

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Автоматики і комп'ютеризованих технологій
(повна назва)
Кафедра Комп'ютерно-інтегрованих технологій, автоматизації та
робототехніки
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА

Пояснювальна записка

рівень вищої освіти перший (бакалаврський)
Розроблення модуля автоматичного керування промисловим маніпулятором
(тема)

Виконав:
здобувач 3 року навчання,
групи АКТСІу-22-1
Єгор СЛУПСЬКИЙ
(власне ім'я, прізвище)

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

Тип програми освітньо-професійна
Освітня програма Системна
інженерія
(повна назва освітньої програми)

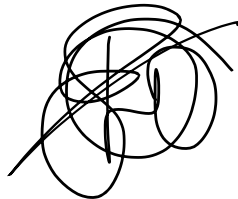
Керівник проф. Сергій НОВОСЕЛОВ
(посада, власне ім'я, прізвище)

Допускається до захисту

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

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

«31» травня 2025 р.

A handwritten signature in black ink, consisting of several overlapping loops and lines, positioned between the date and the name.

Єгор СЛУПСЬКИЙ

Харківський національний університет радіоелектроніки

Факультет Автоматики і комп'ютеризованих технологій
Кафедра Комп'ютерно-інтегрованих технологій, автоматизації та
робототехніки
Рівень вищої освіти перший (бакалаврський)
Спеціальність 151 – Автоматизація та комп'ютерно-інтегровані технології
(код і повна назва)
Тип програми освітньо-професійна
Освітня програма Системна інженерія
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

« _____ » _____ 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Слупському Єгору Юрійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Розроблення модуля автоматичного керування промисловим маніпулятором
затверджена наказом університету від 19 травня 2025 р. № 390 Ст
2. Термін подання здобувачем роботи до екзаменаційної комісії 09 червня 2025 р.
3. Вихідні дані до роботи _____
- 3.1 Тип маніпулятора – ангулярний
- 3.2 Кількість ступенів свободи маніпулятора – 3
- 3.3 Кількість крокових двигунів – 3
- 3.4 Мова програмування – C++
4. Перелік питань, що потрібно опрацювати в роботі
- 4.1 Вступ
- 4.2 Аналіз конструкцій маніпуляторів
- 4.3 Опис принципу роботи модуля автоматичного керування промисловим маніпулятором
- 4.4 Розробка архітектури автоматизованої системи
- 4.5 Вибір компонентів для побудови макету
- 4.6 Розробка алгоритму роботи
- 4.7 Розробка макету
- 4.8 Розробка програмного забезпечення
- 4.9 Експериментальні дослідження
- 4.10 Висновки

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри)

5.1 Графічний матеріал у вигляді презентації у форматі .pptx (12 с.)

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

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

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Строк / терміни виконання етапів роботи	Примітка
1	Аналіз літератури за темою кваліфікаційної роботи	28.04 – 01.05.2025	виконано
2	Аналіз аналогічних конструкцій	02.05 – 07.05.2025	виконано
3	Опис принципу роботи модуля автоматичного керування	08.05 – 10.05.2025	виконано
4	Підбір компонентів для побудови макету	10.05 – 12.05.2025	виконано
5	Розробка програмного забезпечення	13.05 – 20.05.2025	виконано
6	Складання макету	20.05 – 22.05.2025	виконано
7	Проведення експериментальних досліджень	22.05 – 24.05.2025	виконано
8	Оформлення пояснювальної записки	25.05 – 31.05.2025	виконано
9	Подання роботи на перевірку Інтернет-сервісом StrikePlagiarism		виконано
10	Подання роботи на рецензію		виконано
11	Подання роботи на підпис зав. кафедри		виконано
12	Подання роботи в ЕК		виконано

Дата видачі завдання 28 квітня 2025 р.

Здобувач _____

(підпис)

Єгор СЛУПСЬКИЙ

(посада, власне ім'я, прізвище)

Керівник роботи _____

(підпис)

проф. Сергій НОВОСЕЛОВ

(посада, власне ім'я, прізвище)

РЕФЕРАТ

Звіт містить: 73 с., 4 табл., 31 рис., 2 дод., 16 джерел.

МАНІПУЛЯТОР, КОНТРОЛЕР, ARDUINO, G-CODE, IoT,
АВТОМАТИЗАЦІЯ.

Об'єкт розробки – процес автоматизованого управління макетом промислового маніпулятора.

Предмет розробки – електронний модуль управління макетом промислового маніпулятора на основі контролера Arduino.

Мета роботи – розроблення модуля управління навчальним макетом промислового маніпулятора, призначеного для виконання базових операцій переміщення основних рухомих ланок пристрою та вдосконалення методики проведення лабораторних робіт з апаратних дисциплін.

Розроблено алгоритм роботи автоматизованого модуля управління макетом промислового маніпулятора. Виконано вибір компонентів для реалізації автоматизованої системи.

Описана конструкція макету промислового маніпулятора, наведено зовнішній вигляд макету промислового маніпулятора та описано призначення основних його компонентів. Докладно розглянуто механізм забезпечення руху ланок макету промислового маніпулятора, та наведена кінематична схема маніпулятора.

Розроблений модуль керування макетом промислового маніпулятора виконує поставлену задачу та може працювати в автоматичному режимі, виконуючи задану послідовність команд в циклі.

ABSTRACT

The report contains: 73 p., 4 tables, 31 figures, 2 appendices, 16 sources.

MANIPULATOR, CONTROLLER, ARDUINO, G-CODE, IoT, AUTOMATION.

The object of development is the process of automated control of the industrial manipulator model.

The subject of development is an electronic module for controlling the industrial manipulator model based on the Arduino controller.

The purpose of the work is to develop a control module for a training industrial manipulator model, designed to perform basic operations of moving the main moving parts of the device and improve the methodology for conducting laboratory work in hardware disciplines.

The algorithm for the operation of the automated module for controlling the industrial manipulator model has been developed. The components for implementing the automated system have been selected.

The design of the industrial manipulator model is described, the appearance of the industrial manipulator model is given, and the purpose of its main components is described. The mechanism for ensuring the movement of the links of the industrial manipulator model is considered in detail, and the kinematic diagram of the manipulator is presented. The developed control module for the industrial manipulator model performs the task and can operate in automatic mode, executing a given sequence of commands in a cycle.

ЗМІСТ

Скорочення та умовні позначки.....	9
Вступ.....	10
1 Аналіз технічного завдання.....	12
1.1 Аналіз призначення промислових маніпуляторів.....	12
1.2 Аналіз принципів побудови промислових маніпуляторів.....	21
1.3 Постановка задачі.....	27
2 Опис принципу роботи модуля автоматичного керування промисловим маніпулятором.....	29
2.1 Розроблення архітектури учбового лабораторного макету.....	29
2.2 Розроблення структурної схеми автоматизованого модуля управління макетом промислового маніпулятора.....	31
2.3 Опис конструкції макету промислового маніпулятора.....	33
2.4 Розроблення алгоритму роботи автоматизованого модуля керування макетом промислового маніпулятора.....	38
3 Вибір компонентів для реалізації автоматизованої системи.....	40
3.1 Вибір контролера.....	40
3.2 Плата розширення для контролера Arduino Nano.....	44
3.3 Вибір драйвера крокового двигуна.....	45
4 Розробка програми роботи блока управління промисловим маніпулятором.....	50
4.1 Розроблення програми для мікроконтролера.....	50
4.2 Опис принципу обміну командами з модулем керування.....	58
4.3 Моделювання системи автоматичного управління для рішення задачі слідкування маніпулятором за заданою траєкторією.....	62
4.4 Результати експериментальних досліджень.....	63
5 Охорона праці.....	67
Висновки.....	69

Перелік джерел посилань.....	71
Додаток А Код програми	73
Додаток Б Демонстраційний матеріал.....	82

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

ПЗ – програмний засіб;

ПК – персональний комп'ютер;

ПР – промисловий робот;

ПЛК – програмований логічний контролер;

ПУЕ – правила улаштування електроустановок;

FBD – Function Block Diagram (функціональні блокові діаграми);

IL – Instruction List (список інструкцій);

LD – Ladder Diagram (релейно-контакторні схеми);

RPM – Revolutions Per Minute;

SFC – Sequential Function Chart (послідовні функціональні діаграми);

SPI – Serial Peripheral Interface;

ST – Structured Text (структурований текст);

UART – Universal asynchronous receiver/transmitter.

ВСТУП

Розроблення модуля автоматичного керування макетом промисловим маніпулятором є актуальним завданням у галузі автоматизації та робототехніки. З урахуванням стрімкого розвитку промислових технологій та зростання потреби в автоматизованих рішеннях, створення та впровадження подібних макетів стають важливою частиною підготовки фахівців та впровадження нових виробничих рішень на сучасних виробництвах.

Актуальність застосування маніпуляторів на виробництві зумовлена стрімким розвитком промисловості, а саме впровадження концепції Industry 4.0 та 5.0, зростанням вимог до якості продукції, швидкості виконання операцій та зниження витрат на виробництво одиниці продукції.

В умовах високої конкуренції підприємствам необхідно автоматизувати рутинні та трудомісткі процеси, щоб підвищити продуктивність, скоротити вплив людського фактора та забезпечити стабільність очікуваних результатів. Роботизовані маніпулятори є ключовим елементом автоматизованих систем, вони дозволяють виконувати такі завдання, як збирання, упаковка, сортування, зварювання та транспортування деталей в автоматичному режимі. Дані пристрої особливо потрібні в умовах багатоопераційного та безперервного виробництва, де важливі точність, повторюваність та висока швидкість дій. Крім того, використання маніпуляторів підвищує гнучкість виробничої лінії, що дозволяє швидко переналаштовувати обладнання під випуск нової продукції. З розвитком технологій управління, використання інтелектуальних датчиків та систем машинного зору маніпулятори стають все більш інтелектуальними та універсальними, що робить їх незамінними в умовах цифровізації та переходу до концепції «розумного виробництва».

Метою даної роботи є розроблення модуля управління навчальним макетом промислового маніпулятора, призначеного для виконання базових операцій переміщення основних рухомих ланок пристрою.

Об'єктом розробки є процес автоматизованого управління макетом промислового маніпулятора.

Предметом розробки є електронний модуль управління макетом промислового маніпулятора на основі контролера Arduino.

Зокрема, проєкт сприяє реалізації Цілі 4 «Якісна освіта», забезпечуючи практичну підготовку фахівців у сфері робототехніки та інформаційних технологій; Цілі 9 «Індустріалізація, інновації та інфраструктура» через впровадження інноваційних рішень у розробку інтелектуальних технічних систем

Для досягнення поставленої мети необхідно вирішити такі завдання:

- провести аналіз існуючих конструкцій промислових маніпуляторів та принципів керування ними;
- розробити структурну схему модуля керування макетом промислового маніпулятора;
- виконати вибір програмного рішення для керування маніпулятором;
- обрати компоненти для побудови модуля керування макетом промислового маніпулятора;
- розробити програмний інтерфейс для реалізації основних функцій управління рухом маніпулятора;
- оформити кваліфікаційну роботу згідно ДСТУ 3008:2015 [1], а також з методичними вказівками з підготовки й оформлення кваліфікаційної роботи здобувачами першого (бакалаврського) рівня вищої освіти спеціальності 151 Автоматизація та комп'ютерно-інтегровані технології освітньої програми «Системна інженерія» [2].

1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ

1.1 Аналіз призначення промислових маніпуляторів

1.1.1 Класифікація промислових маніпуляторів

Промисловий маніпулятор – автоматична машина з програмним керуванням, яка відтворює рушійні і розумові функції людини при виконанні виробничих процесів, також це клас машин для обслуговування технологічних процесів та автоматизації транспортних робіт [3].

Промислові роботи є важливими компонентами автоматизованих гнучких виробничих систем (ГВС), які дозволяють збільшити продуктивність праці. Типове застосування роботів стосується таких операцій, як зварювання, фарбування, складання, вибірка та встановлення, пакування, контроль продукції та випробування, котрі виконуються з високою надійністю, швидкістю, і точністю.

До основних складових промислових маніпуляторів відноситься:

- механічна система (маніпулятор) – це механізм, який автоматично відтворює функції руки людини при виконанні виробничих операцій шляхом переміщення об'єкта в просторі;

- система програмного керування;

- інформаційна система.

По ступені досконалості і типам систем керування промислові роботи поділяються на три покоління [4]. Перше покоління – програмні роботи (промислові), які працюють за жорсткими програмами. Вони широко застосовуються в машинобудуванні. Друге покоління – адаптивні роботи, які працюють за гнучкими програмами і змінюють характер робіт в залежності від зміни параметрів зовнішнього середовища. Третє покоління – коботи, які працюють разом з людьми у виробничому просторі та мають розвинений адаптивний інтерфейс керування, що поєднується з інноваційною

мехатронною системою, яка забезпечує їх безпечне застосування в людському оточенні [5].

Маніпулятори класифікуються за такими ознаками:

а) за характером виконуючих робіт:

- 1) універсальні;
- 2) спеціальні;

б) за способом керування (рис. 2.1):

- 1) з ручним управлінням (копіюючи);
- 2) з автоматичним управлінням;
- 3) з супервізорним керуванням;

в) спеціальні маніпулятори застосовуються:

- 1) в ковальсько-пресових цехах;
- 2) в ливарному виробництві.

Універсальні маніпулятори застосовуються для виконання підйомно-транспортних робіт в машинобудуванні, а також для механізації операцій по обслуговуванню технологічного обладнання (установки і знімання деталей на металорізальних верстатах, зварочні і свердлильні операції) [6].

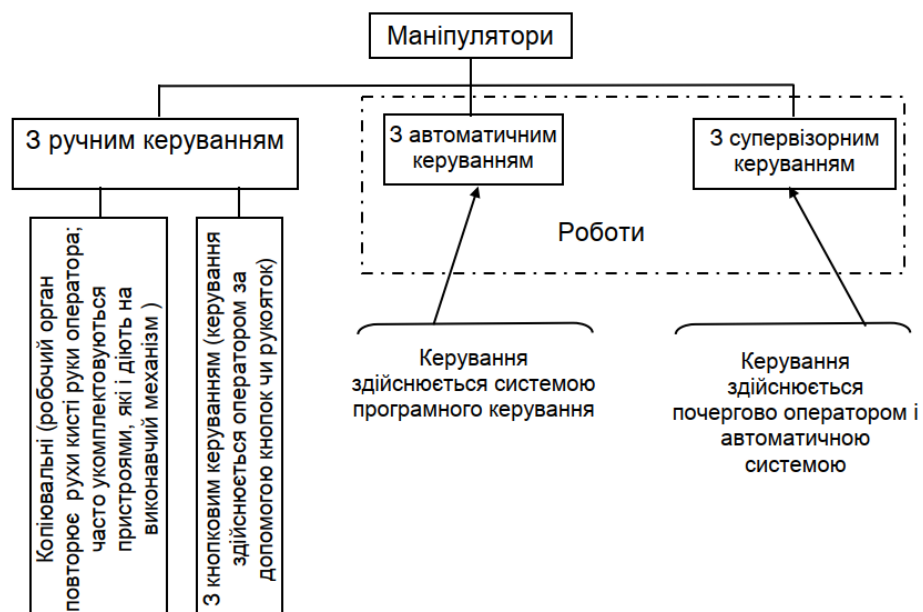


Рисунок 1.1 – Класифікація маніпуляторів за методом керування

Працездатність маніпуляторів характеризується рядом технічних показників: робочий об'єм, маневреність, кут і коефіцієнт сервісу, число ступеней свободи, вантажопідйомність, швидкість руху, енергетичні показники.

Системи керування класифікують за такими ознаками:

а) за способом позиціювання:

- 1) позиційні (малоточкові, багатоточкові);
- 2) контурні;
- 3) комбіновані;

б) за видом енергоносія:

- 1) механічні;
- 2) гідропневматичні;
- 3) електричні;

в) за способом представлення (подання) інформації:

- 1) електромеханічні;
- 2) числові;
- 3) гібридні
- 4) циклові;
- 5) аналогові;

г) за об'єктом інформації в управляючій програмі:

- 1) жорстко програмні;
- 2) адаптивні;

д) за способом програмування:

- 1) методом розрахунку (складання програм);
- 2) методом навчання;
- 3) самонавчаємі.

Інформаційні системи в значній мірі визначають функціональні можливості промислових роботів, складність задач, експлуатаційну надійність і ефективне використання їх в виробничих умовах, а також вони

являються важливими ланками в забезпеченні безпеки обслуговуючого персоналу.

На рисунку 1.2 показана схема, що пояснює принцип класифікації промислових роботів-маніпуляторів.



Рисунок 1.2 – Схема, що пояснює принцип класифікації промислових роботів-маніпуляторів.

Промислові роботи класифікуються за такими ознаками:

а) за характером виконуваних технологічних операцій:

1) технологічні – промислові роботи, які виконують основні операції технологічного процесу;

2) допоміжні – промислові роботи, які виконують допоміжні операції щодо обслуговування основного технологічного устаткування і транспортно-складської системи;

3) універсальні – промислові роботи, які виконують різні операції;

б) за видом виробництва:

1) ливарні;

2) зварювальні;

3) ковальсько-пресові;

4) фарбувальні;

5) транспортно-складські тощо;

в) за системою координат руки маніпулятора:

1) прямокутна (плоска та просторова);

2) полярна і циліндрична;

3) сферична;

4) ангулярна (кутова) (плоска, циліндрична та сферична);

г) за вантажопідйомністю:

1) надлегкі (0,08; 0,16; 0,32; 0,40; 0,50; 0,63; 0,80; 1,0 кг);

2) легкі (1,25; 1,60; 2,0; 2,50; 3,2; 4,0; 5,0; 6,3; 8,0; 10,0 кг);

3) середні (12,5; 16,0; 20,0; 25,0; 32,0; 40,0; 50,0; 63,0; 80,0; 100,0; 125,0; 160,0; 200,0 кг);

4) важкі (250; 320; 400; 500; 630; 800; 1000 кг);

5) надважкі (1250);

д) за типом силового приводу:

1) роботи з електромеханічними приводами;

2) роботи з пневматичними приводами;

3) роботи з гідравлічними приводами;

4) роботи з комбінованими приводами;

е) за видом програми:

1) жорсткопрограмовані – промислові роботи, які не мають пристрою швидкої зміни програми;

2) гнучкопрограмовані – промислові роботи, які мають пристрій швидкої зміни програми;

3) адаптивні – промислові роботи, які здійснюють свої дії на підставі інформації про об'єкти і явища зовнішнього середовища, отриманої в процесі роботи;

4) інтелектуальні – промислові роботи, що здатні самостійно планувати свою поведінку залежно від поставленого завдання, критеріїв якості, власного стану і стану зовнішнього середовища;

ж) за характером керування:

1) позиційні;

2) контурні;

3) комбіновані.

1.1.2 Основні технічні показники промислових роботів

До технічних показників відносяться:

– номінальна вантажопідйомність;

– робочий простір (простір, в якому може знаходитися виконавчий (робочий) механізм робота при його функціонуванні);

– зона обслуговування (частина робочого простору, в якій може виконувати роботу промисловий робот);

– число ступенів рухомості (залежить від його конструкції ходового пристрою (від 1 до 3));

– швидкість переміщення;

– похибка позиціонування робочого органу і похибка траєкторії робочого органу (похибка позиціонування робочого органу – відхилення реального

положення робочого органу від заданого програмою; відхилення траєкторії робочого органу від заданої програмою називається похибкою траєкторії).

Вантажопідйомність – найбільше значення маси об'єктів маніпулювання, включаючи і масу робочого органу, які можуть переміщатися «рукою» при заданих умовах. Для промислового робота з багатьма руками вантажопідйомність визначають як суму вантажопідйомності всіх його «рук». Для деяких типів промислових роботів важливим показником є зусилля (або крутний момент), яке розвивається виконавчим механізмом. До числа таких показників можна віднести зусилля затиску об'єкта маніпулювання захоплювальним пристроєм, робоче зусилля «руки» промислового роботу уздовж її поздовжньої осі, крутний момент при ротації захоплювального пристрою.

За величиною вантажопідйомності промислові роботи поділяють на надлегкі (до 1 кг), легкі (понад 1 кг до 10 кг), середні (понад 10 кг до 200 кг), важкі (понад 200 кг до 1000 кг), надважкі (понад 1000 кг). В даний час випускається до 73% моделей промислових роботів легкого і середнього типу з вантажопідйомністю від 5 кг до 80 кг.

Число ступенів рухливості визначають як суму можливих координатних рухів його робочого органу або об'єкта маніпулювання щодо опорної системи. Для деяких типів додатково враховують число ступенів рухливості захоплювального пристрою, рівне числу ступенів вільності всіх його ланок щодо вузла кріплення до «руки» робота. Серед ступенів рухливості окремого маніпулятора слід розрізняти переносні і орієнтують.

За ступенем рухливості промислові роботи розділяють на три групи: малу (до 3-х ступенів рухливості); середню (4-6 ступенів рухливості) і високу рухливість (понад 6 ступенів). Число ступенів рухливості промислового робота значною мірою визначає його універсальність. Сучасні промислові роботи мають зазвичай від 2 до 7 ступенів рухливості: найпростіші – 1-2; найбільш складні – 7, іноді і більше. У структурі сучасного світового парку

промислових роботів переважають конструкції з 4 і 5 ступенями рухливості (67%).

Величини і швидкості переміщення робочого органу з кожного ступеня рухливості характеризують геометрію робочого простору промислового робота, а також особливості руху та орієнтації об'єкта який переносимо та для рухливих роботів крім робочої зони в якості характеристики розглядається також робочий простір, визначаються механікою маніпулятора промислового робота і можливостями приводу. Величини переміщень по лінійним координатам задаються в метрах, за кутовими – в градусах або радіанах; відповідно швидкості виражаються в метрах в секунду для лінійних і градусах або радіанах в секунду – для кутових координат. За величиною лінійного переміщення, або ходу робочого органу, розрізняють промислові роботи з малим (до 300 мм), середнім (понад 300 мм до 1000 мм) і великим ходом (понад 1000 мм).

Швидкість переміщень ланок маніпулятора характеризують важливу якість промислового робота – швидкодію, від якого залежить час обслуговування технологічного устаткування. Зазвичай швидкості лінійних переміщень робочих органів маніпуляторів не перевищують 1,0 – 1,5 м/с, хоча є окремі роботи зі швидкостями до 9 м/с. Швидкості кутових переміщень робочих органів переважно знаходяться в межах від 15 град/с до 360 град/с (від 0,25 град/с до 6,3 рад/с).

Робоча зона промислових роботів – це простір, в якому може знаходитися робочий орган при його функціонуванні. Вона характеризується своїми формою (або видом) і обсягом.

Форма, або вид робочої зони, просторова фігура, описувана робочим органом промислових роботів при проходженні ним гранично досяжних положень. Вид робочої зони обумовлений призначенням промислового робота і визначається числом ступенів рухливості маніпулятора, типом і взаємної орієнтацією кінематичних пар в просторі і відносними розмірами ланок маніпуляційної системи.

Обсяг робочої зони (або робочий об'єм) – це обсяг простору, в межах якого може переміщатися робочий орган промислового робота при його функціонуванні, вимірюється в метрах кубічних і дозволяє судити про масштаб ручної праці, доступного промислового роботу. За величиною обслуговується робочого об'єму всі роботи можна розділити на п'ять класів.

Найбільшого поширення в даний час отримали середні промислові роботи (45%), на другому місці малі (30%), потім великі (12%), міні-роботи (9%) і, нарешті, мікророботи (4%). в якому може знаходитися виконавчий пристрій при його функціонуванні. Воно визначається без урахування габаритних розмірів об'єкта маніпулювання. Іншою характеристикою простору функціонування промислового робота є зона обслуговування, в якій робочий орган виконує свої функції відповідно до призначення робота і встановленими значеннями його характеристик. При спільній роботі декількох промислових роботів як характеристики робототехнічного комплексу розглядається зона спільного обслуговування – частина простору, в якому переміщення об'єкта маніпулювання можуть виконуватися кількома промисловими роботами.

Похибка позиціонування, або відпрацювання траєкторії – це відхилення фактичного положення (або траєкторії) робочого органу від заданого програмою при багаторазовому позиціюванні (повторенні руху), яке оцінюється в лінійних або кутових одиницях. Похибка позиціонування, або відпрацювання, траєкторії може розглядатися як в цілому для робочого органу (сумарна), так і для окремих виконавчих механізмів промислових роботів.

У характеристиках промислового робота точність позиціонування вказується в абсолютних одиницях, однак для порівняльної оцінки різних роботів і визначення їх типів з точки зору ступеня їх точності такий показник неприйнятний, оскільки не пов'язаний з геометричними параметрами робочої зони. Тому в основу визначення типу промислового робота за показником

точності покладена відносна похибка позиціювання, або відпрацювання траєкторії.

Відносна похибка позиціювання – величина, що характеризує точність робіт з цикловим і позиційним керуванням і рівна відношенню абсолютної похибки позиціювання до максимального віддалі від осі найближчої до основи робота кінематичної пари до кордону робочої зони, виражена у відсотках.

1.2 Аналіз принципів побудови промислових маніпуляторів

Структурна схема промислового робота показана на рисунку 1.3.

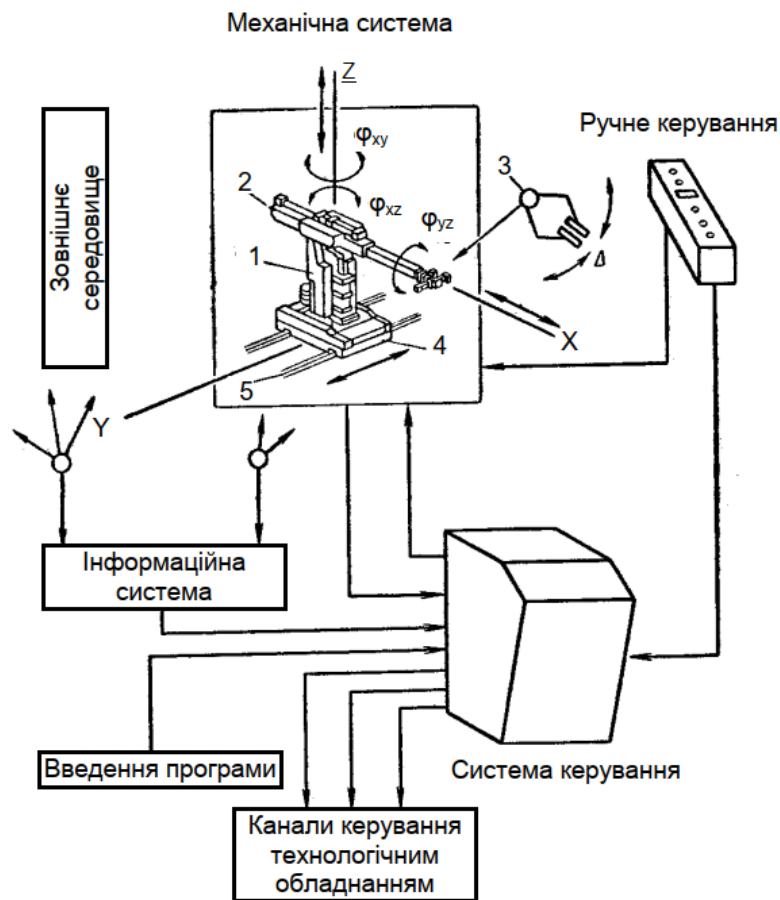


Рисунок 1.3 – Структурна схема промислового робота

На рисунку 1.3 введені наступні позначення:

- 1 – корпус;
- 2 – рука;
- 3 – захватний пристрій;
- 4 – основа;
- 5 – шляхопровід;
- XX – рух руки вздовж поздовжньої осі;
- YY – рух корпуса маніпулятора по шляхопроводу;
- ZZ – рух руки вверх-вниз;
- ϕ_{yz} – обертання ротація захвату;
- ϕ_{xz} – коливання руки (поворот відносно осі YY);
- ϕ_{xy} – поворот корпуса маніпулятора навколо вертикальної осі.

Система програмного керування забезпечує навчання (програмування), збереження програми і її введення (зчитування інформації і передачу керуючих сигналів виконавчим органам промислового робота).

Інформаційна система за допомогою зовнішніх і внутрішніх давачів здійснює збір і передачу в систему програмного керування інформацію про стан навколишнього середовища і функціонування основних вузлів і механізмів робота.

Механічна система відпрацьовує задану програму; її часто називають власне маніпулятором.

Відносні рухи ланок механічної системи промислового робота під дією яких реалізуються степені вільності, поділяються на три групи: орієнтуючі (локальні), транспортуючі (регіональні) і координатні (глобальні).

Сучасні промислові роботи відрізняються різноманітністю форми та розмірів. Вони здатні виконувати різні маніпуляційні операції і мають різні системи реалізації рухів.

На рисунку 1.4 зазначені системи координат при виконанні завдання роботом. У базовій системі координат визначається положення робота й у

більшості випадків початкові положення ланок і зап'ястя. У системі координат, зв'язаної з інструментом виражаються параметри положення інструмента за 6 ступенями рухливості (X, Y, Z і A, B, C).

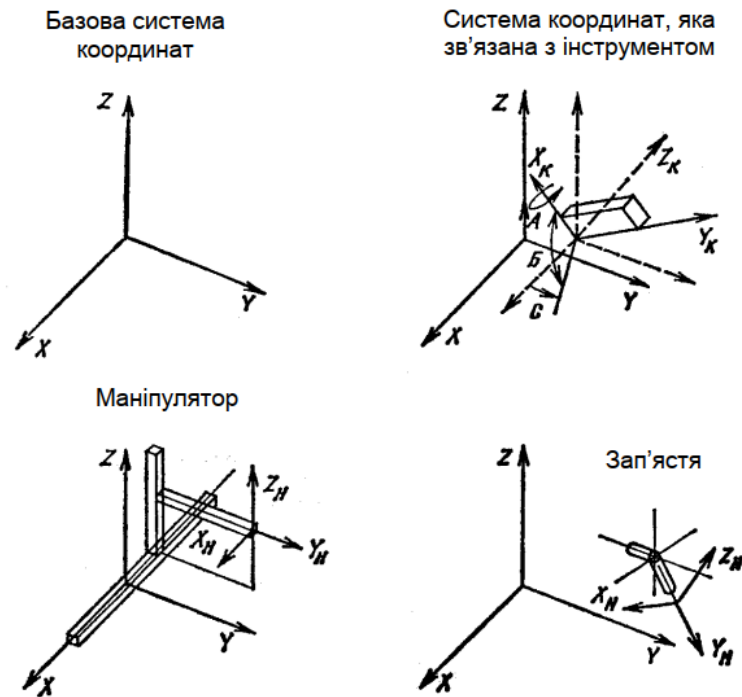


Рисунок 1.4 – Визначення систем координат при виконанні завдання роботом

Робот, маніпулятор і зап'ястя зв'язують базову систему координат із системою координат інструмента.

Розрізняються осі декількох типів:


– вісь обертального руху являє собою таке з'єднання двох жорстких елементів, коли один з них може обертатися відносно іншого навколо фіксованої осі;

– вісь поступального руху являє собою таке з'єднання двох жорстких елементів, коли один може рухатися прямолінійно по іншому елементу.

Складне зчленування являє собою таке з'єднання двох жорстких елементів, коли один з них може обертатися відносно іншого навколо рухливої осі.

У таблиці 1.1 приведені використовувані схематичні позначення. Кінематична схема може бути складена з осей поступального й обертального руху. Можливе також використання складних зчленувань.

Таблиця 1.1 – Символічне позначення вузлів промислових роботів

Назва	Символічне позначення	Схематичне позначення
Поступальна ланка (телескопічна)	X, Y, Z	
Поступальна ланка (поперечна)	U, V, W	
Обертальна ланка (осьова)	A, B, C	
Обертальна ланка (шарнірна)	D, E, P	
Захват	–	
Інструмент	–	
Розділення маніпулятора зап'ястя	i	
Наявність у робота декількох незалежних кінематичних ланок	Початок (Кінець)	–

У таблиці 2.2 показаний приклад кінематичної схеми [7]. Число можливих варіантів побудови промислового робота визначається наступним чином:

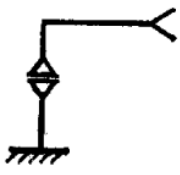
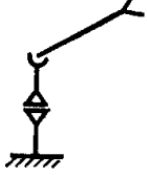


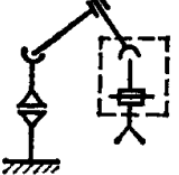
$$V = 6^{DF}, \quad (1.1)$$

де V – число варіантів;

DF – число ступенів рухливості.

Число можливих варіантів побудови кінематичних ланцюгів велике, наприклад, для 6 осей можливо 46656 варіантів. Однак, більшість з них не застосовано з кінематичних розумінь.

Таблиця 1.2 – Варіанти кінематичних схем

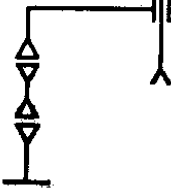
1DOF C	2DOF CB	3DOF CBR	4DOF CBR/E	5DOF CBR/EP
				

У таблиці 1.3 приведені критерії і приклади неприйнятних кінематичних схем [8].

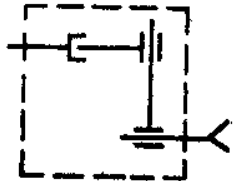
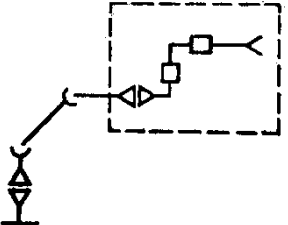
Додаткові обмеження викликаються наступними причинами:

- точність позиціювання зменшується зі збільшенням числа ланок;
- зі збільшенням числа ланок значно збільшується обсяг обчислень, необхідних для керування рухом по траєкторії;
- додавання нової ланки викликає додаткові витрати (ланцюг живлення, гальмо, двигун, давач положення, керування приводом, пам'ять);
- зі збільшенням числа ланок ускладнюється енергопостачання.

Таблиця 1.3 – Критерії і приклади неприйнятних кінематичних схем

Критерій	Приклад
1	2
Послідовне з'єднання ланок одного типу	

Продовження таблиці 1.3

1	2
Неможливо здійснити зміну орієнтації зап'ястя	
Зміна орієнтації досягається в результаті складних переміщень ланок маніпулятора і зап'ястя	

Вибір системи керування роботом (циклове чи контурне) і числа програмно керованих осей залежить від вибору кінематичної схеми. Основна закономірність проектування роботів: застосування ланок поступального руху вимагає складних механічних пристроїв і досить простої системи керування. Для робота, побудованого на ланках обертального руху – навпаки. Таким чином, вибір кінематичної схеми – важливий етап проектування механічної частини і системи керування роботом.

Переважає більшість промислових роботів володіють однією з чотирьох варіантів [9, 10]:

- з полярною системою координатних переміщень;
- з циліндричною системою координатних переміщень;
- з шарнірно-зчленованою рукою;
- з декартовою системою координатних переміщень.

Компонування з полярною системою координатних переміщень має також назву «компонування із сферичною системою координатних переміщень», оскільки робочий простір, у якому робот може переміщувати руку, являє собою частину сфери. Робот має поворотну основу та обертову ступінь рухомості, яка використовується для підняття і

опускання телескопічної руки. Цьому варіанту відповідає конструкція одного з самих відомих роботів – Unimate (серія 2000).

У варіанті «компонування з циліндричною системою координатних переміщень» корпус робота являє собою вертикальну колону, яка обертається навколо своєї вертикальної осі. Рука складається з кількох ортогонально розміщених ланок із зчленуваннями, які ковзають, що дозволяє їй підніматися, опускатися, втягуватися і витягуватися по відношенню до корпусу. Прикладом конструкції з циліндричною системою координатних переміщень може бути робот Prab Versatran (модель FC).

Компонування з шарнірно-зчленованою рукою нагадує руку людини. Рука робота складається з декількох прямолінійних ланок, які з'єднані шарнірними зчленуваннями. Вони аналогічні плечовому, ліктьовому і зап'ястному суглобам людської руки. Рука робота встановлена на основі, яка може обертатись, забезпечуючи роботу можливість працювати в квазісферичному просторі. Прикладом цього компонентування служать роботи моделей Cincinnati Milacron T3 та Unimate PUMA.

Компонування з декартовою системою координатних переміщень, складається з трьох ортогональних ланок з зчленуваннями, які ковзають. Ці три ланки відповідно переміщуються паралельно осям x , y і z декартової системи координат. За допомогою відповідних рухів цих ланок робот здатен перемістити руку у будь-яку точку тримірної робочого простору, який має форму прямокутного паралелепіпеда.

1.3 Постановка задачі

Проведений аналіз аналогічних рішень показав, що найбільш перспективним для мілкосерійного виробництва та електронного приладобудування маніпулятори з ангулярною кінематичної схемою – три

ступені свободи, плюс кріплення робочого інструменту. Наведені різні варіанти кінематичних схем побудови механічної частини пристрою.

Таким чином, в подальшій роботі потрібно:

- розробити структурну схему модуля керування макетом промислового маніпулятора;
- виконати вибір програмного рішення для керування маніпулятором;
- обрати компоненти для побудови модуля керування макетом промислового маніпулятора;
- розробити програмний інтерфейс для реалізації основних функцій управління рухом маніпулятора.

2 ОПИС ПРИНЦИПУ РОБОТИ МОДУЛЯ АВТОМАТИЧНОГО КЕРУВАННЯ ПРОМИСЛОВИМ МАНІПУЛЯТОРОМ

2.1 Розроблення архітектури учбового лабораторного макету

На рисунку 2.1 показана архітектура учбового лабораторного макету на основі автоматизованого модуля управління макетом промислового маніпулятора.

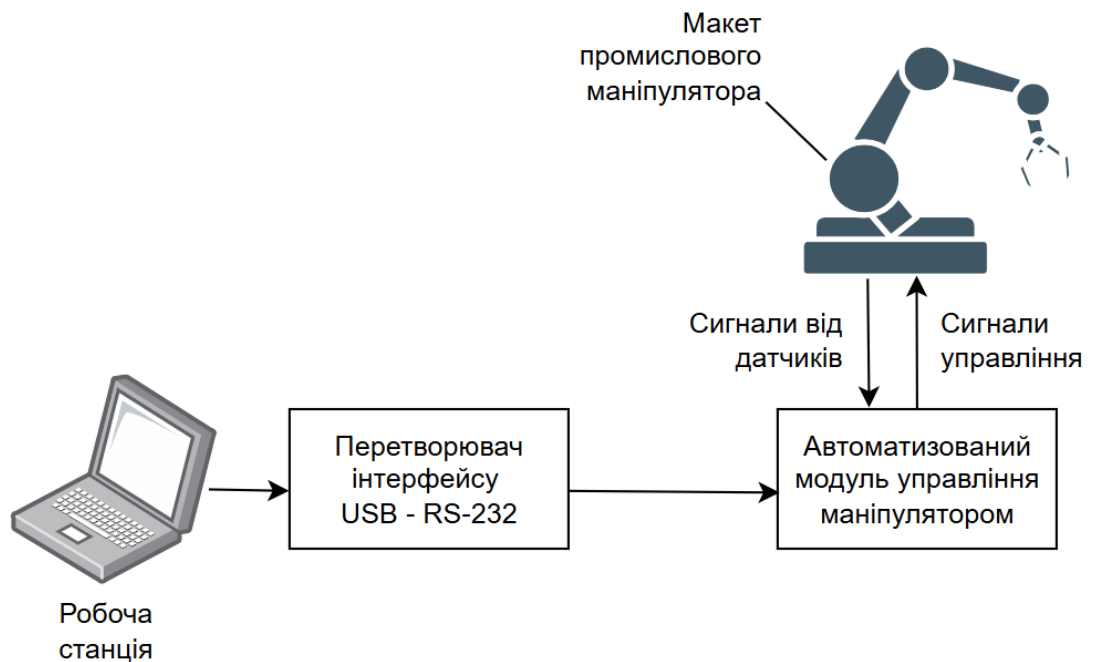


Рисунок 2.1 – Архітектура учбового лабораторного макету на основі автоматизованого модуля управління макетом промислового маніпулятора

Як можна бачити з рисунку 2.1, в основі системи знаходиться робоча станція – персональний комп'ютер або ноутбук, на якому встановлено спеціалізоване програмне забезпечення. Дане програмне забезпечення дає

змогу формувати команди керування маніпулятором, а також отримувати й обробляти дані від датчиків, встановлених на макеті.

Для зв'язку комп'ютера з модулем управління маніпулятором використовується перетворювач інтерфейсу USB – RS-232. Даний пристрій необхіден, тому що у сучасних комп'ютерів немає апаратної можливості підключатись до периферійних пристроїв напряму через COM-порт. Перетворювач інтерфейсу забезпечує конвертацію сигналу між USB-портом робочої станції та послідовним інтерфейсом RS-232, який використовується в модулі керування маніпулятором.

Автоматизований модуль управління маніпулятором побудовано у вигляді електронного блоку на базі мікроконтролера або програмованого логічного контролера (ПЛК). Модуль приймає команди керування від робочої станції через інтерфейс RS-232, обробляє їх і формує сигнали управління для мехатронних модулів макета промислового маніпулятора. Крім того, він здійснює збір сигналів з датчиків, встановлених на маніпуляторі, аналізує їх і приймає рішення про подальше управління рухомими ланками лабораторного макету.

Макет промислового маніпулятора є фізичною моделлю реального пристрою і оснащений виконавчими механізмами, такими як кроковими двигунами із сервоприводами, а також набором кінцевих датчиків (датчики положення та кінцеві вимикачі). Макет призначений для демонстрації принципів роботи промислових маніпуляторів і для практичного відпрацювання алгоритмів їхнього керування.

Таким чином, архітектура учбового лабораторного макета являє собою модульну систему з функціонально-закінченими пристроями. Вона забезпечує двосторонній обмін даними між робочою станцією та автоматизованим модулем керування, використовує стандартизовані інтерфейси для під'єднання частин макету і дає змогу моделювати умови роботи реальних промислових маніпуляторів у навчальних цілях.

2.2 Розроблення структурної схеми автоматизованого модуля управління макетом промислового маніпулятора

На рисунку 2.2 показана структурна схема автоматизованого модуля управління макетом промислового маніпулятора.

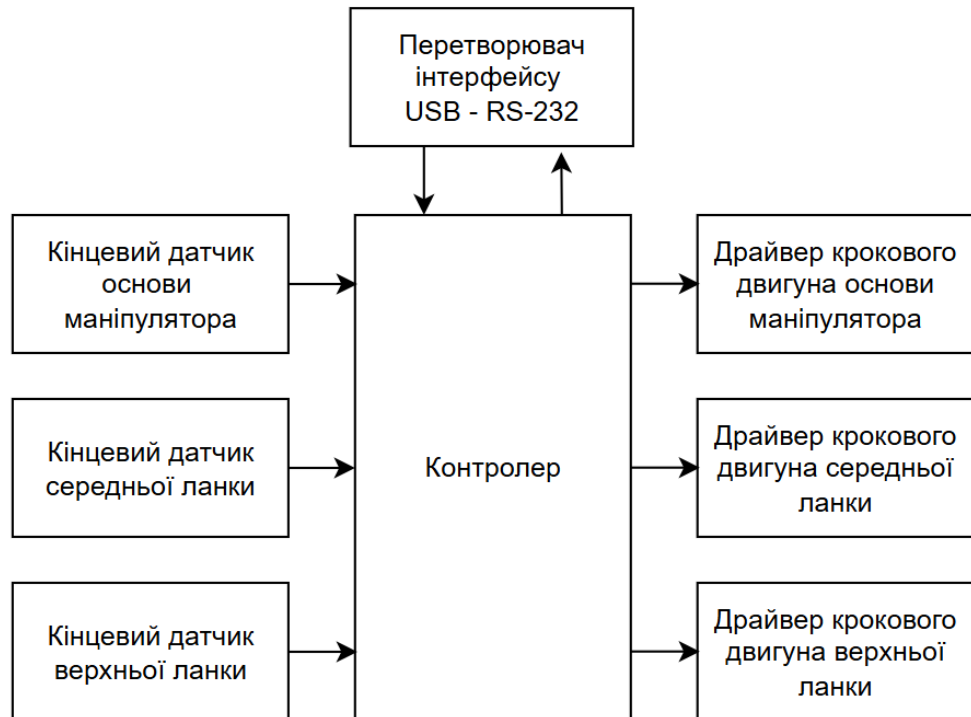


Рисунок 2.2 – Структурна схема автоматизованого модуля управління макетом промислового маніпулятора

Центральним елементом даної схеми є контролер, наприклад Arduino UNO. Даний пристрій виконує функції приймання керівних команд через інтерфейс RS-232, оброблює інформацію від кінцевих датчиків положення та виконує генерацію керованих сигналів для драйверів крокових двигунів.

Перетворювач інтерфейсу є посередником між персональним комп'ютером і даним контролером.

Для відстеження положення маніпулятора використовуються три кінцеві датчики, кожен з яких має підключення до контролера:

- кінцевий датчик основи маніпулятора призначений для визначення крайнього положення обертової основи маніпулятора;
- кінцевий датчик середньої секції фіксує досягнення крайнього нижнього положення середньої ланки маніпулятора;
- кінцевий датчик верхньої секції відстежує положення самої верхньої частини маніпулятора та також використовується при калібруванні для визначення початкового положення.

Дані датчики забезпечують контролеру зворотний зв'язок про початковий стан ланок маніпулятора, що дає змогу реалізувати алгоритм автоматичного калібрування та попередження виходу ланок за межі допустимих механічних положень і коректну зупинку приводів на основі крокових двигунів.

Керування рухом маніпулятора здійснюється за допомогою трьох драйверів крокових двигунів:

- драйвер крокового двигуна основи маніпулятора керує обертанням основи (бази маніпулятора);
- драйвер крокового двигуна середньої секції забезпечує переміщення відповідної частини маніпулятора в кутових координатах;
- драйвер крокового двигуна верхньої секції забезпечує досягнення маніпулятором заданої координати в декартовій системі сітці координат.

Контролер подає на драйвери відповідні сигнали керування:

- дозвіл руху;
- напрямок руху;
- управління поточним кроком.

Таким чином, набір з трьох основних сигналів дозволяє визначити напрямок обертання і кількість кроків, необхідних для досягнення необхідного положення ланками маніпулятора. Драйвери, своєю чергою, підсилюють сигнали до рівнів, необхідних для живлення крокових двигунів.

2.3 Опис конструкції макету промислового маніпулятора

Зовнішній вигляд макету промислового маніпулятора показано на рисунку 2.3.



Рисунок 2.3 – Зовнішній вигляд макету промислового маніпулятора

Як можна бачити з наведеного рисунку, до складу макету промислового маніпулятора входять:

- поворотна основа маніпулятора;
- середня ланка маніпулятора;
- верхня ланка маніпулятора;
- електричний привід основи маніпулятора (на рисунку 2.3 не показано);
- електричний привід середньої ланки маніпулятора;

- електричний привід верхньої ланки маніпулятора;
- платформа для встановлення виконавчого пристрою.

Довжина середньої ланки становить 1 метр від нижньої обертової вісі до верхнього шарніру. Довжина верхньої ланки становить 70 сантиметрів від верхнього шарніру до точки кріплення платформи виконавчого пристрою.

На рисунку 2.4 докладно показано механізм забезпечення руху ланок макету промислового маніпулятора.



Рисунок 2.4 – Механізм забезпечення руху ланок макету промислового маніпулятора

Механізм забезпечення руху ланок макету промислового маніпулятора складається з наступних компонентів:

- шестерня приводу середньої ланки маніпулятора;
- шестерня приводу верхньої ланки маніпулятора;
- кроковий двигун середньої ланки маніпулятора;
- кроковий двигун верхньої ланки маніпулятора;
- кроковий двигун поворотної бази маніпулятора;

- важіль тяги верхньої ланки маніпулятора;
- обертова основа маніпулятора.

Конструкція дволанкового ангулярного маніпулятора з нижнім розташуванням двигунів являє собою механічну систему, призначену для виконання переміщень у просторі з використанням двох послідовно з'єднаних ланок [11, 12]. Кінематична схема маніпулятора показана на рисунку 2.5.

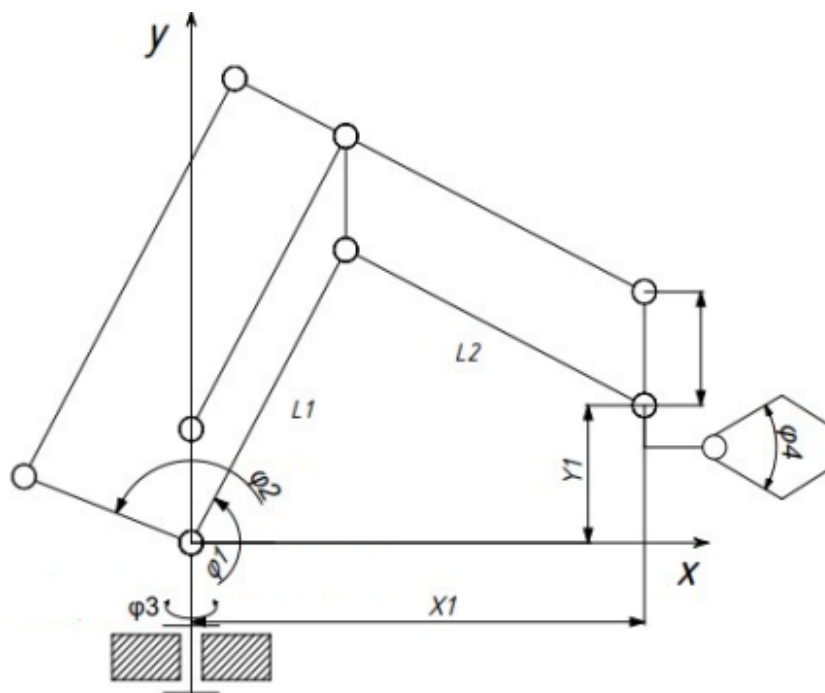


Рисунок 2.5 – Кінематична схема маніпулятора

Особливість цієї конструкції полягає в тому, що всі приводи (двигуни) розміщуються в нижній частині маніпулятора – в основі, що дає змогу знизити навантаження на конструктивні елементи ланок і підвищити загальну стійкість системи.

Основа маніпулятора являє собою масивну нерухому платформу, на якій закріплені два крокові або сервоприводи. Перший двигун відповідає за поворот нижньою ланки щодо основи. Обертання здійснюється навколо

вертикальної осі, що дає змогу першій ланці переміщатися в горизонтальній площині [13].

Другий двигун встановлений також у основі, та через систему передачі руху (через ремінну передачу) він приводить у рух другу ланку, яка шарнірно з'єднана з кінцем першої ланки. Таким чином, забезпечується управління кутом між першою і другою ланкою без розміщення двигуна безпосередньо на рухомій частині маніпулятора. Це суттєво знижує масу другої ланки та зменшує моменти інерції, таким чином підвищується точність позиціонування і швидкість переміщення ланок маніпулятора.

Кожна ланка виконана у вигляді жорсткої конструкції, виготовленої з металевої трубки діаметром 27 мм. У місцях з'єднання ланок встановлені підшипники, що забезпечують плавність руху і мінімальні втрати енергії на тертя.

Для калібрування крайніх положень в конструкції передбачені кінцеві вимикачі, встановлені в базових точках траєкторії руху ланок.

Маніпулятор має управлятись за допомогою контролера, який за заданими програмними алгоритмами розраховує необхідні кути повороту кожної ланки для досягнення потрібного положення кінцевого елемента (захоплення).

Принцип організації горизонтальної стабілізації платформи для встановлення виконавчого пристрою показано на рисунку 2.6.

Принцип роботи конструкції горизонтальної стабілізації платформи для встановлення виконавчого пристрою на основі тяг полягає в тому, щоб незалежно від положення несучої конструкції маніпулятора підтримувати платформу в горизонтальному положенні. Це досягається за допомогою системи механічних тяг, які автоматично компенсують нахил опорних елементів під час їх переміщення.

Основу конструкції становить закріплена на рухомій верхній ланці маніпулятора платформа, яка з'єднана з базою через набір шарнірних тяг. Ці тяги працюють за принципом паралелограма: зміни кутового положення

основи автоматично врівноважуються завдяки взаємоузгодженому руху тяг, що дає змогу підтримувати платформу в одному і тому ж просторовому положенні [13, 14].



Рисунок 2.6 – Принцип організації горизонтальної стабілізації платформи для встановлення виконавчого пристрою

До двох боків платформи прикріплюється дві тяги, які з'єднуються з опорною конструкцією через трикутну пластину (рис. 2.5, 2.6). При цьому важливо, щоб довжина тяг і точки їхнього кріплення були підібрані таким чином, щоб за будь-якого переміщення або нахилу основи результуюче положення платформи залишалося горизонтальним. Усі з'єднання виконуються за допомогою шарнірних зчленувань (рис. 2.6).

Коли ланки маніпулятора рухаються та змінюють кут нахилу, тяги, в свою чергу, змінюють власні кути відносно платформи та основи, але за рахунок жорстких геометричних зв'язків кути компенсуються.

Таким чином, платформа зі встановленим на ній виконавчим пристроєм (захопленням маніпулятора) залишається стабільно орієнтованою в просторі, незалежно від динаміки руху носія.

2.4 Розроблення алгоритму роботи автоматизованого модуля керування макетом промислового маніпулятора

На рисунку 2.7 показано алгоритм роботи автоматизованого модуля управління макетом промислового маніпулятора.

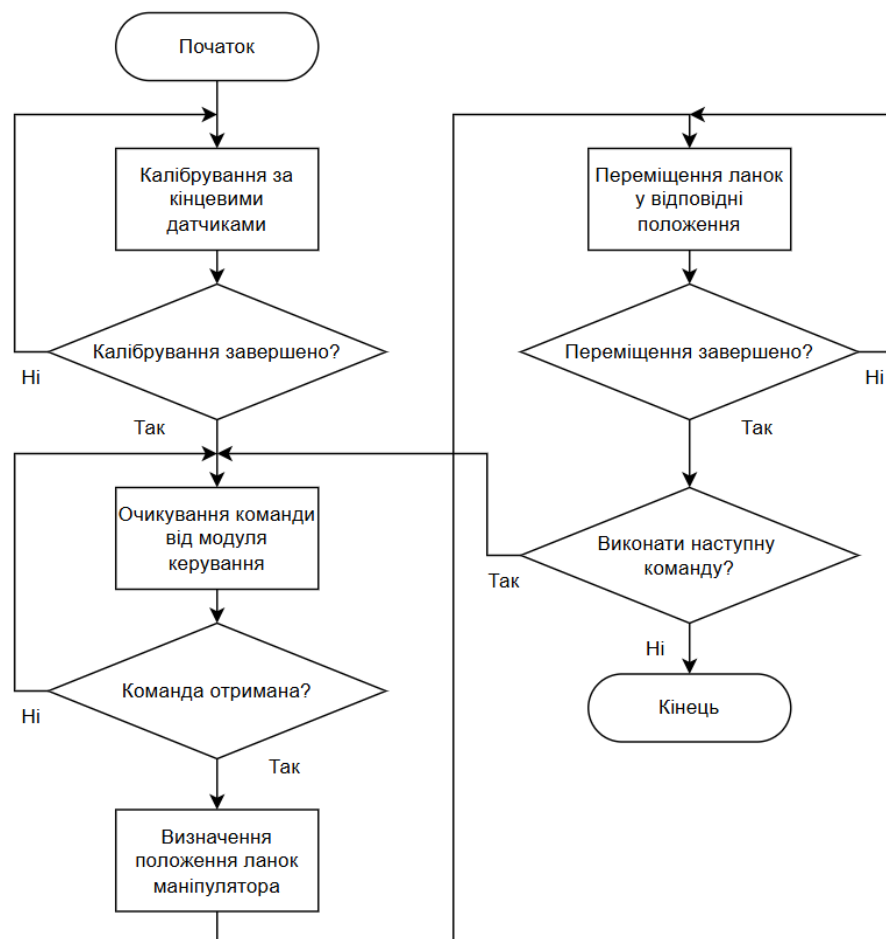


Рисунок 2.7 – Алгоритм роботи автоматизованого модуля управління макетом промислового маніпулятора

Даний алгоритм забезпечує циклічну обробку команд з обов'язковою початковою калібрувкою положення маніпулятора, що гарантує точність позиціонування виконавчих елементів. Алгоритм враховує необхідність перевірки всіх етапів на успішність перед переходом до наступного етапу.

На початку роботи алгоритму відбувається первинний запуск програми та ініціалізація початкового стану змінних. Далі система переходить до етапу калібрування. На цьому етапі проводиться калібрування всіх ланок маніпулятора за допомогою кінцевих датчиків, які визначають початкове (нульове) положення механічних частин. Якщо калібрування ще не завершено, процес продовжується до моменту успішного визначення всіх вихідних позицій.

Після завершення калібрування модуль управління переходить у режим очікування команди від зовнішнього модуля керування в якості якого виступає персональний комп'ютер. Поки команда не отримана, система залишається у стані очікування. При надходженні команди виконується аналіз її змісту, визначення цільового положення кожної з ланок маніпулятора, включно з обертанням базової платформи.

Далі система переходить до етапу переміщення. Відповідно до отриманої команди здійснюється рух ланок маніпулятора у задані позиції. Поки переміщення не завершено, система продовжує керувати процесом руху. Після завершення переміщення перевіряється, чи необхідно виконати ще одну команду. Якщо так, процес повертається до етапу очікування наступної команди та її обробки. Якщо ж більше немає команд для виконання, алгоритм завершується.

3 ВИБІР КОМПОНЕНТІВ ДЛЯ РЕАЛІЗАЦІЇ АВТОМАТИЗОВАНОЇ СИСТЕМИ

3.1 Вибір контролера

Для керування роботою автоматизованої системи необхідно обрати модель контролера, що буде становити основу всієї системи. Управління маніпулятором можливе як із застосуванням контролера типу Arduino, так і на базі промислового контролера (ПЛК). Проведемо порівняльний аналіз цих двох варіантів.

Зовнішній вигляд контролеру Arduino Nano та призначення основних контактів на платі показано на рисунку 3.1.

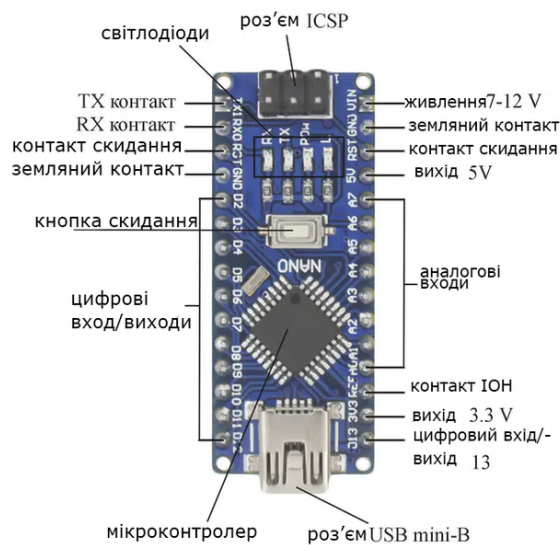


Рисунок 3.1 – Зовнішній вигляд контролеру Arduino Nano та призначення основних контактів на платі

Контролер Arduino використовується для розробки електронних пристроїв, які здатні приймати сигнали від цифрових та аналогових пристроїв підключених до неї, а також для управління різними виконавчими пристроями [15]. На платі встановлені всі необхідні для роботи з контролером

компоненти. Має невеликі розміри, функціональні можливості плати не відрізняються від інших версій, різниця тільки в збірці плати. Дана плата підходить для проектів, в яких важлива компактність.

Проекти, розроблені та працюють на платформі Arduino, можуть працювати самостійно або спільно з програмним забезпеченням на комп'ютері. Плата не вимагає складання, тільки під'єднайте її до комп'ютера, щоб почати роботу. Живлення плата може брати від USB-кабелю або від поданого живлення на входні контакти VIN, GND. У разі підключення декількох джерел живлення оптимальне вибирається платою автоматично.

На рисунку 3.2 наведена схема електрична принципова контролера Arduino на базі мікроконтролера ATmega328P.

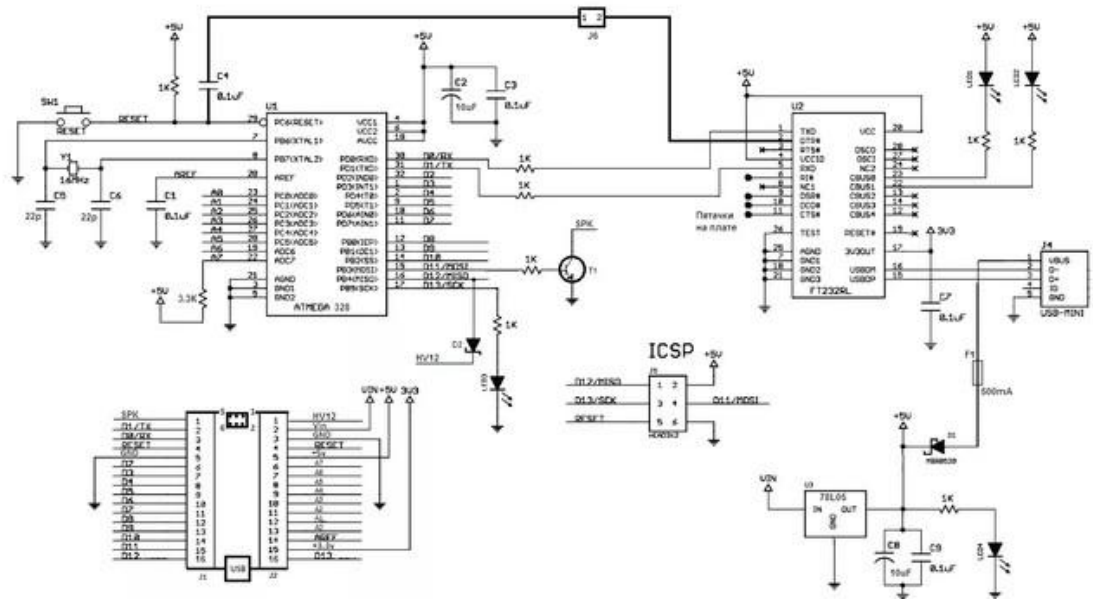


Рисунок 3.2 – Схема електрична принципова контролера Arduino на базі мікроконтролера ATmega328P

В якості керуючого контролеру також можна застосувати програмований логічний контролер (ПЛК), наприклад фірми AQTECK. Таким контролером може бути ПЛК100. Контролер призначено для використання у

складі різних автоматизованих систем контролю і керування на промислових підприємствах. Контролер може керувати:

- виділеними локальними об'єктами;
- локальним об'єктом у складі комплексної інформаційної мережі;
- групою локальних об'єктів у складі комплексної інформаційної мережі.

Логіка роботи пристрою програмується за допомогою ПЗ CODESYS v2.3. Підтримуються всі мови програмування стандарту IEC 61131-3:

- LD – релейно-контакторні схеми;
- FBD – функціональні блокові діаграми;
- SFC – послідовні функціональні діаграми;
- ST – структурований текст;
- IL – список інструкцій.

Зовнішній вигляд контролеру ПЛК110-60 показано на рисунку 3.3.

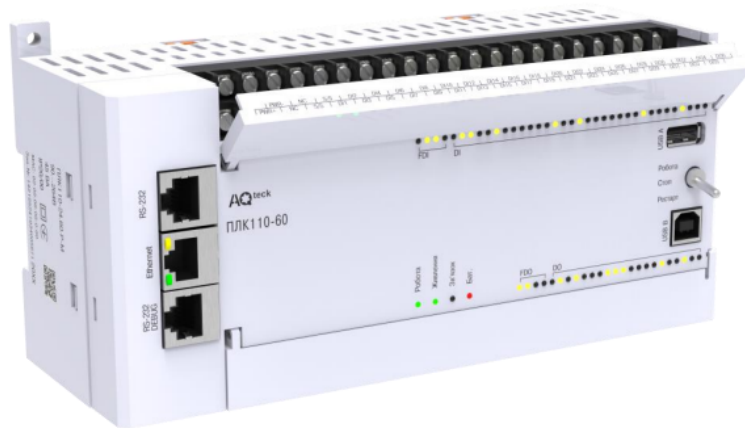


Рисунок 3.3 – Зовнішній вигляд контролеру ПЛК110-60

Таким чином, контролер Arduino Nano і програмований логічний контролер ОВЕН ПЛК110-60 є представниками різних класів пристроїв, призначених для керування автоматизованими системами.

Arduino Nano – це модуль, що побудовано на мікроконтролері ATmega328P. Він призначений для простих автоматизованих завдань,

прототипування та освітніх цілей. Своєю чергою, ОВЕН ПЛК110-60 – це промисловий ПЛК, призначений для тривалої та надійної роботи у складі, на виробництві. Контролер ОВЕН призначений для використання в реальному промисловому середовищі з підвищеними вимогами до надійності, електричної безпеки та сумісності з промисловими протоколами.

Arduino Nano має 14 цифрових входів/виходів, з яких 6 можуть використовуватися як ШІМ-виходи, а також 8 аналогових входів. Він живиться від USB або зовнішнього джерела 7-12 В.

ОВЕН ПЛК110-60 пропонує набагато ширший вибір інтерфейсів і модифікацій: до 60 дискретних і аналогових входів/виходів залежно від конфігурації, наявність портів RS-485, Ethernet, CAN, а також підтримку промислових протоколів (Modbus RTU, TCP/IP, OPC та ін.). Це робить його придатним для інтеграції в складні промислові системи.

Arduino Nano програмується за допомогою середовища Arduino IDE, що використовує мову, засновану на C/C++. Це зручно для швидкого прототипування.

ОВЕН ПЛК110-60 програмується в середовищі Codesys, відповідному стандарту IEC 61131-3, що підтримує мову LAD, FBD, ST, IL та інші. Це дає змогу створювати структуровані, стандартизовані рішення, які легко підтримувати на виробництві.

Arduino не призначена для роботи в промисловому середовищі – він не має захисту від електромагнітних завад, нестабільного живлення або промислових рівнів сигналів.

ПЛК ОВЕН має відповідні сертифікати, підтримує гальванічну розв'язку, а також призначений для безперервної роботи 24/7 з високим ступенем надійності.

Arduino Nano має низьку вартість і доступний для купівлі або заміни. ОВЕН ПЛК110-60 – це професійне промислове рішення, яке коштує значно дорожче, але й надає незрівнянно більший рівень функціональності та стабільності.

Таким чином, для вирішення задачі створення автоматизованого модуля управління макетом промислового маніпулятора обираємо модуль Arduino Nano з відповідними модулями розширення.

3.2 Плата розширення для контролера Arduino Nano

Для підключення до периферійних пристроїв, наприклад, драйверів крокових двигунів будемо використовувати плату розширення Arduino Nano V3 (рис. 3.4).

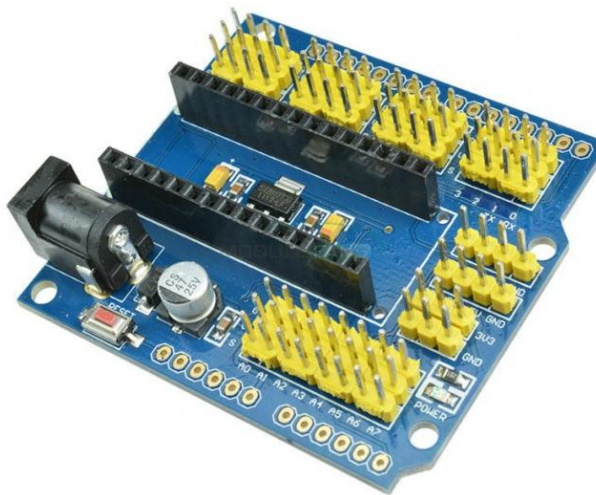


Рисунок 3.4 – Плата розширення Arduino Nano V3

Плата розширення виводів для Arduino Nano V3 призначена для прототипування або простого з'єднання між Arduino Nano та іншими пристроями.

Крім стандартних виводів контролера є кілька груп контактів додаткового призначення – шини живлення 3,3 В і загальний, інтерфейс UART (4 групи), а також інтерфейс I2C (TWI). Всі інтерфейсні роз'єми доповнені контактами живлення. На платі встановлений додатковий

стабілізатор напруги 5 В 450 мА і стандартний роз'єм підключення зовнішнього живлення.

Характеристики модуля розширення:

- сумісність з контролерами: Arduino Nano;
- вхідна напруга стабілізатора напруги: від 7,5 В до 12 В;
- вихідна напруга стабілізатора напруги: 5 В;
- максимальний струм навантаження стабілізатора: 450 мА.

3.3 Вибір драйвера крокового двигуна

3.3.1 Цифровий драйвер крокового двигуна DM556D 5,6 А

Цифровий драйвер крокового двигуна DM556D розроблений на основі сучасних технологій управління цифрових сигнальних процесорів [15]. Високі експлуатаційні якості драйвера засновані потужної технології управління 32-розрядним процесором. Зовнішній вигляд драйверу показаний на рисунку 3.5.

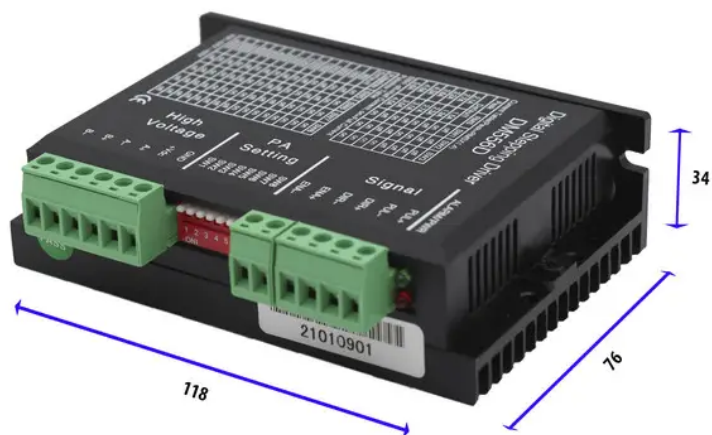


Рисунок 3.5 – Зовнішній вигляд драйверу DM556D 5,6 А

Цифровий драйвер володіє стабільними робочими характеристиками, працює з низьким рівнем шуму, вібрації, температури; мікрокрок має 16

режимів роботи. Для кокового двигуна з кроком $1,8^\circ$ можна досягти максимум 51200 імпульсів на оберт (1/256). Додаткові характеристики пристрою:

- володіє технологією придушення резонансу;
- низьке підвищення температури значно покращує показники крокових двигунів.

Драйвер має таку функцію як самоналаштування онлайн, тому він може без ручних налаштувань автоматично налаштувати необхідні й оптимальні параметри для різних двигунів і забезпечить високу продуктивність.

Напруга живлення 24-50 В, підходить для управління двигунами з максимальним робочим струмом до 5,6 А (можна налаштувати від 2,1 А до 5,6 А).

Драйвер DM556D широко застосовується в обладнанні яке вимагає низький рівень вібрації і шуму, використовується для автоматизації маркувального, пакувального, топографічного обладнання, різних вузлів верстатів з ЧПК (ЧПУ).

Особливості і характеристики драйверу DM556D:

- технологія самоналаштування в режимі реального часу;
- оптично ізольовані вхідні/вихідні сигнали;
- напруги живлення 24-50 В;
- частота 200 кГц;
- робочий струм від 2,1 А до 5,6 А;
- мікрокрок (16 режимів роботи, максимум 51200 імпульсів на оберт (1/256));
- захист від перевантаження по струму і напрузі;
- низьке підвищення температури, плавна робота двигуна;
- розміри: 118 мм x 76 мм x 34 мм;
- вага: 300 г.

3.3.2 Драйвер крокового двигуна BL-TB6600-V1.2

Драйвер крокового двигуна BL - TB6600-V1.2 створений на мікросхемі TB6600HG, це вдосконалена версія раніше популярного драйвера BL-TB6560-V2.0 [16]. Зовнішній вигляд драйверу BL-TB6600-V1.2 показано на рисунку 3.6.

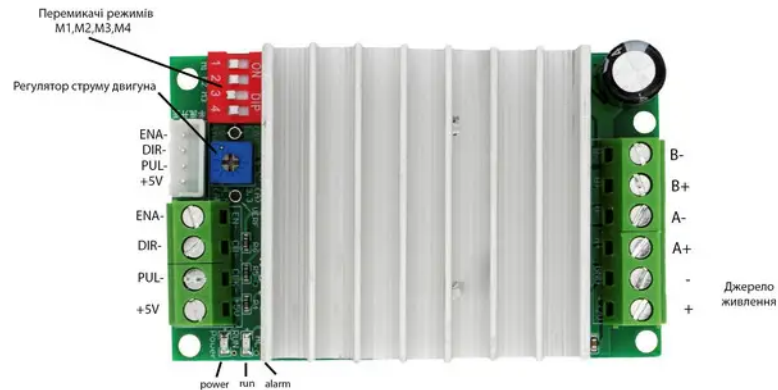


Рисунок 3.6 – Зовнішній вигляд драйверу BL-TB6600-V1.2

Драйвер підходить для керування біполярними та уніполярними кроковими двигунами з робочим струмом до 4,5 А. Якщо порівняти з попередньою версією драйвера TB6560AHG, то вихідна потужність у TB6600HG виросла до 200 Вт, можна помітити чимале подвоєння потужності. Крім цього збільшилася максимальна робоча частота, були додані схеми захисту.

Пристрій має інтелектуальне керування струмом і використовує модуляцію PWM, що дає змогу під'єднувати кроковий двигун будь-якої напруги та струм живлення до драйвера – на фіксованому рівні; сама мікросхема обмежує максимальний струм.

За співвідношенням ціна-якість драйвера BL – TB6600 широко використовується як для аматорських систем, так і для професійних систем ЧПК.

Характеристики драйверу BL-TB6600-V1.2:

- вхідна напруга: від 8 В до 50 В (рекомендовано 32 В);
- вихідний струм: від 0,3 А до 4,35 А (пік. 5 А);
- регулювання макс. струму крокового двигуна: 1,1 А; 2,2 А; 3,3 А; 4,5 А;
- макс. частота PWM (ШИМ): 200 кГц;
- мікрокрок: 1; 2; 8; 16;
- регулювання утримання струму: 50% або 100%;
- наявність захисту від замикання;
- робоча температура: -10 °С до +45 °С;
- габаритні розміри: 80 мм × 50 мм × 29 мм;
- вага: 119 г.

3.3.3 Порівняльний аналіз драйверів крокового двигуна

Контролери крокових двигунів DM556D і BL-TB6600-V1.2 широко застосовуються в системах числового програмного керування (ЧПК), автоматизації та робототехніці. Обидва пристрої призначені для керування двофазними кроковими двигунами, проте різняться за технічними характеристиками та функціональністю. У таблиці 3.1 представлено порівняння цих драйверів.

Таблиця 3.1 – Порівняльний аналіз драйверів крокового двигуна DM556D і BL-TB6600-V1.2

Параметр	DM556D	BL-TB6600-V1.2
Максимальний струм	До 5,6 А (піковий)	До 4,5 А (піковий)
Діапазон напруги	24–50 В	8–32 В DC
Мікрокроки	До 1/256 (макс. 51200 кроків/оберт)	До 1/16 (1, 2, 8, 16)
Частота імпульсів	До 200 кГц	До 200 кГц
Тип керування	PUL/DIR або CW/CCW	PUL/DIR

Продовження таблиці 3.1

Параметр	DM556D	BL-TB6600-V1.2
Захисти	Від перенапруги, зниженої напруги, перевантаження, перегріву	Від короткого замикання, перегріву
Додаткові функції	Автоматичне зниження струму, самотестування, оптоізоляція входів	Інтелектуальне керування струмом, регулювання утримуючого струму
Габарити	118 мм × 75,5 мм × 34 мм	80 мм × 50 мм × 29 мм
Вага	Близько 300 г	Близько 119 г
Ціна	1872 грн.	362 грн.

DM556D рекомендований для систем, що потребують високої точності, низького рівня шуму та підтримки потужних крокових двигунів. Це оптимальний вибір для професійних ЧПК-систем, фрезерних верстатів, граверів та інших промислових застосувань. BL-TB6600-V1.2 підходить для навчальних проєктів, аматорських розробок та малопотужних автоматизованих систем, де важливі компактність і бюджет.

Таким чином, в нашому проєкті будемо застосовувати другий варіант – драйвер крокового двигуна BL-TB6600-V1.2.

4 РОЗРОБКА ПРОГРАМИ РОБОТИ БЛОКА УПРАВЛІННЯ ПРОМИСЛОВИМ МАНІПУЛЯТОРОМ

4.1 Розроблення програми для мікроконтролера

Програма для мікроконтролера має в своєму складі файл конфігурації `config.h`. Для правильного функціонування потрібно внести ряд змін в цей файл. Одним з налаштувань є розміри ланок маніпулятора (рис. 4.1). Ці розміри використовуються при обчисленні кута повороту приводів маніпулятора для досяжності заданої координати в просторі.

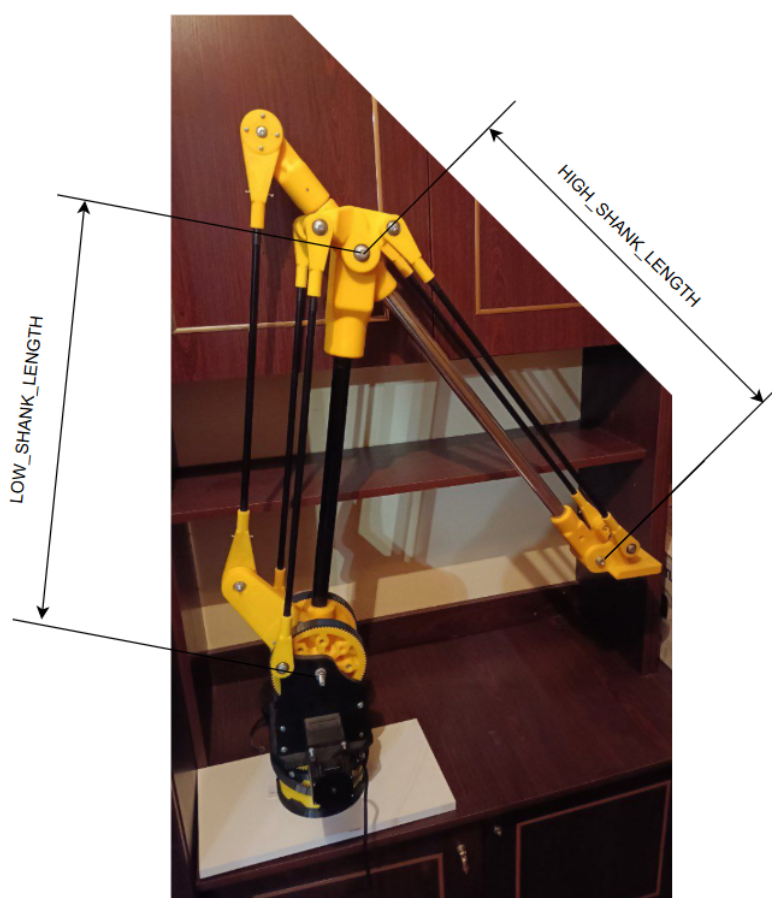


Рисунок 4.1 – Розміри ланок маніпулятора

Ланка маніпулятора LOW_SHANK_LENGTH має довжину 1000 мм. Довжина верхньої ланки HIGH_SHANK_LENGTH дорівнює 700 мм. Виходячи з цих розмірів записуємо такі параметри в файл конфігурації config.h:

```
//ROBOT ARM LENGTH
#define LOW_SHANK_LENGTH 1000.0
#define HIGH_SHANK_LENGTH 700.0
```

Для точного визначення координат в які має потрапити робочий інструмент необхідно визначити відстань від кріплення основи інструменту до центру робочого органу.

На рисунку 4.2 показана конструкція платформи для кріплення інструменту.

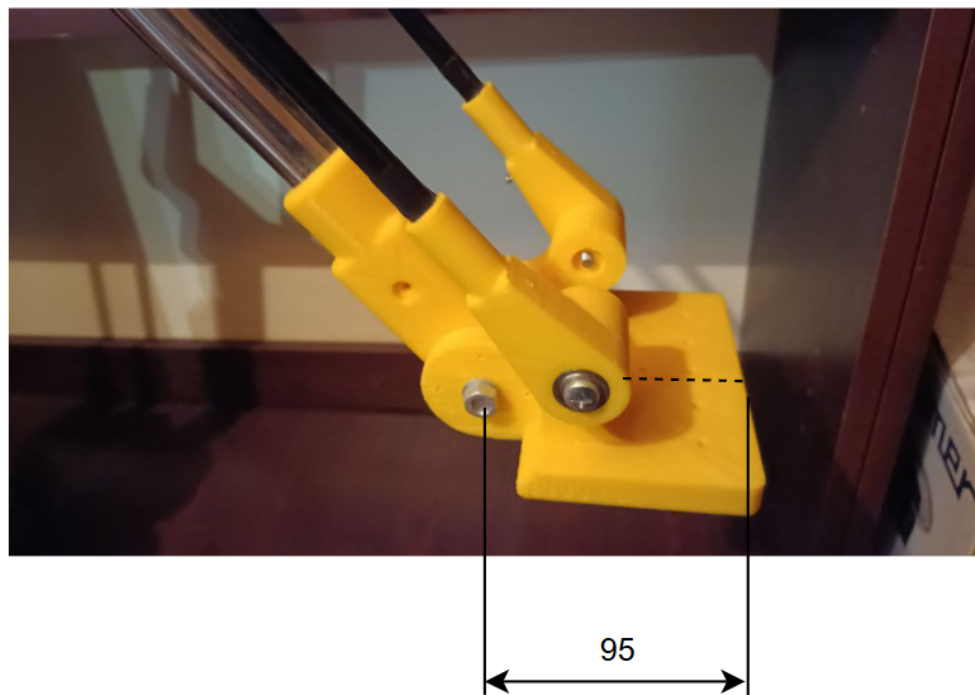


Рисунок 4.2 – Конструкція платформи для кріплення інструменту

Після модифікація рядок налаштування параметрів робочого інструменту буде мати наступний вигляд:

```
#define END_EFFECTOR_OFFSET 95.0 // LENGTH FROM UPPER
SHANK BEARING TO MIDPOINT OF END EFFECTOR IN MM
```

Для налаштування швидкості обміну повідомленнями між блоком управління і програмою ПК або пультом керування необхідно змінити конфігурацію в наступному вигляді:

```
//SERIAL SETTINGS
```

```
#define BAUD 115200
```

Для визначення кутів повороту рухомих суглобів розроблені декілька функцій. Наприклад, функція «void Interpolation::setInterpolation(Point p0, Point p1, float av)» призначена для ініціалізації параметрів при виконанні задачі лінійної інтерполяції (рис. 4.3).

```
void Interpolation::setInterpolation(Point p0, Point p1, float av) {
    v = av; //mm/s

    float a = (p1.xmm - p0.xmm);
    float b = (p1.ymm - p0.ymm);
    float c = (p1.zmm - p0.zmm);
    float e = abs(p1.emm - p0.emm);
    float dist = sqrt(a*a + b*b + c*c);

    if (dist < e) {
        dist = e;
    }

    if (v < 5) { //includes 0 = default value
        v = sqrt(dist) * 10; //set a good value for v
    }
    if (v < 5) {
        v = 5;
    }

    tmul = v / dist;

    xStartmm = p0.xmm;
    yStartmm = p0.ymm;
    zStartmm = p0.zmm;
    eStartmm = p0.emm;

    xDelta = (p1.xmm - p0.xmm);
    yDelta = (p1.ymm - p0.ymm);
    zDelta = (p1.zmm - p0.zmm);
    eDelta = (p1.emm - p0.emm);

    state = 0;

    startTime = micros();
}
```

Рисунок 4.3 – Функція ініціалізації лінійної інтерполяції

Лінійна інтерполяція – це метод знаходження проміжних значень між двома відомими точками шляхом побудови прямої лінії між ними. В даному випадку це визначення траєкторії руху маніпулятора між двома точками у просторі – p_0 (початкова точка) та p_1 (кінцева точка) – з урахуванням заданої швидкості.

Лінійна інтерполяція дозволяє знайти будь-яку проміжну точку $P(t)$ між точками $P_0(x_0, y_0)$ та $P_1(x_1, y_1)$, де $t \in [0, 1]$, за формулою:

$$P(t) = (1 - t) \cdot P_0 + t \cdot P_1. \quad (4.1)$$

Якщо брати в розрахунок координати точок P_0 і P_1 то координати проміжної точки можуть бути розраховані за формулами:

$$x(t) = x_0 + t \cdot (x_1 - x_0), \quad (4.2)$$

$$y(t) = y_0 + t \cdot (y_1 - y_0). \quad (4.3)$$

На початку роботи функції записується значення поточної швидкості в змінну v (у мм/с).

Далі обчислюється відстань у тривимірному просторі між точками P_0 і P_1 з координатами x_{mm} , y_{mm} , z_{mm} , а також модуль зміни додаткової координати e_{mm} . Якщо ця додаткова відстань e більша за геометричну відстань $dist$, то за основу береться саме e .

Якщо швидкість v не задана або дуже мала (менша 5 мм/с), то вона розраховується як корінь з відстані помножений на 10, і знову ж таки перевіряється, щоб вона не була меншою за 5 мм/с.

Далі обчислюється коефіцієнт $tmul$, який використовується для визначення, як швидко повинна відбуватись зміна координат у часі.

Початкові координати зберігаються в окремих змінних ($x_{Startmm}$, $y_{Startmm}$ і т.д.), а також обчислюються дельти зміщень для кожної координати між P_0 і P_1 .

Поточний момент часу зберігається в $startTime$ (у мікросекундах). Він використовується для визначення моменту з якого починається виконання інтерполяції.

Фрагмент функції `updateActualPosition` показано на рисунку 4.4.

```
void Interpolation::updateActualPosition() {
    if (state != 0) {
        return;
    }
    long microsek = micros();
    float t = (microsek - startTime) / 1000000.0;
    float progress;
    switch (SPEED_PROFILE){
        // FLAT SPEED CURVE
        case 0:
            progress = t * tmul;
            if (progress >= 1.0){
                progress = 1.0;
                state = 1;
            }
            break;
        // ARCTAN APPROX
        case 1:
            progress = atan((PI * t * tmul) - (PI * 0.5)) * 0.5 + 0.5;
            if (progress >= 1.0) {
                progress = 1.0;
                state = 1;
            }
            break;
        // COSIN APPROX
        case 2:
            progress = -cos(t * tmul * PI) * 0.5 + 0.5;
            if ((t * tmul) >= 1.0) {
                progress = 1.0;
                state = 1;
            }
            break;
    }
}
```

Рисунок 4.4 – Фрагмент функції `updateActualPosition`

Функція `updateActualPosition()` відповідає за оновлення поточної позиції маніпулятора під час виконання інтерпольованого руху. Вона викликається регулярно (у циклі або таймері) і поступово просуває

маніпулятор від початкової точки до кінцевої по обраній швидкісній траєкторії.

На початку роботи перевіряється стан змінної state. Якщо він не дорівнює 0, тобто інтерполяція вже завершена або неактивна, функція завершує свою роботу. В іншому випадку визначається час, який минув з початку руху (t), і розраховується прогрес переміщення – число від 0 до 1, що вказує, яку частину шляху вже пройдено.

Розрахунок прогресу залежить від профілю швидкості, який визначається змінною SPEED_PROFILE. Параметр 0 означає, що буде використовуватись лінійна швидкість. Графік зміни лінійної швидкості показана на рисунку 4.5.

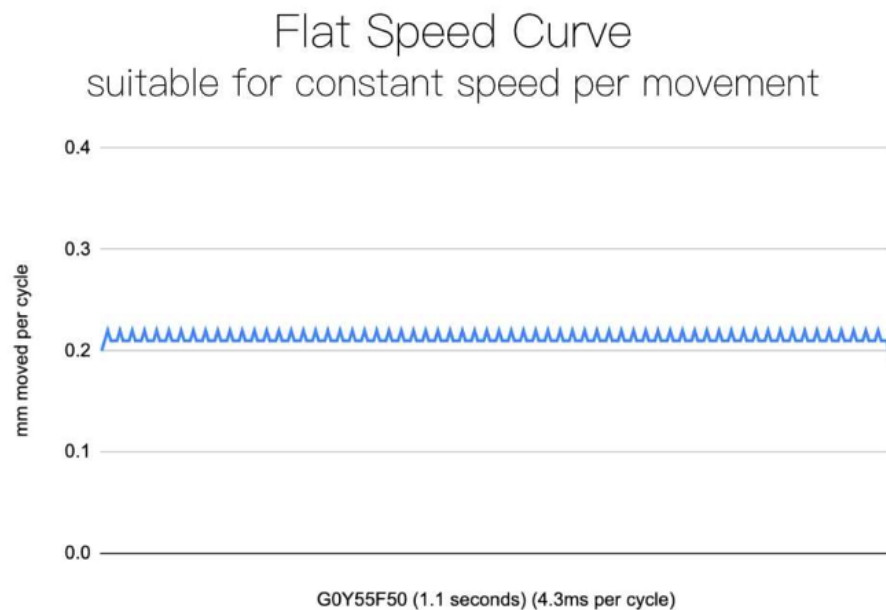


Рисунок 4.5 – Графік зміни лінійної швидкості

Профіль 1 відповідає поведінці, коли швидкість змінюється за арктангенсом. Це означає, що буде спостерігатися плавний старт і плавний фініш. Графік зміни швидкості за законом арктангенса показано на рисунку 4.6.

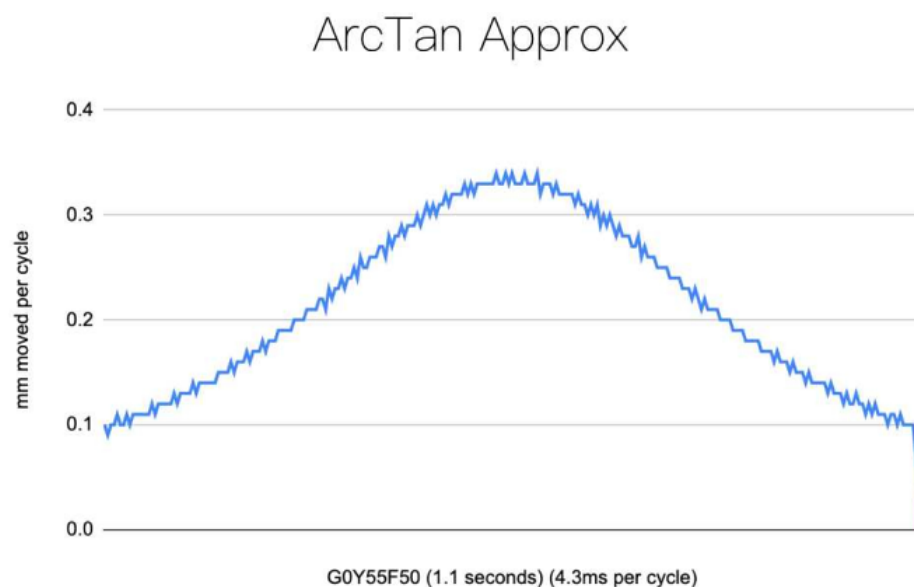


Рисунок 4.6 – Графік зміни швидкості за законом арктангенса

Параметр 2 змінної `SPEED_PROFILE` означає, що швидкість руху маніпулятора буде відповідати косинусоїду. Це означає, що буде застосовуватись ще плавніша зміна швидкості на початку і в кінці переміщення. Графік зміни швидкості за законом косинусу показано на рисунку 4.7.

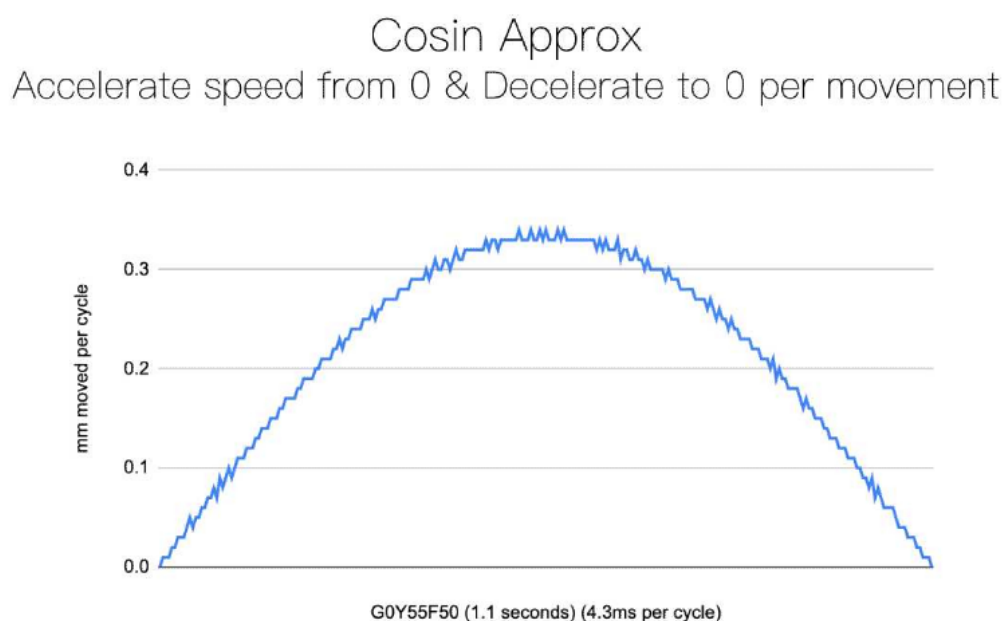


Рисунок 4.7 – Графік зміни швидкості за законом косинусу

Якщо будь-який із зазначених процесів переміщення досягає або буде перевищувати 1.0, це означає, що відбувається завершення руху. В даному випадку змінна state встановлюється в 1.

Далі обчислюється поточне положення кожної осі шляхом лінійної інтерполяції між стартовою позицією (xStartmm тощо) і зміщенням (xDelta, yDelta тощо). Отримане положення зберігається у масиві pos_tracker.

На рисунку 4.8 показано фрагмент функції updateActualPosition в якому обчислюється поточне положення кожної осі маніпулятора.

```
pos_tracker[X_AXIS] = xStartmm + progress * xDelta;
pos_tracker[Y_AXIS] = yStartmm + progress * yDelta;
pos_tracker[Z_AXIS] = zStartmm + progress * zDelta;
pos_tracker[E_AXIS] = eStartmm + progress * eDelta;

if(isAllowedPosition(pos_tracker)){
    xPosmm = pos_tracker[X_AXIS];
    yPosmm = pos_tracker[Y_AXIS];
    zPosmm = pos_tracker[Z_AXIS];
    ePosmm = pos_tracker[E_AXIS];
} else {
    pos_tracker[X_AXIS] = xPosmm;
    pos_tracker[Y_AXIS] = yPosmm;
    pos_tracker[Z_AXIS] = zPosmm;
    pos_tracker[E_AXIS] = ePosmm;
    state = 1;
    progress = 1.0;
    xStartmm = xPosmm;
    yStartmm = yPosmm;
    zStartmm = zPosmm;
    eStartmm = ePosmm;
    xDelta = 0;
    yDelta = 0;
    zDelta = 0;
    eDelta = 0;
}
```

Рисунок 4.8 – Фрагмент функції updateActualPosition в якому обчислюється поточне положення кожної осі маніпулятора

Також в функції реалізовано перевірку, чи ця визначена нова позиція допустима за допомогою функції isAllowedPosition().

Якщо так – нова позиція записується як актуальна (xPosmm, yPosmm і т.д.). Якщо ж позиція недопустима, система повертається до попередньої

позиції, зупиняє інтерполяцію, скидає зміщення і встановлює state = 1, що означає завершення руху.

4.2 Опис принципу обміну командами з модулем керування

В програмі управління маніпулятором передбачено декілька команд для здійснення обміну повідомленнями між блоком керування та пристроєм управління:

- Linear Move;
- Dwell;
- Homing;
- Absolute Mode;
- Relative Mode;
- Set Position;
- Set Speed Curve.

Команда Linear Move використовується для здійснення лінійного переміщення маніпулятора з поточної позиції в нову задану позицію з визначеною швидкістю.

Синтаксис команди має вигляд:

$$G0 X<x_mm> Y<y_mm> Z<z_mm> F<швидкість_мм/с> ,$$

де X, Y, Z – координати кінцевої точки в просторі (в міліметрах);
F – швидкість переміщення (в мм/с).

Наприклад, команда G0 X100 Y50 Z-20 F40 означає переміщення маніпулятора до координат X=100 мм, Y=50 мм, Z=-20 мм зі швидкістю 40 мм/с.

Якщо певна координата не вказана, то рух по відповідній осі не виконується. У режимі абсолютних координат (G90) значення X, Y, Z

вказують конкретну цільову точку в просторі. У відносному режимі (G91) ці значення означають зміщення від поточної позиції.

Команда Dwell (G4) використовується для створення паузи або затримки в програмі на задану кількість часу.

При роботі блока управління макетом промислового маніпулятора після виклику цієї команди маніпулятор призупиняє будь-які подальші дії на вказаний проміжок часу, залишаючись у поточному положенні.

Синтаксис команди:

G4 S<секунд>,

де S – це тривалість затримки в секундах (може бути дробовим числом, наприклад S0.5 – означає затримку на пів секунди).

Наприклад, G4 S5 означає зробити паузу на 5 секунд перед виконанням наступної команди.

Команда G4 корисна у ситуаціях, коли необхідно дочекатися завершення певного процесу перед переходом до наступного етапу – наприклад, коли маніпулятор має зачекати перед тим, як захопити або відпустити об'єкт, перед зміною інструменту тощо.

Приклад реалізації команди cmdDwell показано на рисунку 4.9.

```
void cmdDwell(Cmd(&cmd)){
    delay(int(cmd.valueS * 1000));
}
```

Рисунок 4.9 – Приклад реалізації команди cmdDwell

У внутрішній реалізації програми для мікроконтролера ця команда блокує виконання наступних інструкцій, поки не завершиться заданий інтервал часу, що відлічується з використанням мілісекундного таймера.

Команда Set Position (G92) використовується для встановлення поточного положення маніпулятора у певне значення координат без фізичного переміщення.

Дана команда дозволяє за допомогою пульта керування перевести ланки маніпулятора в потрібне положення та задати його як початкове. Таким чином, вона дозволяє вказати системі керування, що поточне положення дорівнює заданому значенню, навіть якщо фактично пристрій не рухався.

Синтаксис команди:

$$G92 X<x_mm> Y<y_mm> Z<z_mm> E<мм> ,$$

де X, Y, Z, E – це координати, які слід приписати поточному положенню відповідних осей.

Наприклад, команда G92 X100 Y50 встановлює поточну позицію по осі X в 100 мм, а по осі Y – у 50 мм. Це не призведе до руху маніпулятора, а лише оновить внутрішній стан координат.

Дана команда розроблена для калібрування, компенсації механічних зсувів, а також для ручного або програмного встановлення початкових точок перед початком рухів.

Наприклад, після фізичного переміщення маніпулятора до механічного обмежувача (наприклад, за допомогою кінцевих вимикачів), командою G92 можна зафіксувати, що це і є нульова точка.

Команда без параметрів, тобто просто G92, скидає всі зміщення і вважає поточну позицію за нульову по всіх осях. Внутрішньо ця команда оновлює значення в програмних змінних координат і також змінює прив'язку між фізичним положенням пристрою та його внутрішньою координатною системою.

Команда Homing (G28) призначена для повернення всіх осей маніпулятора у початкове положення (нульову точку), якщо це дозволено в налаштуваннях.

Команда Absolute Mode (G90) встановлює абсолютний режим координат. У цьому режимі всі координати у команді G0 трактуються як глобальні (відносно початку координат).

Команда Relative Mode (G91) встановлює відносний режим координат. У цьому режимі координати задаються як зміщення від поточного положення.

Приклад програмної реалізації обробки команд блока керування макетом промислового маніпулятора показано на рисунку 4.10.

```

if (cmd.id == 'G') {
  switch (cmd.num) {
    case 0:
    case 1:
      fan.enable(true);
      Point posoffset;
      posoffset = interpolator.getPosOffset();
      cmdMove(cmd, interpolator.getPosmm(), posoffset, command.isRelativeCoord);
      interpolator.setInterpolation(cmd.valueX, cmd.valueY, cmd.valueZ, cmd.valueE, cmd.valueF);
      Logger::logINFO("LINEAR MOVE: X" +
        String(cmd.valueX-posoffset.xmm) +
        " Y" + String(cmd.valueY-posoffset.ymm) +
        " Z" + String(cmd.valueZ-posoffset.zmm) +
        " E" + String(cmd.valueE-posoffset.emm));
      break;
    case 4: cmdDwell(cmd); break;
    case 28: homeSequence(); break;
    case 29: waitSequence(); break;
    case 90: command.cmdToAbsolute(); break; // ABSOLUTE COORDINATE MODE
    case 91: command.cmdToRelative(); break; // RELATIVE COORDINATE MODE
    case 92:
      interpolator.resetPosOffset();
      cmdMove(cmd, interpolator.getPosmm(), interpolator.getPosOffset(), false);
      interpolator.setPosOffset(cmd.valueX, cmd.valueY, cmd.valueZ, cmd.valueE);
      break;
    default: printErr();
  }
}

```

Рисунок 4.10 – Приклад програмної реалізації обробки команд блока керування макетом промислового маніпулятора

4.3 Моделювання системи автоматичного управління для рішення задачі слідкування маніпулятором за заданою траєкторією

Для вирішення завдання стеження за траєкторією до структури замкнутої системи додають прямий зв'язок як показано на рисунку 4.11.

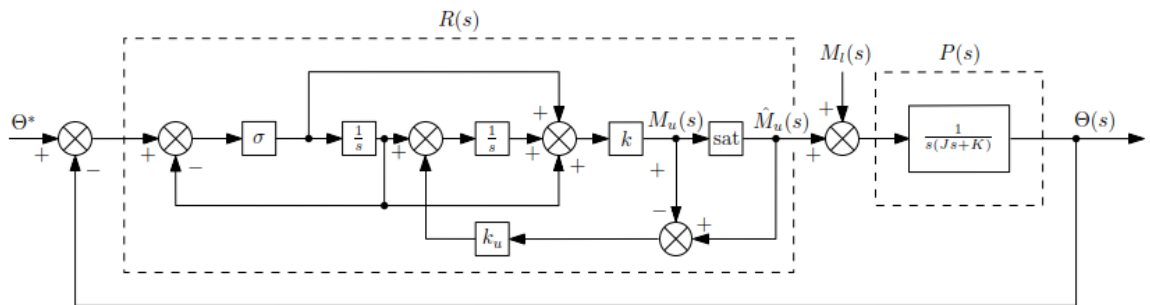


Рисунок 4.11 – Структурна схеми системи для вирішення задачі стеження за траєкторією

При рішенні даної задачі передатна функція $F(s)$ мусить бути стійкою, тобто, ступінь чисельника має перевищувати ступінь знаменника. Виразимо вихідний сигнал у вигляді:

$$\Theta(s) = \frac{R(s)P(s) + F(s)P(s)}{1 + R(s)P(s)} \Theta^*(s) + \frac{P(s)}{1 + R(s)P(s)} M_l(s). \quad (4.4)$$

Оберемо передатну функцію прямого зв'язку у формі:

$$F(s) = \frac{1}{P(s)}, \quad (4.5)$$

тоді вираз (4.4) можна записати у вигляді:

$$\theta(s) = \theta^*(s) + \frac{P(s)}{1 + R(s)P(s)} M_l(s). \quad (4.6)$$

Тоді структурна схема моделювання замкненої системи слідування за заданою траєкторією буде представлена у вигляді, що показано на рисунку 4.12.

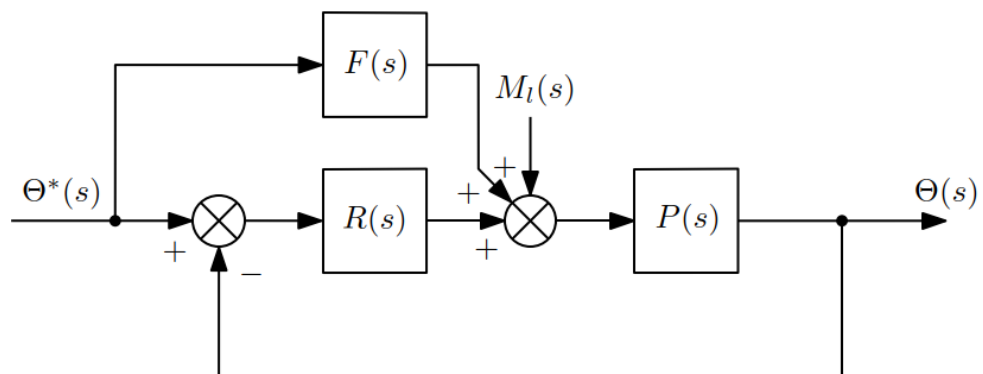


Рисунок 4.12 – Структурна схема моделювання замкненої системи слідування за заданою траєкторією

Таким чином, ми показали, що прямий зв'язок дозволяє системі стежити за заданою траєкторією за умови загальної стійкості замкнутої системи. Встановлена помилка при цьому буде зумовлена лише впливом зовнішнього обурення $M_l(s)$.

4.4 Результати експериментальних досліджень

На рисунку 4.13 показано зовнішній вигляд зібраного макету модуля керування промисловим маніпулятором.



Рисунок 4.13 – Зовнішній вигляд зібраного макету модуля керування промисловим маніпулятором.

На даному рисунку видно, що на даній стадії розробки до макету входять:

- модуль Arduino;
- плата розширення;
- драйвери крокових двигунів;
- крокові двигуни.

Програма керування маніпулятором розроблена у візуальному додатку RoboProg, що використовується на кафедрі КІТАР в якості цифрового двійника промислового маніпулятора (рис. 4.14).

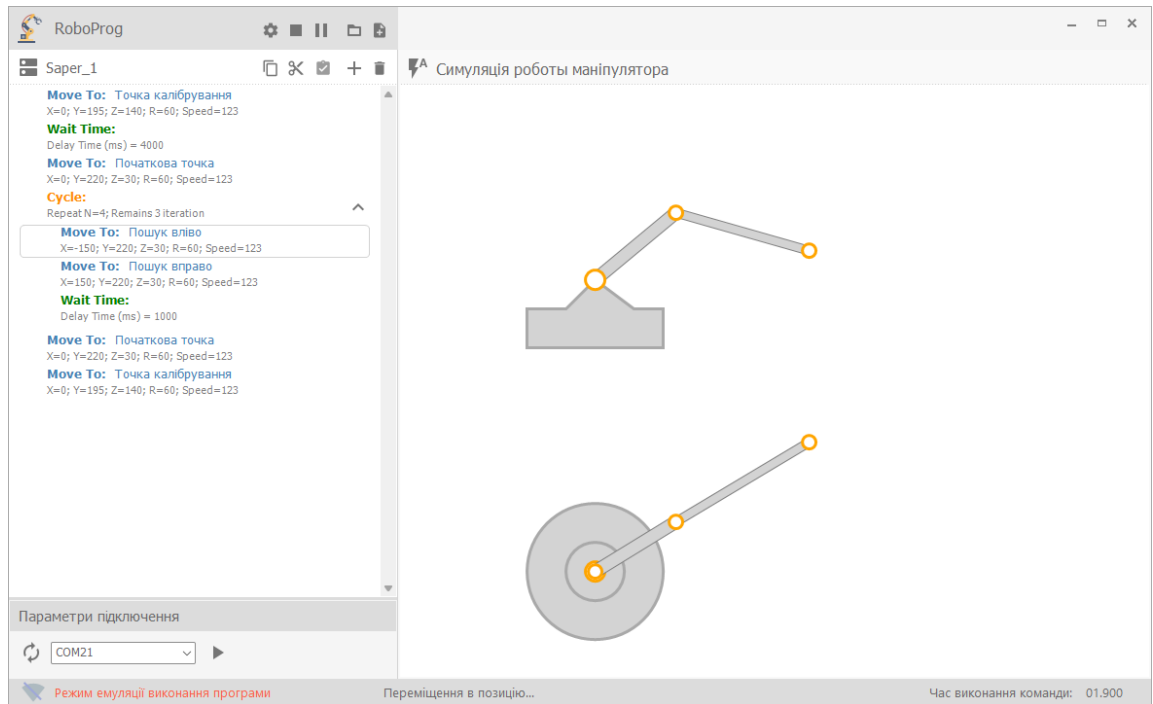


Рисунок 4.14 – Моделювання розробленої програми автоматичного управління рухом маніпулятора

Програма управління маніпулятором складається з декількох команд:

- калібрування;
- очікування завершення калібрування;
- переміщення в початкову точку;
- головного циклу в якому маніпулятор а автоматичному режимі виконує задачі маніпуляції об'єктом виробництва;
- завершальної фази управління, коли маніпулятор повертається в задану координату.

Складена програма показана на рисунку 4.15.

```
Move To: Точка калібрування  
X=0; Y=195; Z=140; R=60; Speed=123  
Wait Time:  
Delay Time (ms) = 4000  
Move To: Початкова точка  
X=0; Y=220; Z=30; R=60; Speed=123  
Cycle:  
Repeat N=4  
Move To: Пошук вліво  
X=-150; Y=220; Z=30; R=60; Speed=123  
Move To: Пошук вправо  
X=150; Y=220; Z=30; R=60; Speed=123  
Wait Time:  
Delay Time (ms) = 1000  
Move To: Початкова точка  
X=0; Y=220; Z=30; R=60; Speed=123  
Move To: Точка калібрування  
X=0; Y=195; Z=140; R=60; Speed=123
```

Рисунок 4.15 – Програма управління маніпулятором в автоматичному режимі

Таким чином, розроблений модуль керування макетом промислового маніпулятора виконує поставлену задачу та може працювати в автоматичному режимі, виконуючи задану послідовність команд в циклічному режимі.

5 ОХОРОНА ПРАЦІ

За ступенем небезпеки ураження електричним струмом згідно з ПУЕ – 2021 приміщення належить до класу приміщень без підвищеної небезпеки ураження електричним струмом. Умови, які створюють підвищену і особливу небезпеку (підвищена вологість, струмопровідний пил, струмопровідні підлоги, можливість одночасного дотику до заземлених металоконструкцій будівлі і металевих поверхонь електроприладів), відсутні.

З метою зниження небезпеки ураження людини електричним струмом проектом передбачається використання таких технічних засобів захисту:

- необхідно проводити контроль ізоляції відповідно до вимог ПУЕ – 2021. Контроль проводити між нульовим і фазним провідниками і між фазами. Опір ізоляції не менше 500 кОм на фазу. Контроль проводити не рідше 1 разу на рік при відключеному електроживленні;

- в приміщенні використовується система живлючих провідників, трифазна, чотирипровідна з глухо заземленою нейтраллю напругою до 1000 В, тому, згідно з НПАОП 40.1-1.32-01, використовується система заземлення TN-C-S типу.

Всі корпуси ПЕОМ з'єднані з глухо заземленою нейтраллю джерела живлення за допомогою нульового захисного провідника. Автомат захисту вибирається за струмом короткого замикання, час відключення 0,2 с.

Роботи в лабораторії відносяться до робіт категорії 1а – легка фізична робота, яка виконується сидячи.

Оптимальні норми мікроклімату для холодного та теплого періоду року згідно і ДСН 3.3.6.042-99:

- температура від 22 °С до 25 °С;
- відносна вологість від 40 % до 60 %;
- швидкість руху повітря не більше 0,1 м/с.

Забезпечуються припливно-витяжна загальнообмінна вентиляція.

В приміщенні використовується сумісне освітлення: природне та штучне. Згідно з ДБН В.2.5-28-2006 категорія зорових робіт, що проводяться у приміщенні – III В. Штучне освітлення виконано як загальне, за допомогою світильників з люмінесцентними лампами.

ВИСНОВКИ

В результаті виконання атестаційної роботи розроблено блок керування макету промислового маніпулятора для виконання виробничих завдань в автоматичному режимі.

Виконавши аналіз літератури визначені основні типи промислових маніпуляторів, проведена їх класифікація за різними критеріями та надані основні характеристики. Виконано аналіз принципів реалізації промислових маніпуляторів. Наведена схема, що пояснює принцип класифікації промислових роботів-маніпуляторів.

Виконано аналіз основних технічних показників промислових роботів та аналіз принципів побудови промислових маніпуляторів. Наведена типова структурна схема промислового робота. Проведено визначення систем координат при виконанні завдання промисловим роботом.

Розроблено архітектуру учбового лабораторного макету, виконано опис принципу роботи модуля автоматичного керування промисловим маніпулятором. Побудовано структурну схему автоматизованого модуля.

Описана конструкція макету промислового маніпулятора, наведено зовнішній вигляд макету промислового маніпулятора та описано призначення основних його компонентів. Докладно розглянуто механізм забезпечення руху ланок макету промислового маніпулятора, та наведена кінематична схема маніпулятора.

Розроблено алгоритм роботи автоматизованого модуля управління макетом промислового маніпулятора. Виконано вибір компонентів для реалізації автоматизованої системи. Для вирішення задачі створення автоматизованого модуля управління макетом промислового маніпулятора обрано модуль Arduino Nano з відповідними модулями розширення.

Виконано порівняльний аналіз драйверів крокового двигуна в результаті якого прийняте рішення, що в нашому проекті будемо застосовувати драйвер крокового двигуна BL-TB6600-V1.2.

Розроблена програма для мікроконтролера та описані основні її функції. Описано протокол обміну повідомленнями між блоком керування та пристроєм управління. Виконано моделювання системи автоматичного управління для рішення задачі слідкування маніпулятором за заданою траєкторією.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Методичні вказівки з підготовки кваліфікаційної роботи для здобувачів першого (бакалаврського) рівня вищої освіти спеціальності 151 Автоматизація та комп'ютерно-інтегровані технології освітньої програми «Системна інженерія» / Упоряд.: І.Ш. Невлюдов, О.М. Цимбал, О.В. Токарева, А.І. Бронніков. – Харків: ХНУРЕ, 2023. – 65 с.
2. ДСТУ 3008-2015. Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлення. – К.: Вид-во стандартів, 2016. – 26 с.
3. Liu, Zhaobing & Peng, Kerui & Han, Lvpeng & Guan, Shengchuang. (2023). Modeling and Control of Robotic Manipulators Based on Artificial Neural Networks: A Review. Iranian Journal of Science and Technology: Transactions of Mechanical Engineering. 1-41. 10.1007/s40997-023-00596-3.
4. Промислові роботи – Кафедра автоматизації технологічних процесів та виробництв [Електронний ресурс] // Кафедра АВ – Кафедра автоматизації технологічних процесів та виробництв. – Режим доступу: <https://kaf-av.tntu.edu.ua/index.php/mn-abiturient/mn-articles/676-art-industrial-robots> (дата звернення: 17.04.2025). – Назва з екрана.
5. Колаборативні роботи [Електронний ресурс]. – URL: <https://www.qweedo-robotics.com/robots/> (дата звернення: 17.04.2025). – Назва з екрана.
6. Цвіркун Л.І. Робототехніка та мехатроніка: навч. посіб. / Л.І. Цвіркун, Г. Грулер ; під заг. ред. Л.І. Цвіркуна ; М-во освіти і науки України, Нац. гірн. ун-т. – 3-тє вид., переробл. і доповн. – Дніпро: НГУ, 2017. – 224 с.
7. Омаров М. Особливості кінематики та управління багатоланкових роботів маніпуляторів [Електронний ресурс]. – URL: <https://doi.org/10.26906/SUNZ.2023.1.127>
8. Невлюдов І. Ш. Комп'ютерно-інтегровані технології виробництва технічних засобів автоматизації. Частина 1: підручник для студентів закладів

вищої освіти ; Харків. нац. ун-т радіоелектроніки. – Харків : ФОП Панов А.М., 2021. – 604 с. ISBN 978-617-7947-67-6

9. Невлюдов І.Ш. Виробничі процеси та обладнання об'єктів автоматизації: Підручник / Кривий Ріг: КК НАУ, 2017. – 444 с.

10. Невлюдов І.Ш. Технічні засоби автоматизації: Підручник / І.Ш. Невлюдов, А.О. Андрусевич, О.І. Филипченко, Н.П. Демська, С.П. Новоселов. – Кривий Ріг: Криворізький коледж НАУ, 2019. – 366 с.

11. Невлюдов І. Ш., Андрусевич А. О., Євсєєв В. В., Новоселов С. П., Демська Н. П. Проектування мобільних маніпуляційних роботів: Монографія. – Х. :2022. – 427 с.

12. Новоселов С.П., Роменський В.І., Циганок С.П. Оптимізація методу керування рухом мобільного робота з використанням двигунів постійного струму // Технологія приборостроєння: Научн.-техн. журнал. – Харків: НИТИП. – 2019. – №2. – С.14-17.

13. Nevlyudov, I., Novoselov, S., Sychova, O. (2022), "Development and study of the operation of the module for determining the orientation of the manipulator joint", Innovative Technologies and Scientific Solutions for Industries, No. 2 (20), P. 86-96. DOI: <https://doi.org/10.30837/ITSSL2022.20.086>

14. Плата Arduino Nano v3.0, контролер ЧПК [Електронний ресурс] // Комплектуючі для ЧПУ. – Режим доступу: <https://cncprom.ua/ua/p1524259529-plata-arduino-nano.html> (дата звернення: 13.05.2025). – Назва з екрана.

15. Цифровий драйвер крокового двигуна DM556D 5,6 А [Електронний ресурс] // Комплектуючі для ЧПУ. – Режим доступу: <https://cncprom.ua/ua/p445157486-tsifrovoj-drajver-shagovogo.html> (дата звернення: 13.05.2025). – Назва з екрана.

16. Драйвер крокового двигуна BL-TB6600-V1.2: [Електронний ресурс] // Комплектуючі для ЧПУ. – Режим доступу: <https://cncprom.ua/ua/p700538092-drajver-shagovogo-dvigatelya.html> (дата звернення: 13.05.2025). – Назва з екрана.