для передпліччя

Servo gripperServo; // Серво для захвату

// Піни для підключення серво-моторів

const int basePin = 9; // Пін для серво основи

const int shoulderPin = 10; // Пін для серво плеча

const int elbowPin = 11; // Пін для серво передпліччя

const int gripperPin = 12; // Пін для серво захвату

void setup() {

// Ініціалізація серво-моторів

baseServo.attach(basePin); // Підключення серво до піну 9

shoulderServo.attach(shoulderPin); // Підключення серво до піну 10

elbowServo.attach(elbowPin); // Підключення серво до піну 11

grripperServo.attach(gripperPin); // Підключення серво до піну 12

// Ініціалізація серво-моторів на початкові кути

baseServo.write(90); // Початкове положення для основи (90°)

shoulderServo.write(90); // Початкове положення для плеча (90°)

elbowServo.write(90); // Початкове положення для передпліччя (90°)

grripperServo.write(0); // Початкове положення для захвату (закритий)

// Затримка, щоб маніпулятор встиг встати на початкову позицію

delay(1000);

// Ініціалізація серійного порту для відладки

Serial.begin(9600);

Serial.println("Маніпулятор ініціалізовано.");
```

```
void loop() {  
  
  // Приклад простого руху маніпулятора:  
  
  // Рух основи (поворот на 45 градусів вправо)  
  
  baseServo.write(135); // 135° вправо  
  
  delay(1000);  
  
  // Рух плеча (підйом на 30 градусів)  
  
  shoulderServo.write(120); // 120° для підйому  
  
  delay(1000);  
  
  // Рух передпліччя (поворот на 45 градусів)  
  
  elbowServo.write(135); // 135° для розтягування  
  
  delay(1000);  
  
  // Відкриття захвату  
  
  gripperServo.write(90); // Відкритий захват  
  
  delay(1000);  
  
  // Закриття захвату  
  
  gripperServo.write(0); // Закритий захват  
  
  delay(1000);  
  
  // Повернення до початкових положень  
  
  baseServo.write(90); // Повернення на початкове положення  
  
  shoulderServo.write(90); // Повернення на початкове положення  
  
  elbowServo.write(90); // Повернення на початкове положення  
  
  delay(1000); // Затримка для коректної роботи
```

#include <Servo.h> Servo myservo; // створити сервооб'єкт для керування сервоприводом

void setup() {Serial.begin(9600);

delay(1000);}

void loop() {myservo.attach(A0); // зміна штифту для налаштування

myservo.write(0); // значення кута

delay(1000);}

#include <Servo.h>

Servo myservo1; // створити сервооб'єкт для керування сервоприводом

Servo myservo2;

Servo myservo3;

Servo myservo4;

int pos1=80, pos2=60, pos3=130, pos4=0;

void setup() {

myservo1.attach(A1); // прикріплює сервопривід на контакті 9 до об'єкта сервоприводу

myservo2.attach(A0);

myservo3.attach(6);

myservo4.attach(9);

myservo1.write(pos1);

delay(1000);

myservo2.write(pos2);

myservo3.write(pos3);

myservo4.write(pos4);

```
delay(1500);

}

void loop() {

  // поворот праворуч для (pos1;pos1>0;pos1--) {

  myservo1.write(pos1);

  delay(5); // затримка 5 мс (використовується для регулювання швидкості
сервоприводу)

  }

  delay(1000); // відкрити кіготь

  for(pos4;pos4<100;pos4++) {

  myservo4.write(pos4);

  }

  delay(1000);

  // правий сервопривід повертається на 100 градусів for(pos2;pos2>50;pos2--

  myservo2.write(pos2);

  delay(5);

  }

  // лівий сервопривід повертається на 5 градусів протягом
(pos3;pos3>50;pos3--)

  {

  myservo3.write(pos3);

  delay(5);

  }
```

```
delay(1500); // закрити кіготь

for(pos4;pos4>0;pos4--) {

myservo4.write(pos4);

}

delay(1000);

// лівий сервопривід обертається на 100 градусів, коромисло піднімається.

for(pos3;pos3<120;pos3++) {

myservo3.write(pos3);

delay(5);

}

delay(1000); // поворот ліворуч

for(pos1;pos1<180;pos1++) {

myservo1.write(pos1);

delay(5);

}

delay(1000); // Опустіть руку

for(pos3;pos3>50;pos3--) {

myservo3.write(pos3);

delay(5);

}

delay(1000); // відкрити кіготь

for(pos4;pos4<100;pos4++) {

myservo4.write(pos4);
```

```

}

delay(1000); // підняти руку

for(pos3;pos3<120;pos3++) {

myservo3.write(pos3);

delay(5);

}

delay(1000);

// замкнути кіготь

for(pos4;pos4>0;pos4--) {

myservo4.write(pos4);

}

delay(1000);

}

const int right_X = A2; // визначте праву шпильку X для A2

const int right_Y = A5; // визначте правий штифт Y для A5

const int right_key = 7; // визначте правий ключ на 7 (це значення Z)

const int left_X = A3; // визначте ліву шпильку X для A3

const int left_Y = A4; // визначте ліву шпильку Y для A4

const int left_key = 8; // визначте лівий ключ на 8 (це значення Z)

void setup() {

pinMode(right_key, INPUT); // встановіть праву/ліву клавішу на ВХІД

pinMode(left_key, INPUT);

Serial.begin(9600); // встановіть швидкість передачі даних на 9600 бод

```

```

}

void loop() {

    int x1,y1,z1; // визначте змінну, яка використовується для збереження
значення джойстика, яке вона читає

    int x2,y2,z2;

    x1 = analogRead(right_X); // прочитайте значення X справа
    y1 = analogRead(right_Y); // прочитайте значення правого Y
    z1 = digitalRead(right_key); /// прочитайте значення правого Z

    x2 = analogRead(left_X); // прочитайте значення лівого X
    y2 = analogRead(left_Y); // прочитайте значення лівого Y
    z2 = digitalRead(left_key); // прочитайте значення лівого Z

    Serial.print("right_X = "); // на послідовному моніторі роздрукуйте right_X
    Serial.println(x1 ,DEC); // роздрукуйте значення правого X і перенос рядків
    Serial.print("right_Y = ");

    Serial.println(y1 ,DEC); //Serial.print("right_key = "); //Serial.println(z1 ,DEC);

    Serial.println(x2 ,DEC);

    Serial.print("left_Y = ");

    Serial.println(y2 ,DEC);

    Serial.print("left_key=");

    Serial.println(z2,DEC);

    #include <Servo.h> // додати бібліотеки сервосистем

    Servo myservo1; // створити сервооб'єкт для керування сервоприводом
    Servo myservo2;

```

Servo myservo3;

Servo myservo4;

int pos1=80, pos2=60, pos3=130, pos4=0; // визначте змінну 4 кута сервоприводу та призначте початкове значення (тобто значення кута позиції завантаження)

const int right_X = A2; // визначте праву шпильку X для A2

const int right_Y = A5; // визначте правий штифт Y для A5

const int right_key = 7; // визначте штифт правої клавіші 7(це значення Z)

const int left_X = A3; // визначте ліву шпильку X для A3

const int left_Y = A4; // визначте ліву шпильку X для A4

const int left_key = 8; // визначте лівий ключ на 8 (це значення Z)

int x1,y1,z1; // визначте змінну, яка використовується для збереження прочитаного значення джойстика.

int x2,y2,z2;

void setup() {

// поза черевики

myservo1.write(pos1);

delay(1000);

myservo2.write(pos2);

myservo3.write(pos3);

myservo4.write(pos4);

delay(1500);

pinMode(right_key, INPUT); // встановить праву/ліву клавішу на ВХІД

pinMode(left_key, INPUT);

```

Serial.begin(9600); // встановить швидкість передачі даних на 9600 бод

}

void loop() {

myservo1.attach(A1); // встановить штифт керування сервоприводу 1 на A1
myservo2.attach(A0); // встановить штифт керування сервоприводу 2 на A0
myservo3.attach(6); // встановить штифт керування сервоприводу 3 на D6
myservo4.attach(9); // встановить штифт керування сервоприводу 4 на D9

x1 = analogRead(right_X); // читайте правильне значення X
y1 = analogRead(right_Y); // читайте правильне значення Y
z1 = digitalRead(right_key); /// читайте правильне значення Z

x2 = analogRead(left_X); // читайте ліве значення X
y2 = analogRead(left_Y); // читайте ліве значення Y
z2 = digitalRead(left_key); // читайте ліве значення Z

//delay(5); // знизити загальну швидкість

zhuazi(); // обертати

zhuandong(); // плече

xiaobi();

void zhuazi() {

//claw

if(x2<50) // якщо натиснути лівий джойстик праворуч {

pos4=pos4-2; // поточний кут сервоприводу 4 віднімає 2 (змінить значення,
яке ви віднімаєте, таким чином змінить закриту швидкість кігтя)

//Serial.println(pos4);

```

myservo4.write(pos4); // сервопривід 4 керує дією, кіготь поступово закривається.

delay(5);

if(pos4<2) // якщо значення pos4 віднімається до 2, кіготь під кутом 37 градусів, який ми перевірили, закритий.{

// (має змінити значення на основі факту)

pos4=2; // зупинити віднімання, коли зменшити до 2

}

}

if(x2>1000) //// якщо натиснути лівий джойстик ліворуч {

pos4=pos4+8; // поточний кут сервоприводу 4 плюс 8 (змінить значення, яке ви плюсуєте, таким чином змініть швидкість відкриття кігтя)

//Serial.println(pos4);

myservo4.write(pos4); // сервопривід 4 управляє рухом, кіготь поступово відкривається.

delay(5); if(pos4>108) // обмеження найбільшого кута при відкритому кігті

pos4=108;

if(x1<50) // якщо натиснути правий джойстик праворуч {

pos1=pos1-1; // pos1 віднімає 1

myservo1.write(pos1); // сервопривід 1 керує рухом, рука повертається праворуч.

delay(5);

if(pos1<1) // обмеження кута при повороті праворуч {

pos1=1;

```

}

if(x1>1000) // якщо натиснути правий джойстик до кінця {

pos1=pos1+1; // pos1 плюс 1

myservo1.write(pos1); // рука повертається вліво

delay(5); if(pos1>180) // обмеження кута при повороті ліворуч {

pos1=180;

}

// надпліччя пуста хіаобі () {

if(y1>1000) // якщо натиснути правий джойстик угору{

pos2=pos2-1;

myservo2.write(pos2); // плече підніметься

delay(5);

if(pos2<0) // обмеження кута підйому {

pos2=0;

}

}

if(y1<50) // якщо натиснути правий джойстик вниз {

pos2=pos2+1;

myservo2.write(pos2); // плече опуститься вниз

delay(5);

if(pos2>180) // обмежте кут при опусканні {

pos2=180;

}

```

```

// нижня рука

void dabi() {

  if(y2<50) // якщо натиснути лівий джойстик вгору

  {

    pos3=pos3+1;

    myservo3.write(pos3); // нижня частина руки витягнеться

    delay(5);

    if(pos3>180) // обмежити розтягнутий кут {

      pos3=180;

    }

  }

  if(y2>1000) // якщо натиснути лівий джойстик вниз {

    pos3=pos3-1;

    myservo3.write(pos3); // нижня частина руки буде відведена назад

    delay(5);

    if(pos3<35) // обмежити кут відведення {

      pos3=35;

    }

  }

  #include <Servo.h> // додати серво бібліотеки Servo

  myservo1; // створити сервооб'єкт для керування сервоприводом

  Servo myservo2;

  Servo myservo3;

  Servo myservo4;

```

int pos1=80, pos2=60, pos3=130, pos4=0; // визначте змінну 4 кута сервоприводу та призначте початкове значення (це значення кута позиції завантаження)

const int right_X = A2; // визначте праву шпильку X для A2

const int right_Y = A5; // визначте правий штифт Y для A3

const int right_key = 7; // визначте правий ключ на 7 (це значення Z)

const int left_X = A3; // визначити лівий X-контакт для A3

const int left_Y = A4; // визначаємо лівий штифт Y для A4

const int left_key = 8; // визначити ліву клавішу на 8 (це значення Z)

int x1,y1,z1; // визначаємо змінну, яка використовується для збереження значення джойстика.

int x2,y2,z2;

int s1,s2,s3,s4;

void setup() {

// положення завантаження

myservo1.write(pos1);

delay(1000);

myservo2.write(pos2);

myservo3.write(pos3);

myservo4.write(pos4);

delay(1500);

pinMode(right_key, INPUT); // встановить праву/ліву клавішу на ВХІД

pinMode(ліва_клавіша, ВХІД);

Serial.begin(9600); // встановити швидкість передачі даних на 9600 бод}

```

void loop() {

myservo1.attach(A1); // встановить штифт керування сервоприводу 1 на A1

myservo2.attach(A0); // встановити контрольний штифт серво 2 на A0

myservo3.attach(6); //встановить контрольний штифт сервоприводу 3 на D6

myservo4.attach(9); //встановить контрольний штифт серво 4 на D9

x1 = analogRead(right_X); // читати правильне значення X

y1 = analogRead(right_Y); // читаємо правильне значення Y

z1 = digitalRead(права_клавіша); /// прочитати значення Z правого ключа

x2 = analogRead(left_X); // читання лівого значення X

y2 = analogRead(left_Y); //читання лівого значення Y

z2 = digitalRead(ліва_клавіша); //читання значення Z лівого ключа

//delay(5); // зменшити загальну швидкість

if(z1==1) // якщо натиснуто праву клавішу джойстика {

затримка (10); // затримка для усунення тремтіння

if(z1==1) // знову оцінити, чи натиснуто праву клавішу {

s1=myservo1.read(); // читання значення кута кожного сервоприводу

s2=myservo2.read();

s3=myservo3.read();

s4=myservo4.read();

if(z2==1) // якщо натиснути ліву клавішу {

delay(10);

if(z2==1) {

```

```
pos1=myservo1.read(); // запис значення кута 4 сервоприводів у поточному  
положенні  
  
pos2=myservo2.read();  
  
pos3=myservo3.read();  
  
pos4=myservo4.read();  
  
if(pos1<s1) // якщо кут сервоприводу 1 менший за значення змінної s1 {  
  
    while(pos1<s1) //цикл while, повертає сервопривід до позиції значення, що  
зберігається в масиві. {  
  
        myservo1.write(pos1); // сервопривід 1 керує рухом  
  
        pos1++; //pos1 плюс 1 затримка(5); // затримка на 5 мс, контроль швидкості  
обертання сервоприводу.}  
  
    }  
  
    else // якщо кут сервоприводу 1 більший за значення, збережене в масиві 1.  
  
        while(pos1>s1) //цикл while, повертає сервопривід до позиції значення, що  
зберігається в масиві. {  
  
            myservo1.write(pos1); // серво 1 управляє рухом  
  
            pos1--; //pos1 віднімає 1  
  
            delay(5); // затримка на 5 мс, контролюючи швидкість обертання  
сервоприводу.  
  
            // пояснення таке ж, як у серво 1 if(pos2<s2) {  
  
            while(pos2<s2) {  
  
                myservo2.write(pos2);  
  
                pos2++;  
  
                delay(5);
```

```
else

while(pos2>s2) {

myservo2.write(pos2);

pos2--;

delay(5);

// пояснення таке ж, як у серво 1 if(pos3<s3){

while(pos3<s3) {

myservo3.write(pos3);

pos3++;

delay(5);

else {

while(pos3>s3) {

myservo3.write(pos3);

pos3--;

delay(5);

if(pos4<s4) {

while(pos4<s4) {

myservo4.write(pos4);

pos4++;

delay(5);

else {

while(pos4>s4) {

myservo4.write(pos4);
```

```

pos4--;

delay(5);

//кіготь void zhuazi() {

//кіготь if(x2<50) // якщо натиснути лівий джойстик праворуч {

pos4=pos4-2; // поточний кут сервоприводу 4 віднімає 2 (змінить значення,
яке ви віднімаєте, таким чином змінить закриття швидкість кігтя)

//Serial.println(pos4);

myservo4.write(pos4); // сервопривід 4 керує дією, кіготь поступово
закривається

delay(5);

if(pos4<2) // якщо значення pos4 віднімається до 2, кіготь під кутом 37
градусів, який ми перевірили, закритий.) {

// (має змінити значення на основі факту)

pos4=2; // зупинити віднімання, коли зменшити до 2

if(x2>1000) //// якщо натиснути лівий джойстик ліворуч {

pos4=pos4+8; // поточний кут сервоприводу 4 плюс 8 (змінить значення
плюса, таким чином змінить швидкість відкриття кігтя)

//Serial.println(pos4);

myservo4.write(pos4); // сервопривід 4 управляє рухом, кіготь поступово
відкривається.

delay(5);

if(pos4>90) // обмежують найбільший кут у відкритому стані {

pos4=90;

void zhuandong() {

```

```
if(x1<50) // якщо натиснути правий джойстик праворуч {  
  
    pos1=pos1-1; //pos1 віднімає 1  
  
    myservo1.write(pos1); // сервопривід 1 керує рухом, рука повертається  
    праворуч.  
  
    delay(5);  
  
    if(pos1<1) // limit the angle when turn right {  
  
        pos1=1;
```

Результати роботи робота-маніпулятора представлено на циклограмі зображеного на рис. 4.7. Ця циклограма ілюструє взаємодію між усіма сервоприводами протягом робочого циклу маніпулятора. Основний цикл включає в себе:

Основа (Base Servo): зміна положення з 90° до 0° , потім до 180° і назад.

Плече (Arm Servo): рухи плеча між 45° і 135° .

Захват (Grip Servo): відкриття та закриття.

Кисть (Wrist Servo): повороти кисті на 45° та 135° .

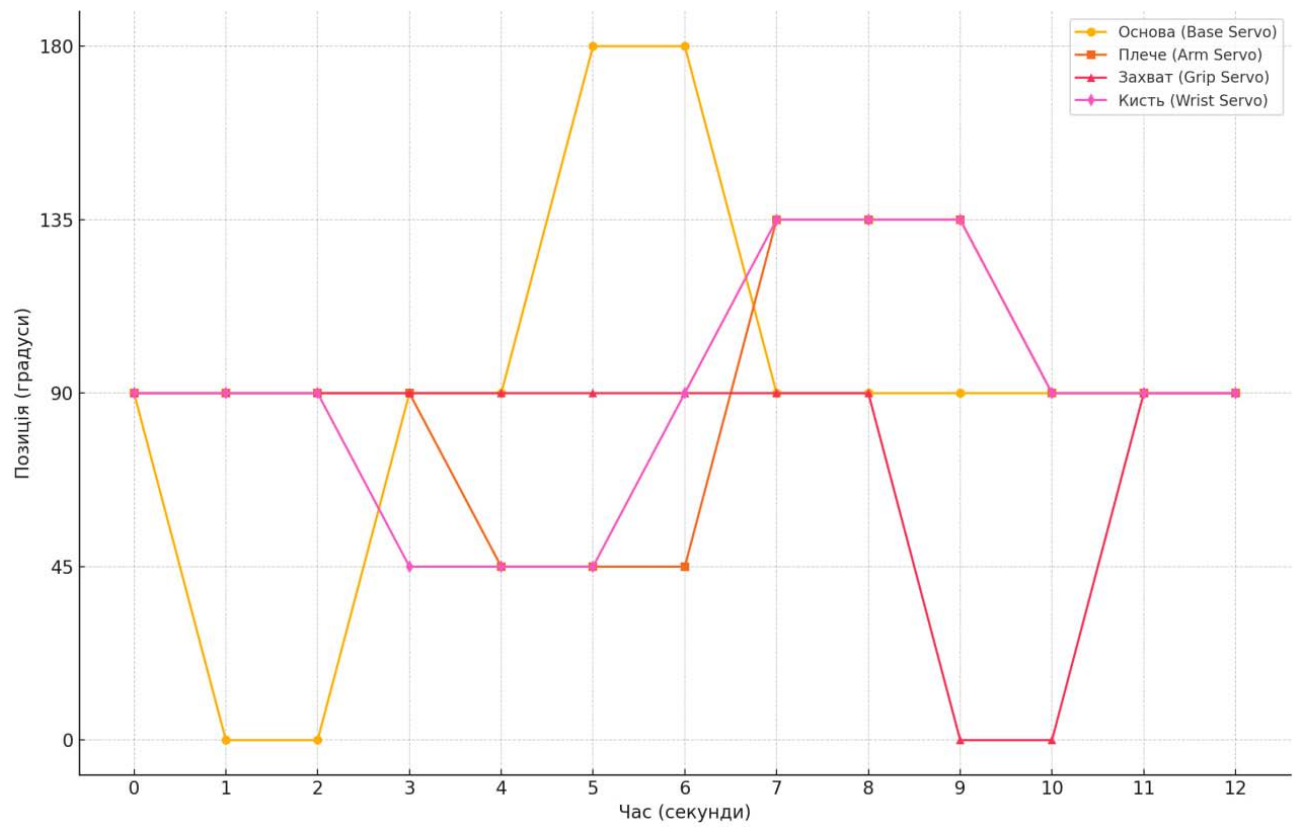


Рисунок 4.7 Циклограма роботи маніпулятора

Висновки за розділом

У даному розділі було детально розглянуто процес розробки програмного забезпечення для системи керування роботом-маніпулятором на базі Arduino UNO. Початковий етап розробки передбачав створення блок-схеми процесів, що допомогло чітко візуалізувати основні компоненти системи та їх взаємодію. Це дозволило краще зрозуміти структуру та логіку роботи всієї системи. Створення блок-схем, включаючи визначення точок входу та виходу даних, взаємодії компонентів і алгоритмів керування, забезпечило точність у проектуванні.

Наступним важливим етапом стало детальне планування послідовності дій і визначення логіки управління системою. Зокрема, розглянуті етапи ідентифікації компонентів, серед яких Arduino UNO, датчики, виконавчі механізми та плату розширення двигуна/сервоприводу keyestudio TB6612FNG. Ідентифікація компонентів дозволила визначити їх ролі, з'єднання та взаємодію, що сприяло належному проектуванню схем керування і забезпеченню стабільної роботи всієї системи.

Розробка схеми системи керування на основі Arduino UNO включала використання таких компонентів, як джойстики, серводвигуни і Bluetooth-модуль для бездротового управління. Це дозволило реалізувати як ручний, так і автоматичний режими керування роботом-маніпулятором, що значно підвищило гнучкість і зручність експлуатації системи.

Програмний код, розроблений для керування сервоприводами, забезпечує точне позиціонування маніпулятора. Підключення до Arduino різних компонентів і налаштування їх через сервооб'єкти дозволяє керувати рухами маніпулятора з великою точністю. Код забезпечує ініціалізацію сервоприводів, їх налаштування та виконання заданих операцій, таких як переміщення в задані позиції. Отримано циклограми роботи, які показують взаємодію між усіма сервоприводами протягом робочого циклу маніпулятора.

Таким чином, цей розділ демонструє важливість комплексного підходу до розробки програмного забезпечення для систем автоматизації. Візуалізація процесів, точне визначення взаємодії компонентів, а також розробка логіки управління допомогли створити ефективну і надійну систему керування роботоманіпулятором. Ці етапи є основою для подальшого розвитку і вдосконалення автоматизованих систем в контексті робототехніки та промислового виробництва.

РОЗДІЛ 5

ОХОРОНА ПРАЦІ

5.1 Безпека використання роботів

Будь-який робот, особливо промисловий, є джерелом підвищеної небезпеки. Великі промислові роботи-навантажувачі або ж роботи- маніпулятори можуть з легкістю вбити людину, яка по своїй необережності виявився на шляху у робота.

Основна проблема полягає у обмеженому інтелекті робота. Хоча робота можна запрограмувати виконувати будь-яку дію, встановити на нього різні датчики, які були б здатні запобігти нещасним випадкам, самі по собі роботи не мають власним інтелектом і здебільшого є бездумні машини, які прямо слідують запрограмованим протоколам.

Робот може виконувати свою програму, не усвідомлюючи, що стався збій і в даний момент робот влаштовує погром, знищуючи продукцію джерелом небезпеки для життя та здоров'я працівників. Як результат, що саме головне, можливі людські жертви. Побічними втратами є пошкодження або знищення готової продукції та сировини, що так само зазнає серйозних матеріальних збитків.

Особливу небезпеку становлять у цій ситуації рухливі роботи, здатні здійснювати переміщення у всіх напрямках. Навіть якщо такий робот має чітко запрограмовану траєкторію руху, не можна забувати, що з різних збоїв, або ж зносу/ушкодження рухомої частини може призвести до різкої зміни курсу руху робота.

Також проблема полягає в тому, що після закінчення робочого циклу та повернення робота на вихідну позицію для початку повторного циклу, або ж при переході робота в іншу зону для виконання іншого циклу, траєкторія руху робота так само може бути неконтрольованою.

До того ж, можливі ситуації, коли робот перебуває в нерухомому стані, але не є вимкненим. Система чекає, коли буде подано вхідний сигнал для початку роботи. І перебування в неподалік такого робота може спричинити несприятливого аваріального стану внаслідок раптового аваріального робота. Те саме можливе при екстреному відключенні робота, коли він при включенні продовжує виконувати свій робочий цикл. Особливу небезпеку становлять ситуації, коли через деякі технічних обставин працівник змушений увійти до небезпечної зони дії робота, для ремонту або калібрування робота. Причому в рідкісних випадках виникають ситуації, коли робота неможливо відключити.

З метою підвищення безпеки під час експлуатації роботів, проводиться низка заходів, спрямованих на усунення ризиків аваріального стану. Одним з найпоширеніших і відносно ефективним є встановлення огорож у зоні дії робота.

Дані огороження являють собою стінки або паркани, захищають роботів від працівників. Вхід до цієї зони представляє собою двері, що закриваються на ключ, що запобігає попаданню людей в зону дії робочого робота.

Другим способом є встановлення блокаторів. Натискання на кнопку, що відповідає за блокування, примусово припиняє роботу робота. Після повторного натискання блокування знімається. Обов'язковим є встановлення кнопок аварійної зупинки.

Третій спосіб полягає в обмеженні зони дії робота. Робиться цю двома основними методами: програмуванням чи установки будь-яких фізичних упорів, що обмежують рухливість робота.

Деяку користь приносить так само правильне розміщення обладнання.

Як правило, на заводах є якісь верстати, безпосередньо до якого надалі додаються роботи. Суть правильного розміщення полягає у мінімізації ризиків для оператора за допомогою створення максимальної відстані між роботом та оператором. У такому разі робот розміщується на максимальній відстані від огорож.

Другим, але обов'язковим засобом підвищення безпеки, є установка попереджувальних знаків і звукової або світлової індикації, які, як правило, сигналізують людині про те, що робот не вимкнений і все ще становить зону підвищеної небезпеки.

Підвищення надійності роботів сприяє зниженню збоїв. Різкий стрибок напруги або перегрів здатний вивести робота з ладу. Тому питання пилу та вологозахисту є першорядними. При експлуатації робота в умовах підвищеної температури слід встановити охолоджувальну установку.

Підвищення безпеки можна досягти навіть на стадії проектування робота. Видалення всіх гострих граней та встановлення м'якої обшивки здатна значно знизити небезпеку робота.

До всього вищесказаного, інструктаж працівників підприємства є важливим пунктом підвищення безпеки праці. Періодичне проведення лекцій з безпеки, на яких робочому персоналу доступно пояснюють правила техніки безпеки та неприпустимість їх порушення, здатне зменшити кількість нещасних випадків, що сталися через недбальство персоналу [14].

5.2 Захист від дотику до об'єктів під напругою

Небезпека ураження струмом при дотику до корпусу та інших струмопровідним металевим частинам електрообладнання, що виявилися під напругою внаслідок замикання на корпус та з інших причин, може бути усунена швидким вимкненням пошкодженої електроустановки від мережі живлення, а також зниженням напруги корпусу щодо землі.

Повинна дотримуватися електробезпеки при проведенні будівельно-монтажних робіт. Будівельні майданчики та приміщення на них належать до об'єктів підвищеної чи особливої небезпеки.

Робітники, які мають із застосуванням електричного струму, у тому числі які користуються електрифікованим інструментом, повинні працювати в гумових рукавичках та калошах та на гумових килимках.

Вмикати електроустановки може лише спеціально навчений робітник.

Корпуси електроустаткування повинні бути надійно заземлені. У приміщеннях з підвищеною вологістю, а також у котлах, баках та поза приміщеннями напруга має бути не вище 36 В. звуку в дБА, використовується для орієнтовної оцінки постійного та непостійного шуму, тому що в цьому випадку ми не знаємо спектру шуму.

Ураження електричним струмом відбувається, коли людський організм вступає у контакт із джерелом напруги. Торкнувшись провідника, який перебуває під напругою, людина стає частиною електромережі, якою починає протікати електричний струм. Як відомо, організм людини складається з великої кількості солей і рідини, що є хорошим провідником електрики, тому Вплив електричного струму на організм людини може бути летальним.

Наслідки, що виникнуть внаслідок дії електричного струму на людину залежать від багатьох факторів, а саме: від величини і роду електричного струму, – змінний струм є більш небезпечним, ніж постійний; тривалості його впливу, – чим більше час дії струму на людину, тим важчі наслідки; шляхи протікання, – найбільшу небезпеку становить струм, що протікає через головний і спинний мозок, область серця та органів дихання (легкі); від фізичного та психологічного стану людини. Організм людини має певний опір, цей опір варіюється в залежності від стану людини. Мінімальна величина струму, яку здатний відчувати людський організм складає 1 мА. При підвищенні струму понад 1 мА людина починає почуватися некомфортно, виникають болючі скорочення м'язів, зі збільшенням струму до 12-15 мА виникає судомне скорочення м'язів, контролювати свою м'язову систему людина вже не в стані та власними силами не може розірвати контакт з джерелом струму.

Дія електричного струму понад 25 мА призводить до паралічу м'язів органів дихання в результаті чого людина може просто задихнутися. При подальшому збільшенні струму виникає фібриляція серця. Електричний струм проходячи через організм людини може чинити їй три види впливів: термічне; електролітичне; біологічне.

Тіло людини за рахунок наявності великої кількості рідини з солями і металами, що містяться в її складі, є провідником електричного струму, вплив якого може призвести до летального наслідку для людини. Оскільки джерелом живлення роботів є електрика, кожен робот може нести потенційну небезпеку.

Джерелами живлення є батарейки, акумулятори або пряме підключення робота до електромережі за наявності відповідного зарядного кабелю з перетворювачем непостійного електричного струму у постійний.

Вражаюча дія електричним струмом відбувається, коли людина без засобів захисту торкається об'єкта під напругою. Торкнувшись здатного проводити електрику об'єкта, до яких відносяться різні металеві об'єкти, а також солевмісті рідини, людина, тіло якої так само є провідником, відразу починає пропускати крізь себе електричний струм. Травматизм від впливу електричного струму безпосередньо залежить від сили та часу дії. Вражаюча дія електричного струму включає: 1) Пошкодження м'яких тканин, що викликає їх розрив; 2) Опіки та обуглювання тканин 3) Розпад органічних сполук у тілі людини, включаючи плазму та кров; 4) Мимовільне скорочення м'язів і тканин, небезпечне тим, що може призвести до зупинки серця та атрофії м'язів.

Найнебезпечнішим є його вплив на органи дихання, мозок і серце. Крім того, електричний струм пригнічує на центральну нервову систему людини, і може призвести до повного або часткового паралічу.

Найважливішими факторами, що впливають на результат ураження електричним струмом, є: 1) величина струму, що протікає через тіло людини; 2) тривалість дії струму; 3) частота струму; 4) шлях проходження струму; 5) індивідуальні властивості організму людини.

У нормальних умовах найменший струм промислової частоти, який викликає фізіологічні відчуття у людини, в середньому дорівнює 1 міліамперу (мА); для постійного струму ця величина дорівнює 5 мА.

Тривалість дії струму. Тривалий вплив електричного струму з параметрами, що не представляли спочатку небезпеки для організму, може призвести до загибелі внаслідок зниження опору тіла людини.

Повинна дотримуватися електробезпеки при проведенні будівельно-монтажних робіт. Будівельні майданчики та приміщення, в яких проводяться електро-монтажні роботи, належать до об'єктів підвищеної чи особливої небезпеки.

Робітники, які мають справу з електричного струму, в тому числі, які користуються електрифікованим інструментом, повинні працювати в гумових рукавичках і галошах та на гумових н. ирал.

Вмикати електроустановки може лише спеціально навчений робітник.

Корпуси електроустаткування повинні бути надійними заземлені. У приміщеннях з підвищеною вологістю, а також у котлах, баках та поза приміщеннями напруга має бути не вище 36 В. звуку в дБА, використовується для орієнтовної оцінки постійного та непостійного шуму, тому що в цьому випадку ми не знаємо спектру шуму.

Батарейки та акумулятори. Не варто забувати і про небезпеку самих джерел живлення, таких як акумулятори та батареї. В разі наявності шлюбу, перебігу надмірної хімічної реакції або ж пошкоджень є ймовірність того, що з джерела живлення може потекти електроліт, що утворює на контактах окиснення. Цей процес може привести в непридатність контакти та вимагати їх заміну.

Крім того, якщо джерело живлення зазнало впливу короткого замикання, або мала місце бути зарядка від струмів більшої потужності ніж та, яку розрахована джерело живлення може різко здутися. Це відбувається внаслідок процесу розпаду н. ирал марганцю, внаслідок чого вивільняються кисень, тоді як у процесі корозії цинку

виділяється водень. Наявність у вільному стані газів усередині батареї та ведуть до збільшення її розмірів, що негативно позначається на міцності її корпусу.

У поєднанні з перегрівом, що виникає, може статися розлом корпусу батареї та вивільнення розпеченого електроліту, який може стати причиною пожежі та знищення обладнання.

Для запобігання подібним ситуаціям потрібне використання батарей та акумуляторів від надійних виробників, з відсутністю шлюбу та механічних пошкоджень, а також правильне заряджання. З часом застарілі акумулятори необхідно замінювати на нові.

Ізоляція струмовідних частин – один з основних заходів електробезпеки.

Опір ізоляції струмопровідних частин електричних установок щодо землі має бути не менше ніж 0,5–10 Мом (1 Мом = 10⁶ Ом).

Електрична ізоляція – це шар діелектрика, яким покривають поверхня струмопровідних елементів, або конструкція з непровідного матеріалу, за допомогою якого струмопровідні елементи відокремлюють від інших частин електроустановки.

Часто як додаткова ізоляція використовується корпус електроприймача, виготовлений з ізоляційного матеріалу. Такий корпус захищає від ураження електричним струмом не тільки при пробію ізоляції всередині виробу, але і при випадковому дотику робочої частини інструменту до струмовідної частини. Якщо ж корпус виробу металевий, то роль додаткової ізоляції відіграють ізоляційні втулки, через які живильний кабель проходить всередину корпусу, та ізолюючі прокладки, що відокремлюють електродвигун від корпусу.

Одним із способів захисту від ураження електрикою є мінімізації можливості зіткнення людини із джерелом струму. З цією метою електричне обладнання екранується діелектричним екраном, який також захищає обладнання та від механічного дії. Також використання захисного екіпірування, таких як гумове взуття та рукавички здатні захистити людину навіть при зіткненні з прямим

джерелом струму Другим способом є встановлення заземлень. Заземлення являє собою штучне з'єднання електрообладнання з заземлюючим пристроєм, що представляє собою набір провідників, та заземлювачем. Заземлювачі є електродами з металу.

Кожен електричний прилад підключається до заземлення Для робочої потужності в межах 1000 Вт, опір заземлення, відповідно до вимог, становить 15, 30 чи 60 Ом залежно від вибраного приладу.

Розрахунок заземлювальних пристроїв зводиться до визначення перехідного опору розтікання струму замикання на землю із заземлювачів, залежить від питомого опору шарів ґрунту ρ . Опір шарів ґрунту залежить від їх складу, вологості, рівня ґрунтових вод і температури. Найбільш точно ρ можна визначити безпосереднім проміром дома одним із існуючих методів.

Рекомендовані при попередніх розрахунках значення для різних ґрунтів та підвищуючі коефіцієнти γ н. ирально ґрунтах наводяться у довідниках [16].

5.3 Забезпечення безпеки виробничого обладнання

Крім забезпечення безпеки працівників підприємства, заходи щодо захисту повинні торкатися як і виробниче устаткування. Забезпечення захисту залежить від самого обладнання, а також від його розташування.

Основну небезпеку для виробничого обладнання представляє пил та вологу. Ступінь захищеності обладнання від останніх оцінюється за індексом IP XX, де перша цифра – ступінь захисту від пилу, а друга – ступінь захисту від вологи. IP 00 означає, що обладнання не має жодного захисту.

Максимальним значенням є IP69, який означає повний захист від пилу та захист від гарячої води при тривалому зануренні на глибину кілька метрів.

Останнє, як правило, необхідне для обладнання, яке знаходиться безпосередньо під водою.

Крім захисту від пилу та вологи, необхідного також забезпечити захист від механічних пошкоджень та збоїв у роботі обладнання. Робиться це шляхом встановлення захисного обладнання, такого як бар'єри, а також огороження [15].

5.4 Пожежна безпека

Пожежею називають процес неконтрольованого горіння. Вражаючи вплив пожеж, крім прямого впливу вогню, є утворення чадного газу в процесі горіння, а також виділення інших їдких хімічних речовин, що відбуваються при нагріванні, що знаходяться на виробництві об'єктів, таких як вироби із пластику та гуми.

Крім прямої небезпеки життя працівників підприємства, пожежа представляє величезну небезпеку і устаткування на підприємстві. Основною причиною пожеж є недбалість співробітників підприємства. Але також пожежі можуть бути спричинені й іншими факторами, такими як коротке замикання, вибух акумуляторів та батарей тощо.

Методи захисту. Встановлення пристроїв протипожежної сигналізації здатне вчасно повідомити персонажа про небезпеку. Також провадиться установка систем автоматичного гасіння пожеж. До таких ставляться розбризкувачі. Найбільш дешевим і одним з Найефективнішими способами гасіння пожеж є вода. Однак слід пам'ятати, що при горінні електроприладів, що знаходяться Крім того, деякі речовини тільки посилюють своє горіння від води. До таких відносяться нафтопродукти, масляні рідини і пластик. У цих випадках потрібне використання безводних засобів гасіння, якими є вогнегасники.

Вогнегасники за видом використовуваних засобів гасіння поділяються на три групи: пінні, газові та порошкові. З вогнегасника вогнегасна речовина може подаватися під тиском газів, що утворюються в результаті хімічної реакції; під

тиском заряду або робочого газу, що знаходиться над вогнегасною речовиною; під тиском робочого газу, що знаходиться в окремому балоні; вільним закінченням вогнегасної речовини [17].

5.5 Віброзахист

Вібрація – це механічні коливання машин, механізмів та їх елементів.

Основними джерелами вібрації є машини для приготування, розподілення та віброуцільнення бетонної суміші, а також будівельні машини, компресори та бульдозери. Вібрація відноситься до факторів, що мають високу біологічну активність.

При тривалому впливі вона викликає хронічне професійне захворювання – вібраційну хворобу. Дія вібрації залежно від тривалості впливу поділяють на постійну вібрацію, для якої контрольований параметр за час спостереження змінюється не більше ніж у 2 рази (на 6 дБ), та непостійну вібрацію, для якої зазначені параметри за час спостереження змінюються більш ніж 2 рази.

За характером на людини вібрація поділяється на: загальну та місцеву (локальну). Загальна вібрація передається через опорні поверхні людини, що сидить (стоїть) і впливає на весь організм. При дії на організм загальної вібрації насамперед страждає опорно-руховий апарат та нервова система і такі аналізатори, як вестибулярний, зоровий. Місцева вібрація – передається через руки або ділянки тіла людини, яка контактує з віброуючими поверхнями. Вона впливає окремі частини організму людини при роботі з віброінструментом, присутній на важелях керування транспортними засобами та н..

Якщо технічними засобами не вдається досягти виконання гігієнічних норм на робочому місці, то необхідно застосовувати засоби індивідуального захисту: віброзахисні рукавиці та віброзахисне взуття, наколінники, килимки, нагрудники, спеціальні костюми. З метою зниження рівня негативного впливу вібрації на

здоров'я працюючих сприяє застосування індивідуальних засобів захисту від вібрації. Віброзахисні рукавиці (з внутрішньої сторони обтягнуті тканиною для того, щоб запобігти роздратування).

Окрім організаційних заходів віробезпечні умови праці забезпечуються застосуванням віробезпечних машин, засобів активної та пасивного віброзахисту, що знижують вплив вібрації на працюючих в зонах її поширення, а також проектуванням технологічних процесів та виробничих приміщень, що забезпечують не перевищення норм вібрації на робочих місцях. Також слід передбачати обмеження ультразвуку, що передається контактним шляхом.

Зниження вібрації машин може досягатися шляхом зниження віброактивності та внутрішнім віброзахистом джерела. Причиною низькочастотних вібрацій насосів, компресорів, електродвигунів є невірноваженість елементів, що обертаються. Дія невірноважених динамічних сил посилюється поганим кріпленням деталей, їх зносом у процесі експлуатації. Усунення невірноваженості обертових мас досягається балансуванням.

Другий метод внутрішнього віброзахисту – вібродемпфування, тобто.

Перетворення енергії механічних коливань системи у теплову енергію.

Віброгасіння в системі досягається за допомогою динамічних віброгасників, що використовують ефекти інерції в'язкого, сухого тертя та т.п. Віброізоляція (у своєму розумінні цього терміна) полягає в зменшенні передачі вібрації від джерела об'єкту, що захищається (людина чи інший агрегат) шляхом введення додаткового пружного зв'язку[18].

5.6 Захист від впливу електромагнітних полів

Вибір того чи іншого способу захисту від дії електромагнітних випромінювань залежить від робочого діапазону частот, характеру виконуваних робіт, напруженості та щільності потоку енергії ЕМП, необхідного ступеня захист.

До заходів щодо зменшення на працівників. ЕМП відносяться: організаційні, інженерно-технічні та лікарсько-профілактичні. Організаційні заходи здійснюють органи санітарного нагляду. Вони проводять санітарний нагляд за об'єктами, у яких використовуються джерела електромагнітних випромінювань. Інженерно-технічні заходи передбачають таке розташування джерел ЕМП, яке б зводило до мінімуму їх вплив на працюючих, використання в умовах виробництва дистанційного керування апаратурою, є джерелом випромінювання, екранування джерел випромінювання, застосування засобів індивідуальної захисту (халатів, комбінезонів з металізованої тканини, з висновком на заземлюючий пристрій). Для захисту очей доцільно використовувати захисні окуляри ЗП5-90. Скло окулярів покрито напівпровідниковим оловом, що послаблює інтенсивність електромагнітної енергії.

Гігієнічне нормування електромагнітних полів. Метою гігієнічного нормування є встановлення гранично допустимих рівнів (ПДУ) на людей ЕМП з урахуванням особливостей опромінення та контингенту осіб.

Забезпечення захисту персоналу, професійно не пов'язаного з експлуатацією та обслуговуванням джерел ЕМП, здійснюється у відповідно до вимог гігієнічних нормативів ЕМП, встановлених населення.

Гранично допустимий рівень напруженості електростатичного поля (ЕСП) при дії менше 1 години за зміну встановлюється рівним 60 кВ/м. При більшій напруженості робота без засобів захисту не допускається. Час перебування у зоні впливу ЕСП при напруженості менше ніж 20 кВ/м не регламентується. У діапазоні 20 – 60 кВ/м допустимий час перебування персоналу в ЕСП без засобів захисту визначається за формулою зазначених норм.

Вплив змінних електромагнітних полів. Вплив електромагнітних полів на людину залежить від напруженостей електричного та магнітного полів, інтенсивності потоку енергії, частоти коливань, локалізації опроміненнь на поверхні тіла та індивідуальних особливостей організму.

Механізм цього впливу полягає в тому, що в електричному полі атоми та молекули, з яких складається людське тіло, поляризуються, а полярні молекули (наприклад, води), крім того, орієнтуються за напрямом поширення електромагнітного поля. У електролітах, якими є рідкі складові тканин, крові, міжклітинної рідини тощо, після застосування зовнішнього поля з'являються іонні струми. Змінне електричне поле викликає нагрівання тканин тіла людини як за рахунок змінної поляризації діелектриків, так та за рахунок появи струмів провідності.

Джерела випромінювань. Відомо, що біля провідника, яким протікає струм, виникають одночасно електричне та магнітне поля.

Якщо струм не змінюється у часі, ці поля залежать друг від друга. При змінному струмі магнітне та електричне поля пов'язані між собою, представляючи єдине електромагнітне поле.

Електромагнітне поле має певну енергію і характеризується електричною та магнітною напруженістю, що необхідно враховувати в оцінці умов праці.

Джерелами електромагнітних випромінювань служать радіотехнічні та електронні пристрої, індуктори, конденсатори термічних установок, трансформатори, антени, фланцеві з'єднання хвильоводних трактів, генератори надвисоких частот та н..

Захист від електромагнітних полів. Електромагнітне поле – це сфера поширення електромагнітних хвиль. Основні параметри електромагнітних хвиль – довжина хвилі λ , частота f та швидкість поширення світла c .

Засоби захисту від електромагнітних полів та випромінювань повинні задовольняти такі вимоги: 1) не спотворювати, суттєво, електромагнітне поле; 2) не знижувати якість технічного обслуговування та ремонту; 3) не знижувати продуктивність праці.

Захист людини від небезпечного впливу електромагнітного випромінювання здійснюється такими способами: 1) зменшення випромінювання джерела; 2) екранування джерела випромінювання та робочого місця; 3) встановлення санітарно-захисної зони; 4) поглинання або зменшення утворення статичних

зарядів електрики; 5) усунення зарядів статичної електрики; 6) застосування засобів індивідуального захисту.

Зменшення потужності випромінювання від джерела реалізується застосуванням поглиначів електромагнітної енергії; блокуванням випромінювання або зниженням його потужності для обертових антен у секторі, в якому знаходиться об'єкт, що захищається.

Поглинання електромагнітних випромінювань здійснюється поглинальним матеріалом шляхом перетворення енергії електромагнітного поля у теплову. Як такий матеріал застосовують каучук, поролон, пінополістирол, феромагнітний порошок зі сполучним діелектриком, волосяні мати, просочені графітом [19].

5.7 Виробничий пил як виробничий фактор шкідливості

Виробничим пилом називають зважені у повітрі, повільно осідають тверді частинки розмірами від кількох десятків часток мікрона. Багато видів виробничого пилу є аерозоль.

За розмірами частинок (дисперсності) розрізняють видимий пил, мікроскопічну та ультрамікроскопічну. Відповідно до загальноприйнятої класифікації всі види виробничої пилу поділяються на органічні, неорганічні та змішані. Перші, у свою чергу, поділяються на пилу природного (деревна, бавовняна, лляна, вовняна та н..) та штучного (пилу пластмас, гуми, смол та н..) походження, а другі – на металеву (залізна, цинкова, алюмінієва та н..) та мінеральний (кварцовий, цементний, азбестовий та н..) пил.

До змішаних видів пилу відносять кам'яновугільний пил, що містить частинки вугілля, кварцу та силікатів, а також пилу, що утворюються в хімічних та інших виробництвах.

Специфіка якісного складу пилу визначає можливість і характер впливу на організм людини. Певне значення має форма та консистенція пилових частинок, які

значною мірою залежать природи вихідного матеріалу. Так, довгі та м'які пилові частки легко осідають на слизовій оболонці верхніх дихальних шляхів і можуть стати причиною хронічного трахеїту та бронхіту.

Ступінь шкідливого впливу пилу залежить також від його розчинності у тканинних рідинах організму. Велика розчинність токсичного пилу посилює та прискорює її шкідливий вплив [20].

Висновок за розділом

У розділі, що стосується охорони праці, особливу увагу приділено безпеці використання роботизованих систем та електричних установок, а також заходам, спрямованим на захист працівників від різноманітних небезпек на виробництві. Забезпечення безпеки при експлуатації роботів є критично важливим, оскільки ці системи можуть становити загрозу для життя та здоров'я працівників, якщо не дотримуватись належних стандартів безпеки. Окрім цього, розглянуті питання захисту від ураження електричним струмом та забезпечення безпеки електрообладнання, що є важливим аспектом при роботі з електричними системами, зокрема роботами, що живляться від електрики.

Захист від дотику до об'єктів під напругою, належне заземлення та використання відповідного ізоляційного обладнання здатні значно знизити ризики, пов'язані з електричною небезпекою. Також важливими є правильне розміщення виробничого обладнання та його захист від пилу, вологи і механічних пошкоджень, що забезпечує безперебійну та безпечну роботу машин і роботів. Не менш важливою є пожежна безпека, оскільки пожежі можуть призвести до значних матеріальних збитків та загроз для життя.

Насамкінець, для підвищення загального рівня безпеки, необхідно проводити регулярний інструктаж працівників, забезпечуючи їх знаннями правил безпеки і наданням необхідних засобів захисту. Комплексний підхід до охорони праці дозволяє зменшити кількість нещасних випадків, що виникають через недотримання стандартів і заходів безпеки.

ВИСНОВКИ

У даній дипломній роботі розроблено систему керування роботом-маніпулятором на базі Arduino Uno 3, що дозволила створити ефективне, економічно доступне рішення для автоматизації задач, які потребують точного та плавного виконання рухів. Завдяки простоті використання платформи Arduino Uno 3, а також наявності великої кількості компонентів та бібліотек, було досягнуто швидкого прототипування та реалізації проекту.

Обрано компоненти для системи керування маніпулятором, зокрема мікроконтролера Arduino Uno 3, сервоприводи та датчики. Застосування сервоприводів для керування рухами маніпулятора та використання датчиків для визначення позиції дозволило досягти необхідної точності та стабільності рухів.

Розроблено алгоритми для керування маніпулятором, що дозволило досягти точного та плавного виконання робочих завдань. Алгоритми, що забезпечують корекцію траєкторії руху та управління швидкістю сервоприводів, забезпечили високу точність виконання операцій, зокрема при зміні напрямку руху та зупинках, що є важливим для автоматизованих систем.

Виконане тестування та оцінка ефективності системи показали, що система керування роботом-маніпулятором на базі Arduino Uno 3 здатна виконувати широкий спектр завдань з достатньою точністю і стабільністю. Маніпулятор продемонстрував високу ефективність при виконанні простих та середньо складних завдань, таких як переміщення об'єктів, маніпуляції з різними предметами в межах заданої траєкторії.

Оптимізовано алгоритми для забезпечення безперервності та мінімізації помилок у реальному часі, особливо при виконанні більш складних маніпуляцій або в умовах змінних навантажень на сервоприводи. Проте, завдяки застосуванню корекції траєкторії та плавного регулювання швидкості, вдалося досягти належної точності навіть у таких умовах.

В цілому, розроблена система керування роботом-маніпулятором на базі Arduino Uno 3 має великий потенціал для використання в навчальних цілях, а також

у малих і середніх виробничих процесах, де важлива висока точність, гнучкість і економічність. Цей проект підтверджує можливість ефективного застосування доступних технологій для створення роботизованих рішень, які можуть бути використані в реальних умовах.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. FANUC America. *Collaborative Robot Series Overview*. Режим доступу: <https://www.fanucamerica.com>.
2. Freschi, C., Ferrari, V., et al. (2012). *Technical review of the da Vinci surgical telemanipulator*. *International Journal of Medical Robotics and Computer-Assisted Surgery*. DOI: 10.1002/rcs.
3. Pires, J. N. (2022). *Industrial Robotics: Theory, Modelling, and Practical Applications*. Springer. ISBN: 9783030567988.
4. Корсак К. В., Гладисhevський М. Є. *Основи теорії автоматичного керування*. – Львів: Видавництво НУ "Львівська політехніка", 2013. – 356 с.
5. Шаблій О. І. *Системи керування з адаптацією: навчальний посібник*. – Харків: НТУ "ХПІ", 2018. – 256 с.
6. Жадан В. В., Дробітько В. І., Топчій С. І. *"Arduino у навчальних проектах робототехніки"* // Вісник Вінницького національного технічного університету. – 2020.
7. Факультет математики, природничих наук та технологій. *"Робототехніка: інструменти та платформи"* // Сайт ЦДПУ ім. В. Винниченка. Розглядаються переваги відкритих платформ, включно з Arduino.
8. Тітенко В.С. *Основи робототехніки: платформи Arduino та їх застосування* / В.С. Тітенко. – Київ: Наукова думка, 2020. – 310 с.
9. Ковальов Ю.А., Шевченко А.М. *Проектування і технології створення роботів-маніпуляторів* – Київ: Видавництво НТУУ "КПІ", 2020. – 300 с.
10. Скворцов, О.М. *Електроніка для робототехніки* / О.М. Скворцов. – Київ: Вища освіта, 2021. – 276 с.
11. Шевченко О.М. *Технології вимірювань у робототехніці* – Київ: НТУУ "КПІ", 2019. – 290 с.
12. Богданов, С. О., Ковальчук, М. Ю. *Основи робототехніки: теорія і практика* – Київ: НТУУ "КПІ", 2019. – 312 с.

13. Петренко, О. І. *Автоматизовані системи управління на базі мікроконтролерів* – Харків: ХНУ, 2021. – 310 с.
14. Специфікація роботів Makeblock Ultimate 2.0 // Електронна версія на сайті <https://www.makeblock.com/steam-kits/mbot-ultimate>
15. S. Nof. Handbook of Industrial Robotics, John Wiley & Sons, 2007.
16. M. Higuchi. Industrial Robots Programming: Building Applications for the Factories of the Future. Springer, 2020. – 384 p.
17. R. M. V Vasquez. Industrial Robots: Robot Selection, Programming, and Maintenance. Springer, 2016. – 297 p.
18. Ковальчук, В. І. *Основи вібраційної техніки та віброзахисту*. – Київ: Наукова думка, 2019.
19. Гринько, І. М. *Електромагнітні поля та їх вплив на здоров'я людини: технічні і медичні аспекти* – Київ: НТУУ "КПІ", 2018. – 200 с.
20. Горбатюк, М. О. *Охорона праці у гірничій промисловості* – Київ: Наукова думка, 2015.

Міністерство освіти і науки України
Національний університет кораблебудування імені адмірала Макарова

Кваліфікаційна магістерська робота на тему:
**СИСТЕМА КЕРУВАННЯ РОБОТОю МАНІПУЛЯТОРА НА БАЗІ ARDUINO
UNO R3**

Виконав: студент групи 6371мз
Блащук О. О.
Керівник: к.т.н., асистент кафедри
Іванов А. В.

Миколаїв – 2024

Слайд 1. Титульний лист

МЕТА РОБОТИ, ОБ'ЄКТ І ПРЕДМЕТ ДОСЛІДЖЕННЯ

Мета роботи – розробка системи керування робота-маніпулятора на базі Arduino Uno 3, яка забезпечить точне і ефективне виконання робочих завдань.

Об'єкт дослідження – система керування роботом-маніпулятором, що базується на платформі Arduino Uno R3.

Предмет дослідження – розробка та реалізація програмного забезпечення та апаратної частини системи керування маніпулятором на базі мікроконтролера Arduino Uno R3, зокрема, методи керування рухами маніпулятора, використання датчиків та сервоприводів.

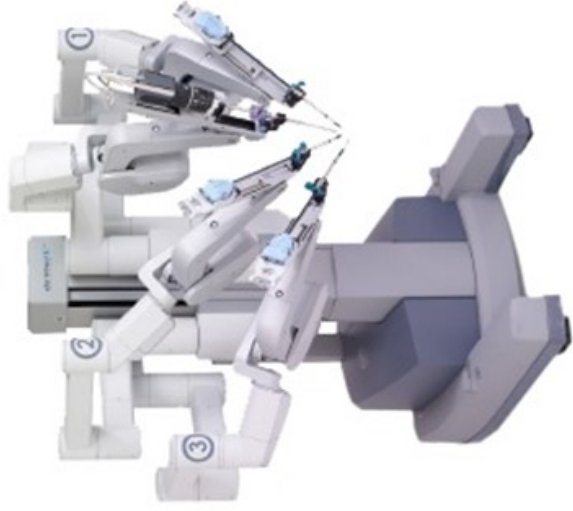
ЗАДАЧІ ДОСЛІДЖЕННЯ

- вивчення принципів роботи систем керування роботами-маніпуляторами;
- провести аналіз різних мікроконтролерів, датчиків, сервоприводів та інших компонентів, що використовуються в системах керування маніпуляторами, і вибрати оптимальні для реалізації на базі Arduino Uno R3;
- розробити алгоритми, які забезпечать точне і плавне виконання рухів маніпулятора, зокрема алгоритми для керування обертами сервоприводів, корекції траєкторії та управління швидкістю рухів;
- тестування та оцінка ефективності системи.

ІСНУЮЧІ МОДЕЛІ РОБОТІВ-МАНІПУЛЯТОРІВ



Промислові роботи

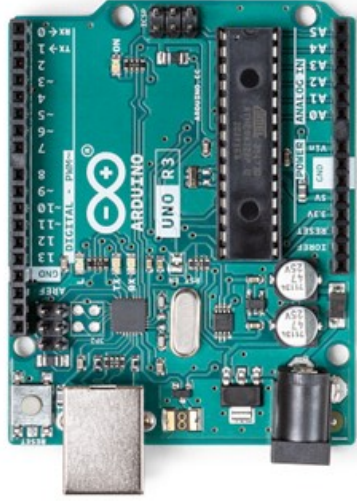


Медичні роботи



Логістичні роботи

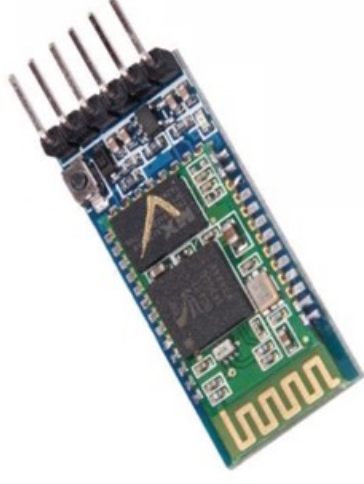
ОСНОВНІ ЕЛЕМЕНТИ СИСТЕМИ КЕРУВАННЯ



Arduino Uno R3



Сконструйований маніпулятор

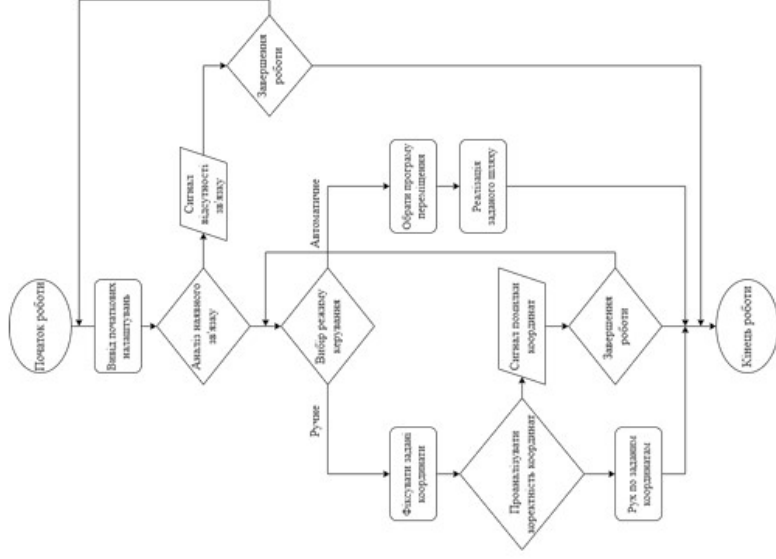


Bluetooth HC-05/06

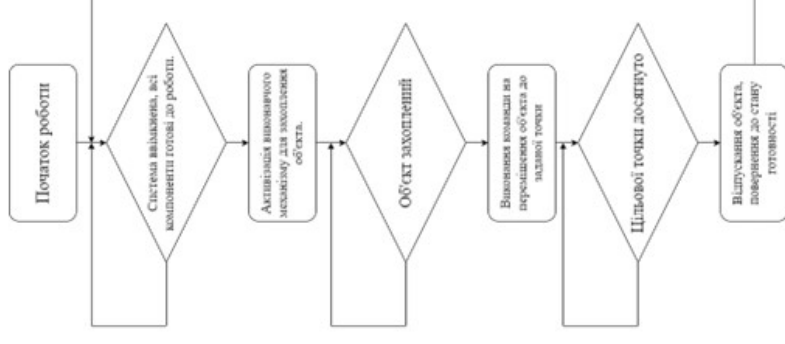


Сервомотор MG996R

АЛГОРИТМИ ОСНОВНОЇ РОБОТИ ТА ПОСЛІДОВНОСТІ ДІЙ СИСТЕМИ РОБОТА-МАНІПУЛЯТОРА

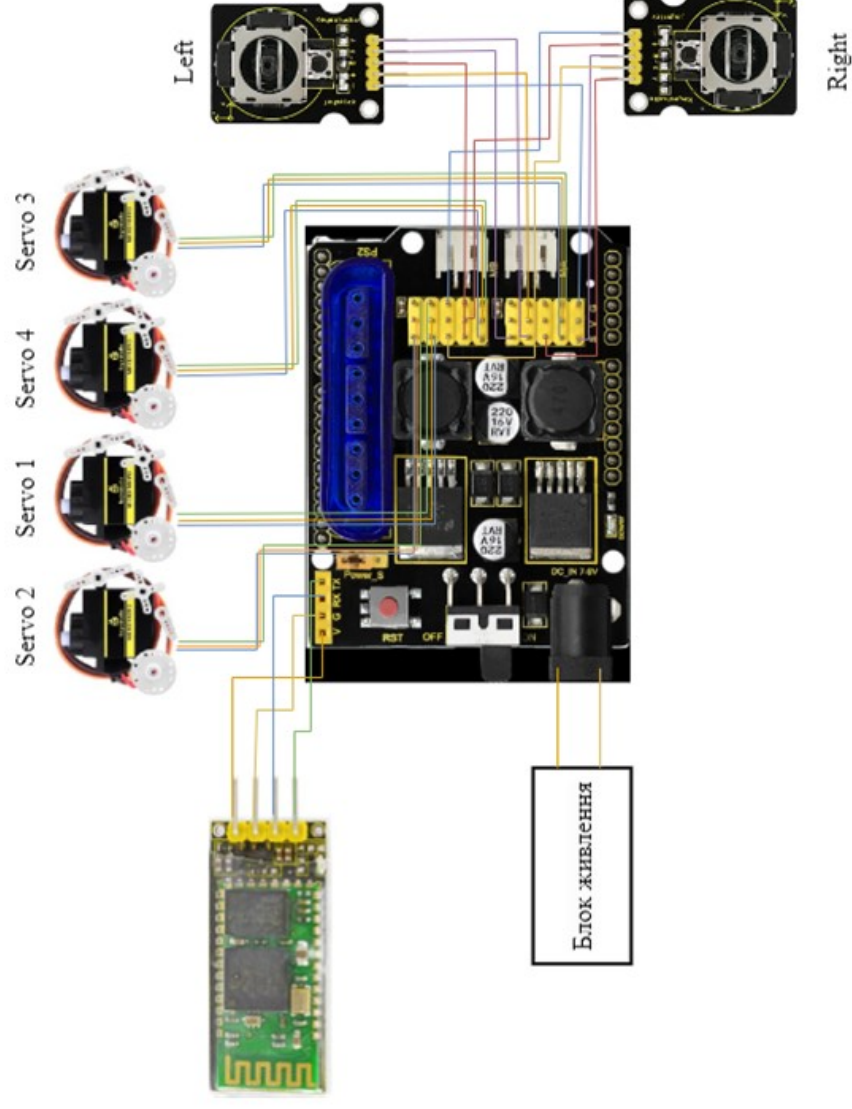


Алгоритм роботи робота-маніпулятора



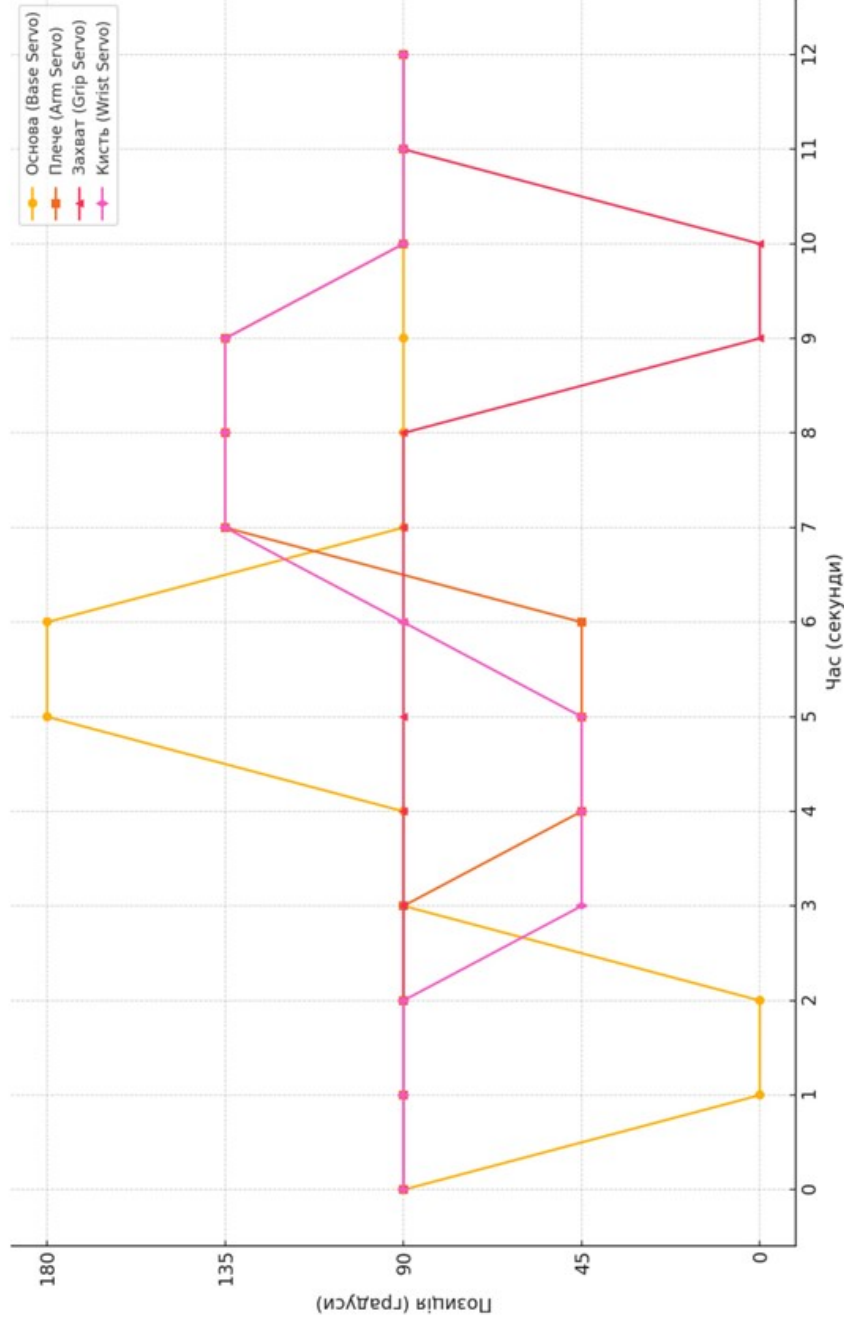
Алгоритм послідовності дій системи

СХЕМА ПІДКЛЮЧЕННЯ ЕЛЕМЕНТІВ СИСТЕМИ КЕРУВАННЯ



Слайд 7. Схема підключення елементів системи керування

РЕЗУЛЬТАТИ ПРОГРАМУВАННЯ РОБОТА-МАНІПУЛЯТОРА



Циклограма робота-маніпулятора

ВИСНОВКИ

- Обрано компоненти для системи керування маніпулятором, зокрема мікроконтролера Arduino Uno 3, сервоприводи та датчики.
- Розроблено алгоритми для керування маніпулятором, що дозволило досягти точного та плавного виконання робочих завдань.
- Виконане тестування та оцінка ефективності системи показали, що система керування роботом-маніпулятором на базі Arduino Uno 3 здатна виконувати широкий спектр завдань з достатньою точністю і стабільністю.
- Оптимізовано алгоритми для забезпечення безперервності та мінімізації помилок у реальному часі, особливо при виконанні більш складних маніпуляцій або в умовах змінних навантажень на сервоприводи.