

Міністерство освіти і науки України
Державний заклад
«Луганський національний університет імені Тараса Шевченка»

Навчально-науковий інститут математики та інформаційних технологій

Кафедра інформаційних технологій та систем

Рябець Андрій Володимирович

**СИСТЕМА КЕРУВАННЯ РОБОТОМ-МАНІПУЛЯТОРОМ НА БАЗІ
ПЛАТФОРМИ ARDUINO**

кваліфікаційна робота

здобувача вищої освіти другого (магістерського) рівня

освітньої програми «Комп'ютерні мережі»

за спеціальністю 123 Комп'ютерна інженерія

Особистий підпис



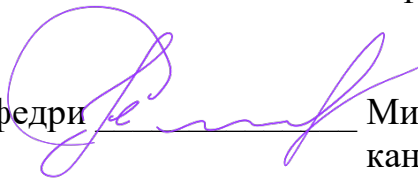
Андрій РЯБЕЦЬ

Науковий керівник



Володимир ДОНЧЕНКО,
старший викладач кафедри
інформаційних технологій та систем

Завідувач кафедри



Микола СЕМЕНОВ,
кандидат педагогічних наук, доцент
кафедри інформаційних технологій
та систем

Лубни – 2026

АНОТАЦІЯ

Рябець А. В.

Тема: Система керування роботом-маніпулятором на базі платформи Arduino.

Спеціальність: 123 «Комп'ютерна інженерія».

Установа: ЛНУ імені Тараса Шевченка, 2026 р.

Магістерська робота містить: 107 с., 68 рис., 11 табл., 2 додат., 60 джерел.

Об'єктом дослідження є процес автоматизованого керування багатоланковими маніпуляційними механізмами.

Предметом дослідження є апаратні та програмні засоби системи керування роботом-маніпулятором, побудованої на базі мікроконтролера Arduino та бездротових технологій передачі даних.

Мета дослідження полягає у проєктуванні, математичному моделюванні та практичній реалізації системи автоматизованого керування чотириланковим роботом-маніпулятором на базі платформи Arduino з можливістю дистанційного управління через Android-застосунок та голосових команд.

Результати роботи. У ході виконання роботи проведено аналітичний огляд сучасних промислових роботів-маніпуляторів (KUKA, Fanuc, Universal Robots), проаналізовано їхні конструктивні особливості, типи захватних пристроїв та систем керування. Обґрунтовано доцільність використання платформи Arduino та сервоприводів для створення навчального прототипу. Розроблено математичну модель механічної системи маніпулятора. Застосовано метод Денавіта–Хартенберга для опису кінематики та рівняння Лагранжа для моделювання динаміки руху ланок, що дозволило визначити необхідні параметри для точного керування.

Спроектовано 3D-модель конструкції маніпулятора в середовищі Autodesk Inventor, адаптовану для виготовлення методом 3D-друку, та обрано оптимальну кінематичну схему з чотирма ступенями свободи. Розроблено апаратну частину системи керування на базі мікроконтролера Keyestudio UNO R3 та плати розширення для сервоприводів. Створено та змодельовано електричну принципову схему в середовищі Proteus, яка включає модуль Bluetooth HC-05 для бездротового зв'язку. Створено програмне забезпечення двох рівнів: скетч для мікроконтролера (в середовищі Arduino IDE) для обробки сигналів та керування сервоприводами; мобільний додаток для ОС Android (в середовищі MIT App Inventor), який реалізує інтерфейс користувача з можливістю ручного керування повзунками, голосового керування через Google Voice Recognition API, а також функцією запису та відтворення траєкторій. Проведено тестування розробленої системи на віртуальній моделі та фізичному макеті. Виявлено та усунуто проблеми з живленням та плавністю ходу приводів, підтверджено працездатність голосового керування та стабільність Bluetooth-з'єднання.

Ключові слова. МАНІПУЛЯЦІЙНА СИСТЕМА, МАТЕМАТИЧНА МОДЕЛЬ, РІВНЯННЯ ЛАГРАНЖА, BLUETOOTH, ПРОГРАМА ДЛЯ УПРАВЛІННЯ НА БАЗІ ANDROID, ARDUINO, ARDUINO IDE.

ABSTRACT

Riabets Andrii

Theme: Robot manipulator control system based on the Arduino platform.

Speciality: 123 "Computer Engineering"

Institution: Luhansk Taras Shevchenko National University (LTSNU), 2026.

Master's work of: 107 pages, 68 Fig., 11 Table, 2 adj., 60 source.

A research object is the process of automated control of multi-link manipulation mechanisms.

The article of research is hardware and software of the robot-manipulator control system, built on the basis of the Arduino microcontroller and wireless data transmission technologies.

An aim of work is consists of designing, mathematical modeling and practical implementation of an automated control system for a four-link robot-manipulator based on the Arduino platform with the possibility of remote control via an Android application and voice commands.

Job performances. In the course of the work, an analytical review of modern industrial manipulator robots (KUKA, Fanuc, Universal Robots) was conducted, their design features, types of gripping devices and control systems were analyzed. The feasibility of using the Arduino platform and servo drives to create a training prototype was substantiated. A mathematical model of the mechanical system of the manipulator was developed. The Denavit–Hartenberg method was used to describe kinematics and the Lagrange equation was used to model the dynamics of the movement of links, which made it possible to determine the necessary parameters for precise control. A 3D model of the manipulator design was designed in the Autodesk Inventor environment, adapted for manufacturing by 3D printing, and the optimal kinematic scheme with four degrees of freedom was selected. The hardware part of the control system was developed based on the Keyestudio UNO R3 microcontroller and expansion boards for servo drives. An electrical schematic diagram was created and simulated in the Proteus environment, which includes a Bluetooth HC-05 module for wireless communication. Two-level software was created: a sketch for a microcontroller (in the Arduino IDE environment) for signal processing and servo control; a mobile application for Android OS (in the MIT App Inventor environment), which implements a user interface with the ability to manually control sliders, voice control via the Google Voice Recognition API, as well as the function of recording and playing back trajectories. The developed system was tested on a virtual model and a physical layout. Problems with power supply and smoothness of the actuators were identified and eliminated, the operability of voice control and the stability of the Bluetooth connection were confirmed.

Keywords. MANIPULATION SYSTEM, MATHEMATICAL MODEL, LAGRANGE EQUATION, BLUETOOTH, ANDROID-BASED CONTROL PROGRAM, ARDUINO, ARDUINO IDE.

ЗМІСТ

ВСТУП.....	6
РОЗДІЛ 1. ВИКОРИСТАННЯ РОБОТІВ-МАНІПУЛЯТОРІВ ТА ПЛАТФОРМ ARDUINO У СИСТЕМАХ АВТОМАТИЗАЦІЇ	9
1.1. Аналіз функціональних можливостей існуючих роботів-маніпуляторів ...	9
1.2. Аналіз технічних характеристик роботів-аналогів.....	20
1.3. Аналіз програмних засобів керування	24
1.4. Визначення моделі захвату робота-маніпулятора	27
1.5. Використання Arduino у робототехніці	31
Висновки до розділу	34
РОЗДІЛ 2. МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ТА КОНСТРУКТОРСЬКЕ ПРОЄКТУВАННЯ МЕХАНІЧНОЇ СИСТЕМИ РОБОТА-МАНІПУЛЯТОРА.....	35
2.1. Типові схеми маніпуляторів	35
2.2. Математична модель робота маніпулятора для системи управління.....	39
2.3. Проєктування 3-D моделі робота маніпулятора	46
2.4. Керування приводом.....	50
Висновки до розділу	54
РОЗДІЛ 3. ПРОЄКТУВАННЯ ТА ПРАКТИЧНА РЕАЛІЗАЦІЯ СИСТЕМИ АВТОМАТИЗОВАНОГО КЕРУВАННЯ РОБОТОМ-МАНІПУЛЯТОРОМ НА БАЗІ ПЛАТФОРМИ ARDUINO	55
3.1 Структура та принципи роботи системи автоматизованого керування.....	55
3.2. Розробка системи керування роботом маніпулятором на основі ARDUINO	62
3.3. Розробка електричної схеми системи керування маніпулятором.....	76
3.4. Розробка додатку системи управління для смартфона на базі Android	80
3.5. Програмне забезпечення для автоматизованого керування роботом-маніпулятором	88
3.6. Тестування автоматизованої маніпуляційної системи.....	93
Висновки до розділу	97

ВИСНОВКИ	99
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	101
ДОДАТОК А	108
ДОДАТОК Б	114

ВСТУП

Сучасний етап технологічного розвитку характеризується стрімкою інтеграцією робототехнічних систем у всі сфери людської діяльності — від важкого машинобудування до медицини та побуту. Промислові маніпулятори провідних виробників, таких як KUKA, Fanuc чи Universal Robots, забезпечують еталонну точність та продуктивність. Водночас їх висока вартість, складність технічного обслуговування та закритість програмного забезпечення створюють суттєві бар'єри для їх використання у малому бізнесі, дослідницьких лабораторіях та освітньому процесі.

У зв'язку з цим виникає нагальна потреба у розробці гнучких, відкритих та економічно доступних систем керування. Ефективним рішенням є поєднання апаратної доступності платформи Arduino з обчислювальною потужністю сучасних мобільних пристроїв. Використання смартфона як основного інструменту керування не лише знижує витрати на спеціалізоване обладнання, але й забезпечує мобільність системи, спрощує інтерфейс «людина-машина» (зокрема через голосове керування) та відкриває можливості інтеграції з хмарними сервісами і базами даних.

Попри очевидний прогрес у цій галузі, низка питань залишається невирішеною. Зокрема, потребують вдосконалення алгоритми керування в реальному часі та методи забезпечення стійкості систем до зовнішніх впливів. Ключовим викликом є створення універсальних математичних моделей, які гарантували б точність, адаптивність та самодіагностику бюджетних маніпуляторів.

Таким чином, розробка автоматизованої системи управління маніпуляційними роботами на базі мікроконтролерів та смартфонів із впровадженням інноваційних математичних моделей є актуальним науково-прикладним завданням. Це дослідження спрямоване на подолання існуючих обмежень, що дозволить розширити використання робототехніки у виробничих, побутових та спеціальних умовах, забезпечуючи їх високу ефективність та економічну доцільність.

Об'єктом дослідження є процес автоматизованого керування багатоланковими маніпуляційними механізмами.

Предметом дослідження є апаратні та програмні засоби системи керування роботом-маніпулятором, побудованої на базі мікроконтролера Arduino та бездротових технологій передачі даних.

Мета дослідження полягає у проєктуванні, математичному моделюванні та практичній реалізації системи автоматизованого керування чотириланковим роботом-маніпулятором на базі платформи Arduino з можливістю дистанційного управління через Android-застосунок та голосових команд.

Для досягнення поставленої мети необхідно вирішити такі завдання:

1. Провести аналіз існуючих аналогів промислових маніпуляторів, їх технічних характеристик та методів керування, а також обґрунтувати вибір апаратної платформи.
2. Розробити математичну модель кінематики та динаміки маніпулятора (використовуючи метод Денавіта–Хартенберга та рівняння Лагранжа) для забезпечення точності позиціонування.
3. Здійснити конструкторське проєктування та 3D-моделювання механічної системи маніпулятора.
4. Розробити електричну принципову схему системи керування та обґрунтувати вибір компонентів (мікроконтролер, сервоприводи, Bluetooth-модуль).
5. Створити програмне забезпечення для мікроконтролера та мобільний додаток для ОС Android із функціями ручного та голосового керування.
6. Виконати тестування розробленої системи (віртуальне моделювання та фізичні випробування) та перевірити її працездатність.

У роботі використано комплексний підхід, що включає:

- аналітичні методи — для огляду сучасного стану робототехніки та вибору компонентної бази;
- методи математичного моделювання — для розрахунку кінематики та динаміки маніпулятора (рівняння Лагранжа II роду);

- методи 3D-моделювання (Autodesk Inventor) — для конструювання механічних вузлів;
- схемотехнічне моделювання (Proteus) — для верифікації електричних з'єднань;
- об'єктно-орієнтоване та блокове програмування (Arduino IDE, MIT App Inventor) — для розробки керуючих алгоритмів;
- експериментальний метод — для тестування прототипу та перевірки точності позиціонування.

Наукова новизна одержаних результатів полягає у подальшому розвитку підходів до створення бюджетних систем керування робототехнічними комплексами шляхом інтеграції платформи Arduino, сервоприводів та технологій розпізнавання мови Google Voice API, що дозволило реалізувати інтуїтивний інтерфейс керування маніпулятором.

Практичне значення одержаних результатів полягає у створенні діючого прототипу роботизованої системи, яка може бути використана як навчальний стенд для вивчення основ робототехніки, мехатроніки та програмування мікроконтролерів.

Запропоновані схемотехнічні та програмні рішення дозволяють легко масштабувати систему та адаптувати її для виконання різних маніпуляційних завдань малого навантаження.

РОЗДІЛ 1. ВИКОРИСТАННЯ РОБОТІВ-МАНІПУЛЯТОРІВ ТА ПЛАТФОРМ ARDUINO У СИСТЕМАХ АВТОМАТИЗАЦІЇ

1.1. Аналіз функціональних можливостей існуючих роботів-маніпуляторів

Компанія Universal Robots спеціалізується на розробленні та виробництві роботів-маніпуляторів, призначених для автоматизації повторюваних виробничих операцій. Асортимент продукції представлений серією з трьох легких промислових маніпуляторів — Universal Robots UR10, UR5 та UR3, які мають розімкнену кінематичну структуру (рис. 1.1).

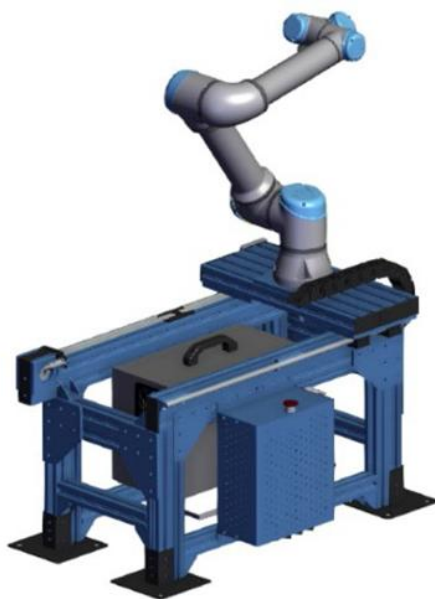


Рис. 1.1. Легкі промислові роботи-маніпулятори компанії Universal Robots

Моделі роботів Universal Robots характеризуються наявністю шести ступенів свободи, з яких три забезпечують поступальні переміщення, а ще три — орієнтацію робочого органу в просторі (рис. 1.2). Маніпулятори цієї серії виконують виключно кутові переміщення ланок. Класифікація роботів-маніпуляторів здійснюється залежно від максимально допустимого корисного навантаження. Основними параметрами, що відрізняють окремі моделі, є радіус робочої зони, маса конструкції та діаметр опорної основи.



Рис. 1.2. Ступені свободи роботів-маніпуляторів Universal Robots

Маніпулятори серії UR оснащені високоточними датчиками абсолютного положення, що значно спрощує їх інтеграцію з додатковим обладнанням і зовнішніми системами керування. Завдяки компактним габаритам ці роботи не потребують значного робочого простору та можуть бути легко встановлені як у окремих робочих зонах, так і безпосередньо на виробничих лініях.

Важливою перевагою роботів-маніпуляторів Universal Robots є зручність і доступність програмування. Спеціально розроблена та запатентована технологія дозволяє операторам без спеціальної технічної підготовки швидко виконувати налаштування та керування пристроями за допомогою інтуїтивно зрозумілої системи 3D-візуалізації. Процес програмування здійснюється шляхом послідовного переміщення робочого органу маніпулятора в потрібні положення або за допомогою керуючих елементів у спеціалізованому програмному середовищі [2].

Суттєвою перевагою маніпуляторів серії UR у порівнянні з іншими роботизованими системами є висока швидкість введення в експлуатацію. Для виконання первинного запуску обладнання оператору, як правило, достатньо менше однієї години, що включає розпакування, монтаж і програмування першої базової технологічної операції, яку виконуватиме робот-маніпулятор.

У системі керування роботами-маніпуляторами здійснюється постійний облік зовнішніх впливів, що виникають під час їхньої експлуатації. Маніпулятори Universal Robots здатні ефективно замінювати операторів, які

виконують монотонні завдання в небезпечних або забруднених виробничих умовах. Завдяки вбудованим засобам безпеки маніпуляційні системи серії UR можуть працювати без використання захисних огорожень та розміщуватися безпосередньо поблизу робочих місць персоналу. Системи безпеки цих роботів пройшли сертифікацію та отримали схвалення Союзу працівників технічного нагляду Німеччини. Додатковою перевагою маніпуляторів даної серії є універсальність монтажу, оскільки їх конструкція не потребує спеціальних умов встановлення.

Застосування промислових роботів-маніпуляторів серії UR відкриває широкі можливості для автоматизації практично всіх циклічних і рутинних виробничих процесів. Роботи компанії Universal Robots успішно використовуються в різних галузях промисловості. Зокрема, впровадження маніпуляторів серії UR на операціях перекладання та пакування дозволило підвищити точність виконання робіт і зменшити втрати продукції. Значна частина таких операцій нині виконується в автоматичному режимі без постійного контролю з боку оператора або працівників підрозділу [2].

Одним із прикладів важкої промислової робототехніки є робот-маніпулятор Fanuc M-2000iA/1200, розроблений компанією Fanuc (рис. 1.3). Даний робот належить до класу надважких маніпуляторів і створений для роботи з вантажами великої маси. Використання такого підйомного обладнання забезпечує високу надійність і ефективність роботизованих систем, оскільки застосування сучасних роботів дозволяє замінити до чотирьох звичайних вантажопідйомних механізмів або операторів.

Традиційні підйомні засоби, зокрема транспортери та крани, потребують безпосередньої участі людини, а робота з ними зазвичай супроводжується підвищеним ризиком. Перевагою робота Fanuc M-2000iA/1200 є повна автоматизація процесів при збереженні функціональних можливостей, що не поступаються стандартним тяговим пристроям. Це дозволяє суттєво знизити ймовірність виробничого травматизму. Висока міцність і рівень безпеки є ключовими характеристиками даного вантажопідйомного робота. Виробник

забезпечив відповідність цієї моделі найжорсткішим вимогам, що висувуються до вантажно-розвантажувальних операцій. Робот-маніпулятор функціонує в діапазоні температур від 0 °С до 45 °С, має п'ять осей керування, здатний підіймати вантажі масою до 1200 кг і переміщувати їх на відстань до 3,7 м. Автономна робота без участі людини практично усуває вплив людського фактору та зводить ризик травматизму до мінімуму.

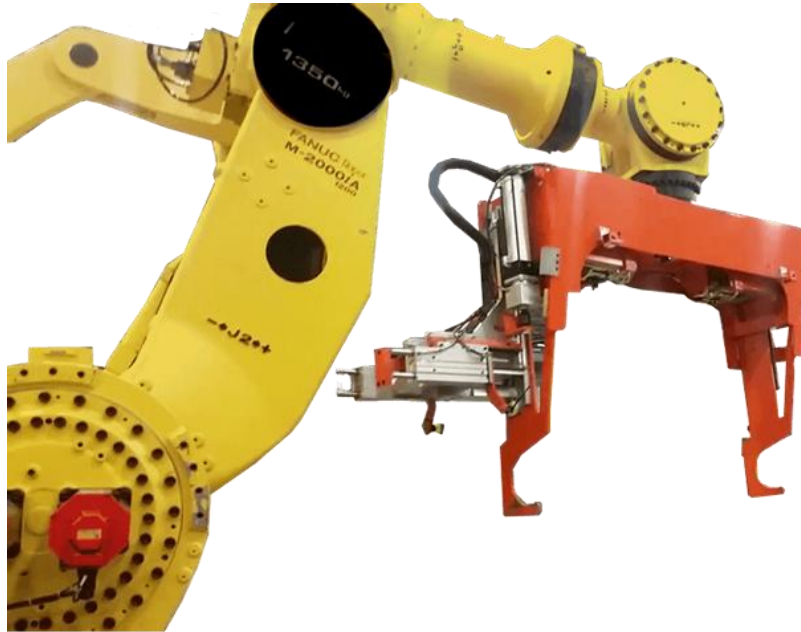


Рис. 1.3. Зовнішній вигляд робота-маніпулятора Fanuc M-2000iA/1200

Робот-маніпулятор Fanuc M-2000iA є найпотужнішою моделлю в лінійці компанії Fanuc, оскільки здатний ефективно працювати з надважкими вантажами масою до 1200 кг. Один такий робот може виконувати операції, для яких зазвичай необхідно залучати від двох до чотирьох стандартних роботизованих систем. При цьому керуючі системи даних маніпуляторів були суттєво спрощені, що підвищило їх надійність і зручність програмування. Завдяки робочому вильоту 1250 мм дана модель є оптимальною для переміщення та палетування масивних і габаритних конструкцій (рис. 1.4).

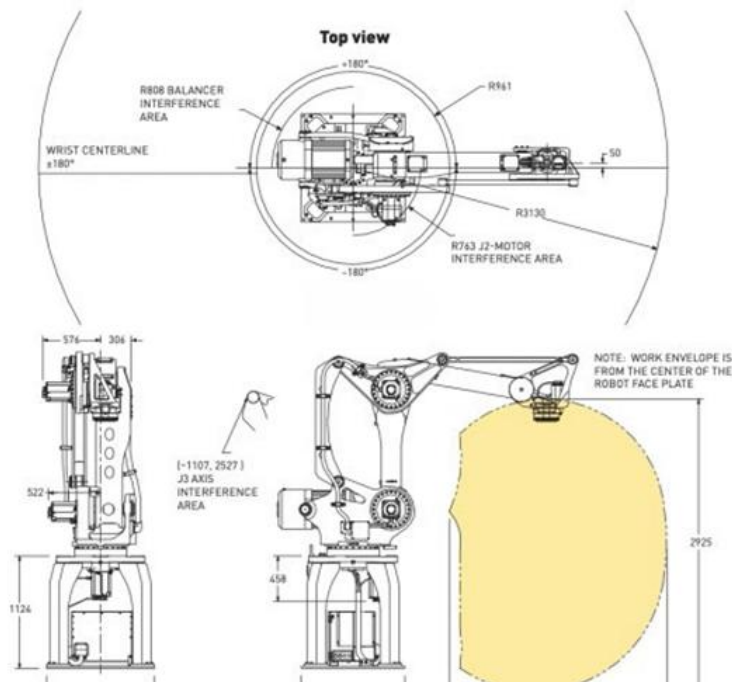


Рис. 1.4. Схема-діаграма робочого діапазону робота-маніпулятора Fanuc M-2000iA

Визначальною особливістю робота Fanuc M-2000iA є надпотужне зап'ястя, яке вважається одним із найміцніших у світі та дозволяє переміщувати важкі залізничні колісні пари масою до 1300 кг. Завдяки цьому робот може повністю замінити традиційні крани, підйомні механізми та транспортні візки, усуваючи небезпечну ручну працю та суттєво підвищуючи продуктивність виробничих процесів.

Модель Fanuc M-2000iA/1200 є ефективним рішенням для застосування у важкій промисловості, де ключовими завданнями є мінімізація впливу людського фактора та скорочення використання менш потужних промислових роботів.

На рис. 1.5 представлено робот-маніпулятор KR Quantec PA, розроблений компанією KUKA Robotics. Особливістю цієї моделі є здатність стабільно працювати за умов низьких температур — до -30°C . Завдяки цій характеристиці робот є оптимальним для використання в харчовій промисловості. Модифікація KUKA KR Quantec PA Arctic була спеціально спроектована для виконання операцій палетування у морозильних камерах і не потребує додаткових захисних кожухів для експлуатації при мінусових температурах [3].



Рис. 1.5. Робот-маніпулятор KUKA KR QUANTEC PA

Модель KR Quantec PA вирізняється застосуванням передових технологічних рішень, спрямованих на глибоку оптимізацію виробничих процесів. Висока швидкість роботи в межах максимальної робочої зони, мінімальна кількість виступаючих елементів конструкції та висока надійність забезпечують ефективну автоматизацію. Дані роботи спеціально розроблені для виконання складних навантажувально-розвантажувальних операцій.

Основними перевагами робота-маніпулятора KR Quantec PA є висока швидкодія, економічність, значна продуктивність і компактні габарити. Додатково слід відзначити коротку тривалість робочих тактів, раціональне використання простору та високу точність позиціонування при мінімальних габаритах зон перешкод [3]. За показниками швидкості виконання тактів KR Quantec PA належить до найшвидших роботів-маніпуляторів у своєму класі, зберігаючи стабільну пропускну здатність навіть за повного навантаження та мінімальної тривалості циклів.

Слід зазначити, що всі роботи серії Quantec PA оснащені маніпуляторною рукою з порожнистим валом, призначеним для внутрішнього розміщення кабелів і систем енергопостачання, що дозволяє раціонально використовувати простір. Подовжена, легка та компактна конструкція руки забезпечує зручне під'єднання захватних пристроїв і мінімізує зони можливих перешкод під час роботи. Завдяки збільшеній довжині робочої ланки та малій монтажній площі KR Quantec PA потребує мінімального простору для встановлення, що створює

умови для розроблення інноваційних роботизованих осередків, зокрема для виконання операцій з розвантаження.

Робот-маніпулятор KR Quantec PA був спеціально спроектований для складних вантажно-розвантажувальних завдань. Його потужність і динамічні характеристики дозволяють здійснювати штабелювання декількох палетів на значній висоті, забезпечуючи високу ефективність роботи [4].

Основні сфери застосування роботів з великою вантажопідйомністю охоплюють:

- усі види зварювальних операцій: точкове зварювання, роботизоване зварювання в середовищі захисних газів, дугове та лазерне зварювання;
- різноманітні маніпуляційні операції: навантаження і розвантаження, комплектування (у тому числі вибіркове), встановлення, вставляння, пакування та палетування;
- паяння;
- фарбування, нанесення клеїв і герметизуючих матеріалів;
- механічну обробку поверхонь металевих, полімерних та інших матеріалів;
- різні способи різання;
- проведення вимірювань, сервісних перевірок, діагностики й тестування обладнання та готових виробів;
- монтажні роботи, зокрема складання, з'єднання, розбирання та встановлення кріпильних елементів;
- виконання операцій на пресових і формувальних ділянках.

Ще одним прикладом промислового робота-маніпулятора є модель IRB 120, розроблена компанією ABB. Це промисловий робот четвертого покоління, який належить до найменших за габаритами серед усіх роботів, створених ABB. ABB IRB 120 призначений для виконання операцій зі складання та обробки матеріалів (рис. 1.6).



Рис. 1.6. Промисловий робот-маніпулятор ABB IRB 120

Робот-маніпулятор ABB IRB 120 знаходить застосування у різних галузях промисловості, таких як електронна, харчова, енергетична, фармацевтична, а також у медицині та науково-дослідних лабораторіях. Цей компактний шестикоординатний промисловий робот здатний переміщати вантажі вагою до 5 кг. Всі силові та інформаційні кабелі прокладені всередині руки маніпулятора, що виключає перешкоди, підвищує безпеку й забезпечує гнучкість інтеграції в різні виробничі середовища. Можливість монтажу під будь-яким кутом робить його ідеальним для установки в обмежених просторових умовах.

Легкий алюмінієвий корпус у поєднанні з потужними компактними моторами дозволяє IRB 120 працювати з високою швидкістю та точністю. Маніпулятор оснащений компактным контролером для забезпечення стабільного керування. Вага робота становить всього 25 кг, а вантажопідйомність — 3 кг (4 кг при відключеній п'ятій осі), робочий радіус досягає 580 мм. Це робить його ефективним, надійним та бюджетним рішенням з швидкою окупністю. Існує також сертифікована версія для роботи в чистих приміщеннях Clean Room ISO 5 (Class 100). Завдяки високій швидкості і повторюваності операцій до 10 мкм, IRB 120 є найшвидшим серед лінійки роботів ABB IRB [5]. Компактний дизайн і мала вага дозволяють збільшити швидкість осей 4, 5 і 6, підвищуючи продуктивність та скорочуючи тривалість циклу до 25% (рис. 1.7).

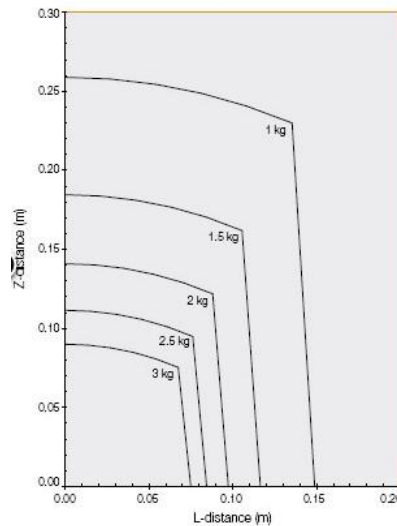
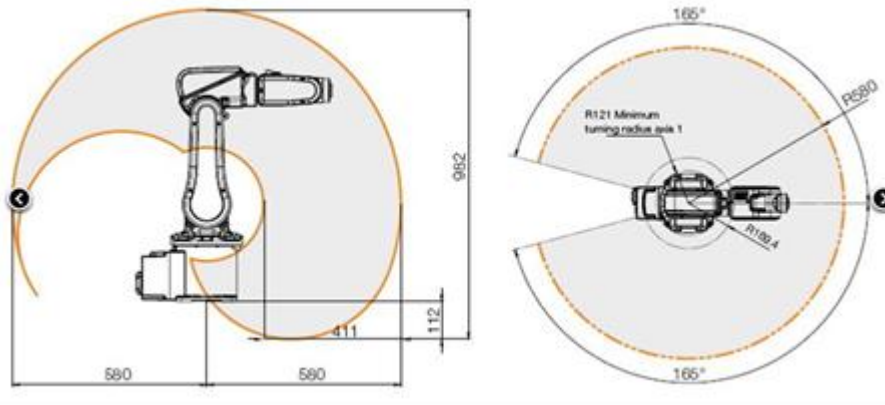


Рис. 1.7. Діаграма навантаження та робочий діапазон IRB 120

У IRB 120 горизонтальна смуга руху становить 580 мм, робочий хід бази по вертикалі — 112 мм, а компактний радіус повороту забезпечує ефективну роботу навіть у обмеженому просторі.

Розглянемо п'ять основних серій роботів-маніпуляторів KR QUANTEC від компанії KUKA. Серія Nano призначена для умов з обмеженими виробничими площами (рис. 1.8). Ці роботи успішно виконують завдання навіть у компактних робочих зонах, демонструючи можливості підвищення точності роботизованого точкового зварювання. Моделі серії оснащені механізмами повороту з великим робочим діапазоном, що дозволяє ефективно вирішувати різноманітні технологічні задачі.



Рис. 1.8. Модель робота-маніпулятора KUKA KR QUANTEC Nano

Серія Pro від KUKA KR QUANTEC вирізняється підвищеною міцністю, великим робочим радіусом, високою точністю руху руки та значною потужністю моделей, що дозволяє їм стабільно функціонувати навіть за максимальних навантажень (рис. 1.9). Компактна конструкція корпусу без виступаючих елементів дозволяє економити виробничий простір та організовувати робочі осередки у вузьких зонах при високій вантажопідйомності маніпулятора.



Рис. 1.9. Зовнішній вигляд маніпулятора KUKA KR QUANTEC Pro

Серія Extra здатна виконувати широкий спектр завдань без значних додаткових інвестицій (рис. 1.10). Завдяки витонченому дизайну та зниженню маси рухомих елементів, робот зберігає надзвичайну міцність і надійність конструкції, забезпечуючи великі робочі радіуси для свого класу.



Рис. 1.10. Зовнішній вигляд маніпулятора KUKA KR QUANTEC Extra

Модель Prime поєднує елегантну конструкцію та легкість з високою жорсткістю і відмінною вантажопідйомністю (рис. 1.11). Основними перевагами цієї серії є точність виконання траєкторій і здатність працювати з короткими циклами, при цьому робочий радіус руки залишається таким же широким, як у серії Extra. Зокрема, точкове зварювання демонструє високу енергоефективність та успішну роботу у обмеженому просторі.

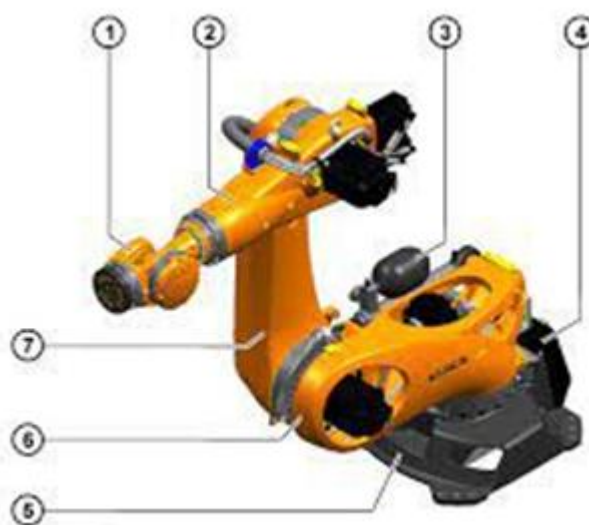


Рис. 1.11. Зовнішній вигляд маніпулятора KUKA KR QUANTEC Prime

Серія Ultra призначена для задач, що потребують максимальної питомої потужності та найбільшого робочого радіусу при великій вантажопідйомності (рис. 1.12). Вага та габаритні розміри цих роботів практично не відрізняються від інших моделей KUKA у цьому сегменті. Роботи серії Ultra демонструють неймовірну швидкість та стабільність, дозволяючи досягати нового рівня продуктивності [5].



Рис. 1.12. Зовнішній вигляд маніпулятора KUKA KR QUANTEC Ultra

На основі проведеного аналізу різних серій роботів-маніпуляторів були визначені їхні переваги та недоліки. Це дозволяє оцінити критичні обмеження існуючих моделей та спроектувати власний маніпулятор із врахуванням усунення цих недоліків. Найважливішим недоліком більшості розглянутих роботів є вплив людського фактора на процес керування. Проаналізована інформація підказує можливі шляхи його мінімізації або повного усунення під час керування та контролю [6].

1.2. Аналіз технічних характеристик роботів-аналогів

У цьому підрозділі розглядаються технічні характеристики раніше описаних роботів-маніпуляторів, оскільки саме технічні параметри є ключовими при проєктуванні, розробці та виготовленні подібних систем. Роботи-маніпулятори суттєво відрізняються між собою, що зумовлено насамперед масштабами їхнього застосування та цільовим призначенням.

Технічні характеристики маніпуляторів серії Universal Robots (UR) наведені у таблиці 1.1. З аналізу даних випливає, що основним недоліком цієї серії є порівняно невисока точність позиціонування у порівнянні з іншими аналогічними роботами. Водночас, UR має ряд суттєвих переваг: невеликий діаметр основи, компактну вагу та порівняно низьке споживання енергії. Вага і габарити моделей цієї серії незначно відрізняються від інших промислових роботів KUKA у своєму класі.

Таблиця 1.1

Технічні характеристики роботів-маніпуляторів Universal Robots

Найменування характеристик	Одиниці вимірювання	UR3	UR5	UR10
Точність позиціонування	мм	±0,1	±0,1	±0,1
Діапазон робочих температур	°C	0...+50	0...+50	0...+50
Споживана потужність	Вт	125	150	250
Вантажопідйомність	кг	3	5	10
Кількість ступенів свободи	—	6	6	6
Робочий діапазон (суглоби)	град.	±360	±360	±360
Рівень шуму	дБ	70	72	72
Діаметр основи	мм	128	149	190
Матеріал	—	алюміній, поліпропілен		
Довжина кабелю	м	6	6	6
Маса (вага) моделі	кг	11	18,4	28,9

Технічні параметри робота-маніпулятора Fanuc M-2000iA / 1200 наведені у таблиці 1.2. Незважаючи на значну кількість переваг цієї моделі, у Fanuc M-2000iA / 1200 виявляються певні недоліки. Серед них – складність конструкції,

велика маса та об'ємність. Цей робот-маніпулятор призначений для широкого застосування, зокрема у автомобільній промисловості, де потрібна висока вантажопідйомність та стабільність роботи.

Таблиця 1.2

Технічні характеристики робота-маніпулятора Fanuc M-2000iA

Найменування характеристики	Одиниці вимірювання	Показник
Точність позиціонування (повторюваність)	мм	$\pm 0,18$
Діапазон робочих температур	°C	0...+45
Споживана потужність	Вт	470
Вантажопідйомність	кг	1200
Кількість ступенів свободи	—	6
Робочий діапазон (осі)	град.	± 360
Рівень шуму	дБ	72,8
Діаметр основи	мм	474
Матеріал	—	алюміній
Маса (вага) моделі	кг	8600

Технічні характеристики робота-маніпулятора KR Quantec PA наведені у таблиці 1.3. Модель KR Quantec PA Arctic не потребує додаткових захисних кожухів для роботи, що забезпечує економію в експлуатації порівняно з іншими роботами. Крім того, відсутня необхідність у підігріванні електроніки, що усуває теплове випромінювання і пов'язані з цим додаткові витрати в промислових морозильних камерах.

Таблиця 1.3

Характеристики робота-маніпулятора KR Quantec PA

Найменування характеристики	Од. вим.	Показник
Точність позиціонування	мм	$\pm 0,14$
Діапазон робочих температур	°C	-30...+5
Споживана потужність	Вт	160

Найменування характеристики	Од. вим.	Показник
Вантажопідйомність	кг	120...180
Кількість ступенів свободи	—	5
Робочий діапазон	град.	±360
Рівень шуму	дБ	70
Діаметр основи	мм	388
Матеріал	—	алюміній
Маса моделі	кг	1075

Технічні параметри робота-маніпулятора IRB 120 компанії ABB наведено у таблиці 1.4. Робот характеризується компактним дизайном та невеликою вагою, що дозволяє значно підвищити максимальну швидкість осей і, як наслідок, продуктивність, скорочуючи час циклу до 25%. Конструкція маніпулятора виконана з легкого алюмінію і оснащена двигунами, що забезпечують більш високі прискорення порівняно з аналогами.

Таблиця 1.4

Технічні характеристики робота-маніпулятора IRB 120 (ABB)

Найменування характеристики	Од. вим.	Показник
Точність позиціонування	мм	±0,18
Діапазон робочих температур	°C	0...+45
Споживана потужність	Вт	470
Вантажопідйомність	кг	1200
Кількість ступенів свободи	—	6
Робочий діапазон	град.	±360
Рівень шуму	дБ	72,8
Діаметр основи	мм	474
Матеріал	—	алюміній
Маса моделі	кг	8600

За результатами аналізу технічних характеристик розглянутих роботів-маніпуляторів сформовано вимоги до проектного пристрою та визначено технічне завдання. Основними недоліками існуючих моделей є велика вага,

значний діаметр основи, високе споживання енергії, підвищений рівень шуму під час роботи, а також висока вартість обладнання.

1.3. Аналіз програмних засобів керування

Сучасні роботи-маніпулятори здебільшого програмуються через вбудовані контролери, які виконують функції керування та контролю процесів у системі. Контролер перетворює програмний код на керуючі сигнали для зовнішніх пристроїв і водночас отримує від них інформацію про стан робочих процесів та навколишнє середовище. Це дозволяє роботам-маніпуляторам самостійно виконувати певні дії та адаптуватися до умов роботи.

Методи програмування залежать від типу та призначення робота. Для прикладу розглянемо навчальну модель Lego Mindstorms Education EV3. Контролер EV3 є програмованим інтелектуальним пристроєм, який керує датчиками та моторами робота (рис. 1.13).

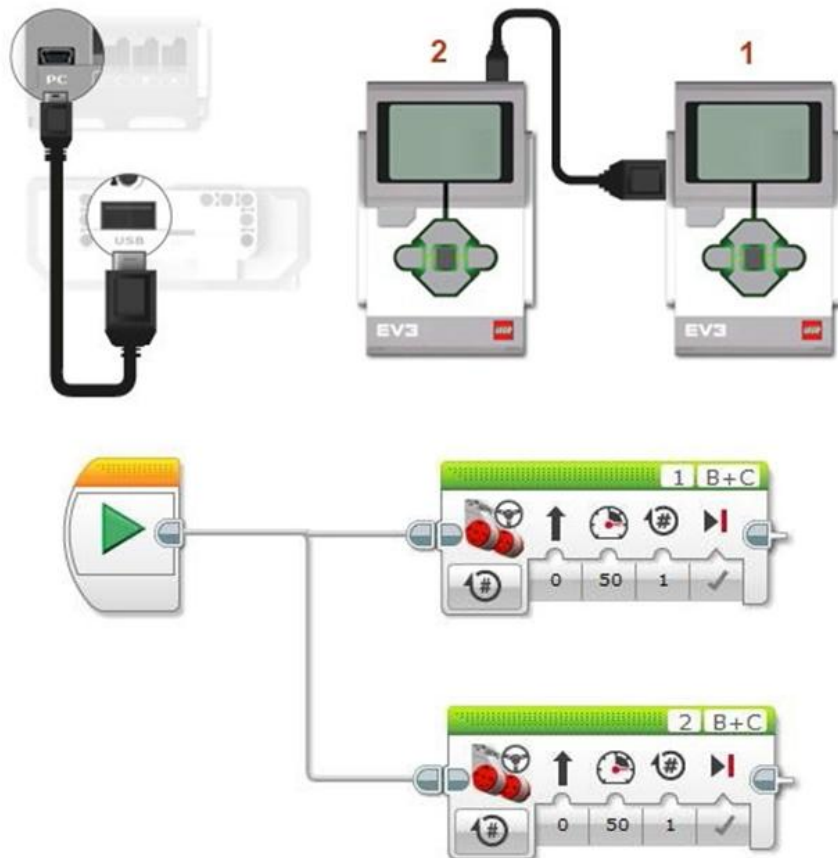


Рис. 1.13. Програмування контролера Lego Mindstorms Education EV3

Основні технічні характеристики контролера EV3:

- Операційна система – Linux;

- Процесор – ARM9;
- FLASH-пам'ять – 16 МБ;
- Оперативна пам'ять – 64 МБ;
- Слот для SD-карт;
- USB 2.0 (підтримка USB Host, можливість підключення до WiFi);
- Bluetooth 2.1;
- Динамік;
- 4 порти введення і 4 порти виведення.

Завдяки інтуїтивно зрозумілому інтерфейсу та готовим програмним блокам, програмування на EV3 дозволяє швидко створювати програми та тестувати робота. Компанія Lego розробила спеціальне програмне середовище, яке містить усі необхідні інструменти для керування та програмування маніпулятора.

Процес програмування починається зі створення нового проекту через меню «Файл» → «Новий проект» → «Програма». Далі вибирається блок дій, наприклад «Рульове керування», і з'єднується з автоматично створеним блоком старту. У налаштуваннях блоку задається кількість обертів або переводиться в режим «Ввімкнути на кількість градусів», а потужність моторів встановлюється вручну. Після виконання всіх налаштувань формується готова програма керування.

Завершальним етапом є перевірка підключення до контролера за допомогою команди «Перегляд портів», яка звіряє номери портів з моторами (зазвичай порти В та С). Після підтвердження правильності підключення програма завантажується та запускається, і робот готовий виконувати задані команди.

Сучасна промислова робототехніка активно розвивається у всьому світі, проте її ефективність неможлива без сучасного програмного забезпечення, яке забезпечує виконання роботом необхідних завдань. Впровадження та експлуатація роботів має низку труднощів.

На великих виробництвах, що виготовляють унікальні вироби, замовлення часто є складними, але одиничними. Перепрограмування робота для зварювання нових деталей може зайняти декілька днів, при цьому основне виробництво доводиться зупиняти. На середніх підприємствах велика номенклатура продукції поєднується з невисокою серійністю, а на малих – нестача кваліфікованого персоналу ускладнює програмування робота-маніпулятора та прогнозування обсягу замовлень. Крім того, підприємства протягом років часто купують роботів різних виробників із різними мовами програмування, пультами керування та контролерами, що робить роботизацію економічно менш вигідною. У той же час вимоги до якості продукції постійно зростають.

Для вирішення цих проблем використовується програмне середовище Ostopuz, яке дозволяє ефективно автоматизувати виробництво як для малих серій, так і для унікальних великих конструкцій. Ostopuz швидко створює повністю функціональні програми для роботів і дозволяє налаштовувати роботу маніпулятора повністю офлайн: встановлення траєкторій у кутах, «пошук дотиком», контроль зварювального шва за довжиною дуги, багатопрохідне зварювання, контроль зіткнень і сингулярності. Це скорочує час зупинки виробничого комплексу для переналагодження до приблизно 10% від часу, необхідного при традиційному програмуванні.

Основні функції Ostopuz для автономного програмування зварювальних роботів-маніпуляторів:

- Створення зварювального шляху у кілька кроків;
- Автоматичне програмування «пошуку дотиком» і корекції траєкторії пальника;
- Програмування різних типів швів, включно з багатопрохідним зварюванням;
- Координація роботи роботів різних виробників;
- Автоматичне визначення точок старту і завершення зварювання, а також кута входу пальника у кут деталі;
- Підключення і управління трьома додатковими зовнішніми осями;

- автономні роботи — інтелектуальні системи, здатні самостійно виконувати складні цикли дій, адаптуватися до змін у робочому середовищі та повністю охоплювати робочу зону конвеєра без участі людини.

На фланці останньої ланки промислових маніпуляторів передбачене стандартизоване кріплення (зазвичай згідно з ISO 9409-1) для монтажу робочих органів (End-of-Arm Tooling, EoAT). Ключовим елементом системи є захватний пристрій, характеристики якого визначають «зону захвату» та типи об'єктів, з якими може взаємодіяти робот.

Між робочим органом і фланцем маніпулятора можуть бути інтегровані додаткові модулі:

Силомоментні сенсори — для контролю зусилля притискання.

Системи технічного зору (камери) — для ідентифікації положення об'єктів.

Пристрої автоматичної зміни інструменту — для швидкої переналадки лінії.

На рис. 1.15 представлено модель комплексного рішення для універсальних роботів (коботів), що забезпечує синхронізацію маніпулятора з периферійним обладнанням: конвеєрними стрічками, автоматичними дверима безпеки та верстатами з ЧПК..



Рис. 1.15. Модель захвату робота-маніпулятора

Для реалізації механічного впливу на об'єкт використовують два основні фізичні принципи, сумісні з екосистемою сучасних маніпуляторів:

Смарт-пневматичний принцип: базується на використанні енергії стисненого повітря. Пневматичні модулі підтримують різні конфігурації пальців:

- Паралельний захват — для роботи з плоскими гранями об'єктів.
- Трьохпальцевий центричний захват — для надійного фіксування циліндричних та сферичних деталей.
- Діапазон сили захвату: регулюється в межах від 230 до 550 Н і більше, що дозволяє працювати з широким спектром вантажів.

Сервоелектричний принцип: забезпечує найвищу точність позиціонування губок та можливість програмного регулювання сили стискання. Це критично важливо при роботі з крихкими або асиметричними об'єктами.

Електронна частина обраного модуля оснащена високошвидкісним мікропроцесором, який реалізує алгоритми зворотного зв'язку. Система автоматично детектує факт підхоплення деталі («Part Detection») та підтримує стабільне зусилля протягом усього циклу переміщення. Це мінімізує ризик випадання об'єкта та підвищує загальну надійність системи за рахунок зменшення кількості зовнішніх датчиків і кабелів.

Для інтеграції захватного пристрою в проект передбачено:

- Механічний монтаж: встановлення на рукоятку через спеціалізовану монтажну пластину.
- Комунікація: підключення через перетворювач інтерфейсу до USB-порту або цифрових входів/виходів контролера робота.
- Програмне забезпечення: використання спеціалізованих плагінів (наприклад, URCaps), що дозволяє налаштовувати параметри захвату безпосередньо з пульта керування роботом.

Впровадження таких комплексних рішень дозволяє значно підвищити продуктивність автоматизованої ділянки, особливо в умовах дрібносерійного виробництва, де необхідна гнучкість при роботі з різними типами заготовок..

Вибір між пневматичним та електричним (сервопривідним) захватом залежить від вимог до гнучкості виробничого процесу та характеристик об'єктів

маніпулювання. Пневматичні системи часто використовуються для однотипних операцій, тоді як електричні — для складних завдань, що потребують делікатного поводження.

Таблиця 1.5

Порівняльна характеристика типів захватних пристроїв

Характеристика	Пневматичний захват	Електричний (серво) захват
Швидкість спрацювання	Дуже висока (відкриття/закриття за частки секунди)	Середня (залежить від швидкості сервоприводу)
Точність позиціонування	Низька (має лише два стани: «відкрито» та «закрито»)	Висока (можливість зупинки губок у будь-якій точці)
Контроль зусилля	Грубий (регулюється тиском повітря в системі)	Прецизійний (програмне налаштування сили стискання)
Ціна	Низька (економічно вигідний для масового виробництва)	Висока (вимагає контролера та сервомоторів)
Гнучкість	Низька (потребує механічної заміни пальців під новий об'єкт)	Висока (адаптація до різних розмірів деталей без переналадки)

Для забезпечення надійного утримання об'єкта під час переміщення необхідно розрахувати мінімальну силу стискання губок маніпулятора. Розрахунок базується на силі тертя, яка повинна компенсувати вагу об'єкта та інерційні навантаження, що виникають при русі робота.

Основна формула для розрахунку необхідної сили стискання F має вигляд:

$$F = \frac{m * (g + a) * S}{n * k}$$

Де:

- m — маса об'єкта маніпулювання (кг);
- g — прискорення вільного падіння (9,81 м/с²);
- a — максимальне прискорення робота при русі (м/с²);
- S — коефіцієнт запасу (зазвичай приймається в межах 1,5- 2,0);

- n — кількість притискних поверхонь (пальців захвату);
- k — коефіцієнт тертя між поверхнею об'єкта та матеріалом губок захвату.

Коефіцієнт тертя k значно варіюється залежно від матеріалу. Наприклад, для пари «сталь-сталь» він становить приблизно $0,1 \dots 0,15$, а для поверхонь з гумовим покриттям може сягати $0,4 \dots 0,6$, що дозволяє використовувати меншу силу стискання для утримання того самого вантажу.

Зазначений у технічних характеристиках діапазон сили захвату для розглянутих моделей (від 230 до 550 Н) дозволяє впевнено маніпулювати об'єктами масою до 10 - 15 кг (залежно від коефіцієнта тертя та прискорень), що повністю покриває потреби більшості операцій із роботами серії UR.

1.5. Використання Arduino у робототехніці

Платформа Arduino є однією з найпопулярніших апаратно-програмних платформ у сучасній робототехніці, що пояснюється її відкритою архітектурою, доступністю для широкого кола користувачів та простотою програмування. Arduino об'єднує мікроконтролер, програмне середовище та набір бібліотек, що забезпечує швидку розробку як навчальних, так і промислових проектів. У контексті створення системи керування роботом-маніпулятором, Arduino виступає ядром, яке координує роботу всіх компонентів: виконавчих механізмів, сенсорів та інтерфейсів взаємодії з користувачем.

Arduino дозволяє ефективно керувати різними електромеханічними компонентами робота-маніпулятора:

- **Сервоприводи** – забезпечують точне позиційне керування ланками маніпулятора. Arduino через ШІМ-сигнали (PWM) задає положення сервомоторів з високою точністю, що важливо для виконання складних маніпуляційних операцій.
- **Крокові двигуни** – використовуються для більш точного регулювання кута повороту та переміщення ланок. Arduino керує кроковими драйверами, задаючи напрямок, частоту кроків і режим роботи.

- **Колекторні двигуни** – часто застосовуються для переміщення бази маніпулятора або виконання силових операцій. Через ШІМ і Н-мости Arduino може регулювати швидкість та напрямок обертання двигунів.

Крім того, Arduino підтримує підключення датчиків різного типу, що дозволяє реалізувати системи з елементами зворотного зв'язку:

- датчики положення та кута (потенціометри, енкодери),
- датчики сили та навантаження,
- датчики відстані (ультразвукові, інфрачервоні),
- сенсори струму та напруги для контролю живлення та безпеки роботи.

За допомогою вбудованих таймерів і ШІМ-виходів (PWM) плата дозволяє точно задавати положення ланок маніпулятора, швидкість і плавність рухів, що є критично важливим для виконання маніпуляційних операцій.

Підключення датчиків положення, струму, сили або відстані дає змогу організувати адаптивне керування, підвищити точність роботи маніпулятора та забезпечити його безпечну експлуатацію. Завдяки аналоговим і цифровим входам Arduino може обробляти сигнали з різних типів сенсорів у режимі реального часу.

Програмна складова Arduino надає широкі можливості для реалізації різноманітних алгоритмів керування. Середовище **Arduino IDE** використовує мову програмування на базі C/C++, що дозволяє розробляти як прості, так і складні системи керування. Для роботи з роботом-маніпулятором можуть бути реалізовані:

- **Позиційне керування** – забезпечує точне встановлення ланок маніпулятора у задані кути, використовуючи сервомотори або крокові двигуни.
- **Керування по траєкторії** – дозволяє рухати маніпулятор по заданій кривій, що необхідно для складних операцій, наприклад, складання або збирання деталей.

- **Адаптивне та зворотньо-зв'язкове керування** – використовуючи дані з сенсорів, система Arduino може коригувати рухи в реальному часі для підвищення точності та безпеки.

Arduino підтримує численні бібліотеки для керування сервоприводами, кроковими двигунами, датчиками, дисплеями та бездротовими модулями, що значно спрощує програмування та зменшує час розробки.

Arduino забезпечує інтеграцію різних інтерфейсів керування:

- **Локальні керуючі пристрої:** кнопки, джойстики, потенціометри;
- **Комп'ютерні програми:** зв'язок через USB, UART, I2C або SPI для передачі команд з ПК;
- **Мобільні пристрої та бездротові системи:** Bluetooth, Wi-Fi, модулі радіозв'язку;
- **Голосове керування** – Arduino здатна приймати сигнали від модулів розпізнавання голосу та виконувати відповідні команди.

Ця гнучкість дозволяє створювати як автономні системи керування, так і інтегровані рішення, які працюють у складі складних робототехнічних комплексів.

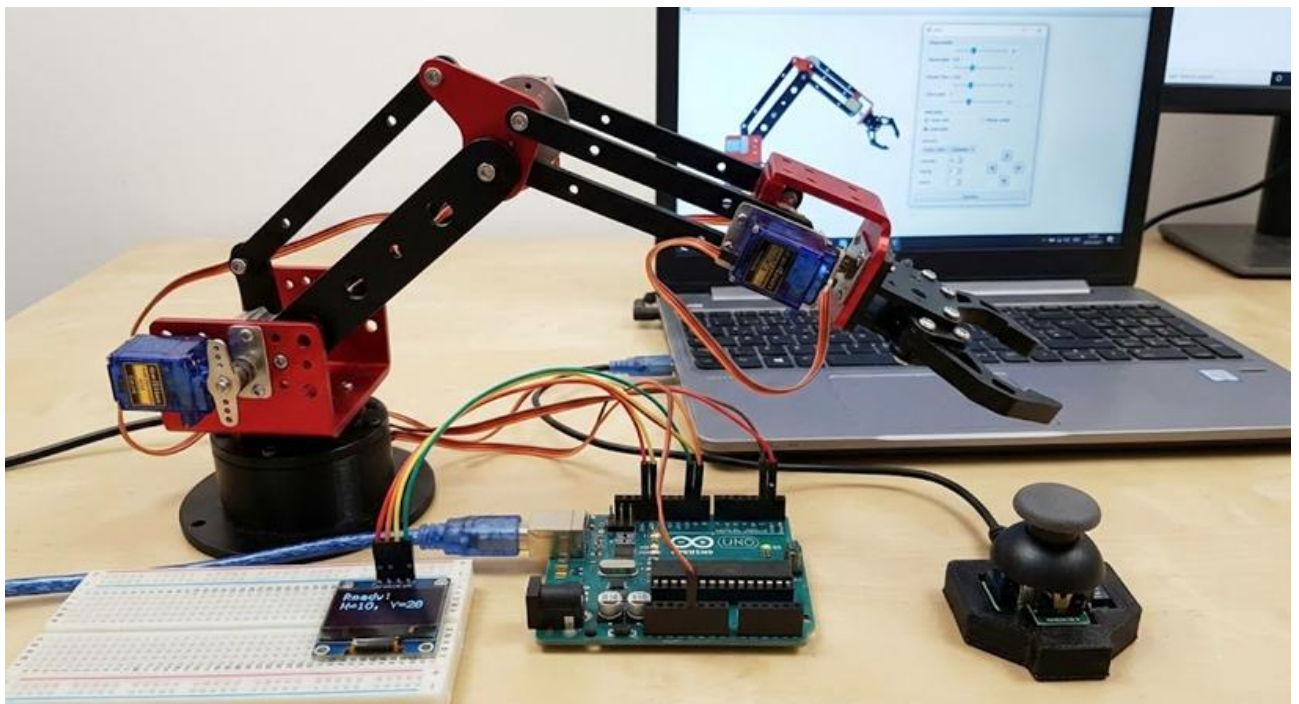


Рис. 1.16. використання платформи Arduino при створенні системи керування роботом-маніпулятором

Таким чином, використання платформи Arduino у робототехніці, зокрема при створенні системи керування роботом-маніпулятором, є доцільним і обґрунтованим. Вона поєднує в собі апаратну гнучкість, програмну простоту та високу масштабованість, що робить її ефективним інструментом як для навчальних і наукових проєктів, так і для практичних інженерних рішень.

Висновки до розділу

Розглянуто провідні моделі маніпуляторів від компаній Universal Robots, Fanuc, KUKA та ABB. Встановлено, що сучасні роботи (зокрема, UR3, UR5, UR10, Fanuc M-2000iA, KUKA KR Quantec) характеризуються високою точністю, значною вантажопідйомністю та адаптивністю до умов виробництва. Однак, суттєвими недоліками цих систем для лабораторного та навчального застосування є їх висока вартість, значні масогабаритні показники, високе енергоспоживання та складність технічного обслуговування.

Порівняльний аналіз технічних характеристик аналогів дозволив сформулювати вимоги до проєктованої системи. Для усунення виявлених недоліків промислових зразків, розроблюваний робот-маніпулятор повинен мати компактну конструкцію, низьке енергоспоживання та доступну вартість, забезпечуючи при цьому достатню точність для виконання базових маніпуляційних операцій.

Встановлено, що для маніпуляторів легкого класу найбільш ефективним є використання сервоелектричних захватів. На відміну від пневматичних систем, вони не потребують компресорного обладнання, забезпечують високу точність позиціонування губок та можливість програмного регулювання зусилля стискання, що є критичним при роботі з об'єктами різної крихкості.

Платформа Arduino є ефективним засобом для створення систем керування роботами-маніпуляторами, забезпечуючи точне керування виконавчими механізмами, інтеграцію сенсорів і гнучке програмування, що робить її доцільною для прототипування, навчання та розробки малих або середніх автоматизованих систем.

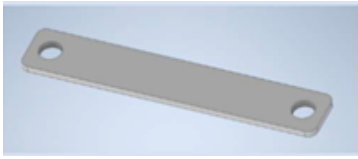
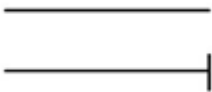


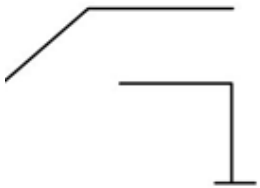
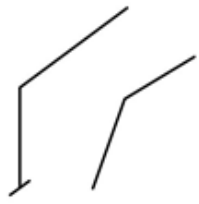
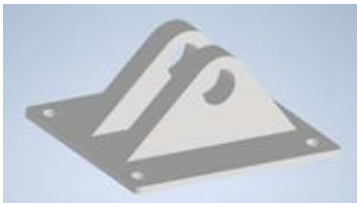
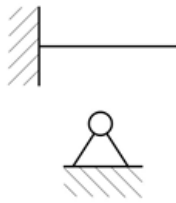

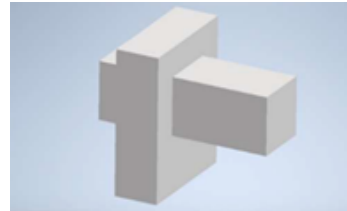
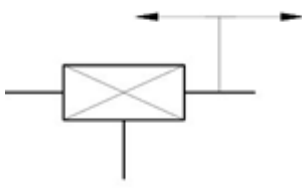
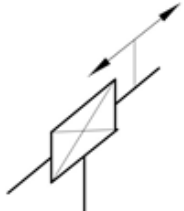
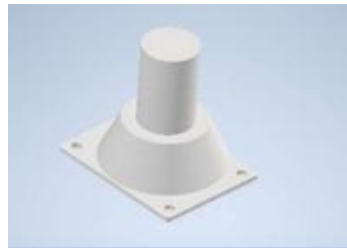
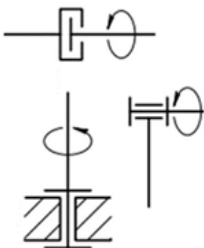

РОЗДІЛ 2. МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ТА КОНСТРУКТОРСЬКЕ ПРОЄКТУВАННЯ МЕХАНІЧНОЇ СИСТЕМИ РОБОТА-МАНІПУЛЯТОРА

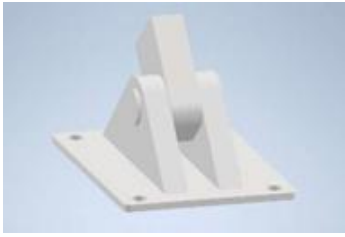
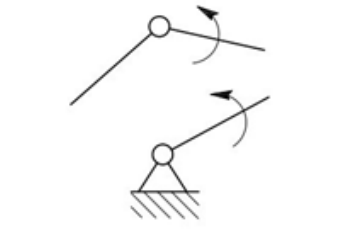
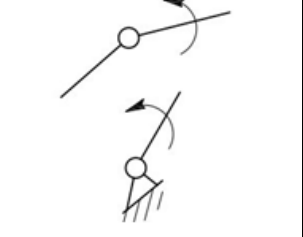

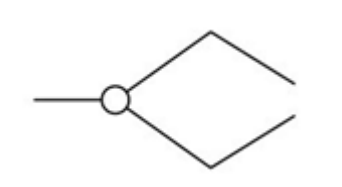
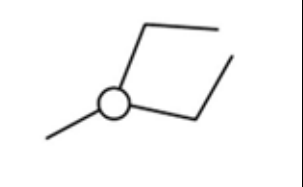
2.1. Типові схеми маніпуляторів

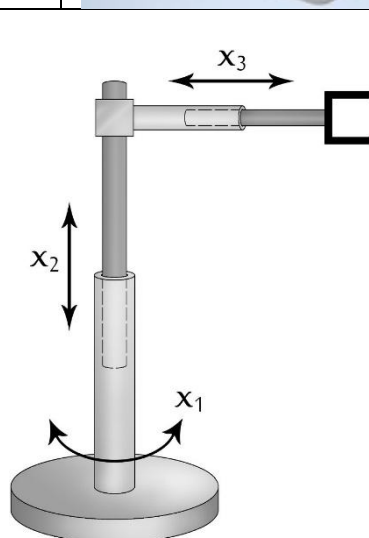
Проектування виконавчих механізмів роботів, зокрема маніпуляторів для захоплення та транспортування об'єктів, передбачає розробку кінематичних схем, які формалізують структуру рухомих ланок та характер їх взаємодії. Типові елементи кінематичних схем маніпуляторів наведені в таблиці 2.1.

Таблиця 2.1

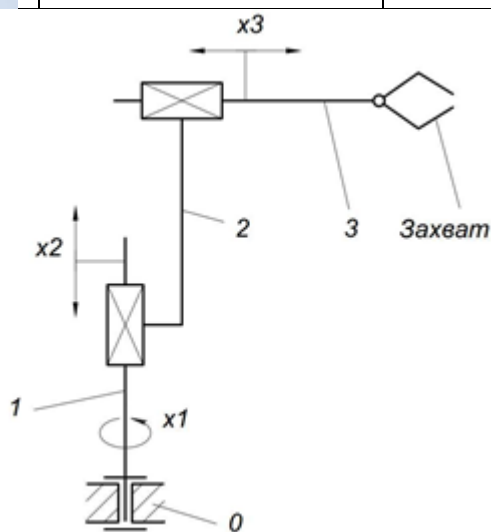
Типові елементи кінематичних схем маніпуляторів[7]

Назва	Приклад	Позначення на схемі	
		плоский	просторовий
Ланка рухома: прямолінійна			
Складної форми			
Ланка нерухома (стійка, основа)			
Поступальна кінематична пара			
Обертальна кінематична пара — ротація			

Назва	Приклад	Позначення на схемі	
		плоский	просторовий
Обертальна кінематична пара згинання			
Захватний пристрій			



а)



б)

Рис. 2.1. Приклад загального вигляду простого маніпулятора (а) та його кінематичної схеми (б)

На рисунку 2.1 представлено загальний вигляд простого маніпулятора та відповідну кінематичну схему. Маніпулятор включає нерухому основу (стійку) 0, рухомі ланки 1–3 та захватний пристрій. Кінематичні зв'язки між ланками реалізовані у вигляді однієї обертальної пари ротаційного типу та двох поступальних пар.

Ступені рухливості ланок маніпулятора, що забезпечують перенесення об'єкта у просторі, реалізуються через поступальні або обертальні рухи відносно осей відповідних кінематичних пар. Величини переміщень (координати положень ланок) визначаються відносно цих осей і вимірюються в лінійних (м) або кутових (рад., град.) одиницях. Таким чином, осі кінематичних пар

формують координатну систему, за якою програмуються рухи ланок. У сукупності ці осі утворюють базову систему координат, у якій доцільно описувати переносні рухи маніпулятора.

Види кінематичних пар (поступальні або обертальні), їх послідовність у кінематичному ланцюгу та взаємне розташування осей сусідніх пар у просторі (перпендикулярне, паралельне або під довільним кутом) визначають базову систему координат та відповідну структурну схему маніпулятора. Від виду базової системи координат залежить не лише форма та об'єм робочої зони, але й функціональні можливості маніпуляційної системи робота.

Для стислого опису структурних схем маніпулятора використовують умовні позначення: П – поступальна кінематична пара, В – обертальна кінематична пара; \perp , \parallel – перпендикулярне та паралельне взаємне розташування осей сусідніх пар.

Для більшості роботів маніпуляційна система містить три переносні ступені рухливості, утворені трьома кінематичними парами, що забезпечують об'ємну робочу зону. Будь-яка комбінація пар ППП, ВПП, ВВП або ВВВ дозволяє реалізувати об'ємну зону при правильному взаємному розташуванні осей пар.

Серед великої кількості можливих схем у практиці сучасних роботів-маніпуляторів найчастіше застосовують п'ять типових схем (рис. 2.2) [5]:

- 1) Схема П \perp П \perp П (рис. 2.2, а) реалізує три поступальні ступені рухливості у взаємно перпендикулярних напрямках. Положення захватного органу визначається координатами x , y , z , що відповідає прямокутній (декартовій) базовій системі координат. Форма робочої зони – паралелепіпед. Переваги цієї схеми – простота програмування та точне позиціонування робочого органу, що робить її придатною для високоточних операцій збирання та обробки. Недоліком є великий простір, який займає робот при порівняно невеликому робочому об'ємі.
- 2) Схема В \parallel П \perp П (рис. 2.2, б) поєднує одну обертальну та дві поступальні ступені рухливості. Положення робочого органу визначається кутовою

координатою α та двома лінійними – ρ і z , утворюючи циліндричну базову систему координат. Форма робочої зони – частина порожнистого циліндра. Порівняно з попередньою схемою ця конфігурація забезпечує більший об'єм робочої зони при однаковій величині поступальних переміщень. Недолік – суттєве збільшення висоти робота при великій робочій зоні, що веде до зростання маси маніпулятора.

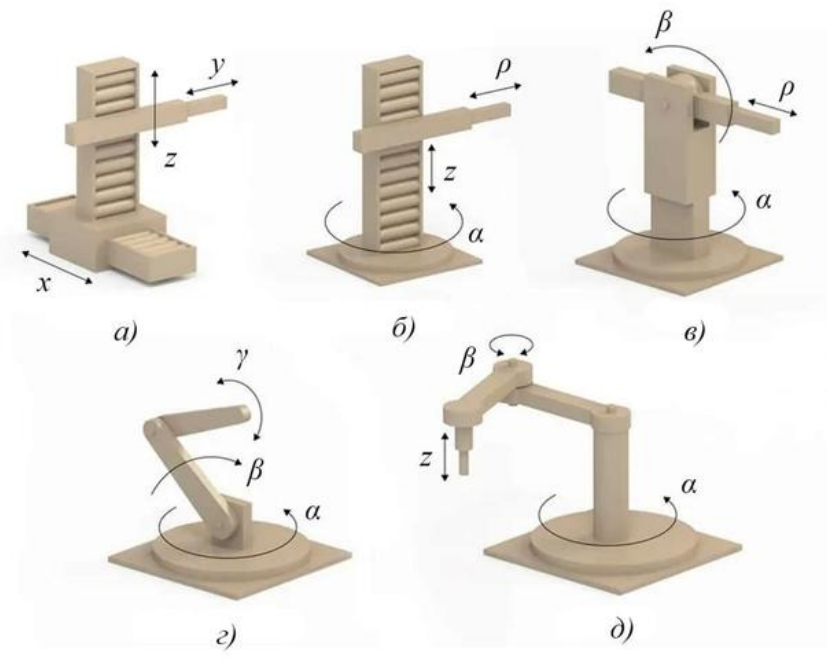


Рис. 2.2. Типові схеми маніпуляторів [7]

- 3) Схема $V \perp V \perp P$ (рис. 2.2, в) забезпечує дві обертальні та одну поступальну ступені рухливості (ротація щодо вертикальної осі та згинання у вертикальній площині). Положення робочого органу задається двома кутовими координатами α і β та однією лінійною ρ , що формує сферичну (полярну) базову систему координат. Форма робочої зони – частина порожньої кулі. Перевага цієї схеми полягає у тому, що при відносно невеликій висоті колони можна досягти значної висоти робочої зони, що підвищує універсальність використання робота.
- 4) Схема $V \perp V \parallel V$ (рис. 2.2, г) включає лише обертальні ступені рухливості: одна пара забезпечує ротацію першої
- 5) ланки навколо вертикальної осі, а дві інші – згинання наступних ланок у вертикальній площині відносно двох паралельних горизонтальних осей.

Положення робочого органу задається трьома кутовими координатами α , β , γ . Така система координат називається кутовою або ангулярною сферичною (іноді – складною сферичною), оскільки складну об’ємну форму робочої зони можна розбити на окремі елементи сферичних поверхонь з різними радіусами. Перевага цієї схеми – компактність, недолік – складність керування та конструкції маніпулятора через необхідність застосування складних приводів, врівноважувальних механізмів тощо.

- б) Схема В || В || П (рис. 2.2, д) включає три кінематичні пари з паралельними осями, що формують дві обертальні (згинання у горизонтальній площині) та один поступальний ступінь рухливості. Положення робочого органу визначають дві кутові координати α і β та одна лінійна z . Система таких координат, де об’ємна форма робочої зони складається з окремих циліндричних елементів, називається ангулярною циліндричною або складною циліндричною базовою системою координат. Цю схему прийнято називати SCARA. Її перевага – компактність та здатність забезпечувати точне позиціонування захватного органу у необхідних положеннях.

2.2. Математична модель робота маніпулятора для системи управління

З метою досягнення високої точності та оперативності роботи автоматичної системи керування маніпулятором робота доцільно розробити його математичну модель [21]. На цьому етапі пропонується використати метод Денавіта–Хартенберга у поєднанні з рівняннями Лагранжа [21]. Застосування зазначених методів дає змогу вивести рівняння динаміки ланок маніпулятора та здійснити поглиблений аналіз його руху й поведінки [21]. Отримана математична модель створює підґрунтя для проєктування ефективної системи керування й оптимального добору параметрів, що сприятиме підвищенню продуктивності та точності керування маніпулятором [21].

Характерним прикладом робота-маніпулятора, що забезпечує відтворення основних типів рухів людської руки, є чотириланковий робот, зображений на рис. 2.3 [21]. До його основних елементів належать база, плече, лікоть і захват [21]. Кінематичну структуру цього робота наведено на рис. 2.3, де вона подана у вигляді системи з чотирьох ланок a_1, a_2, a_3, a_4 [21].

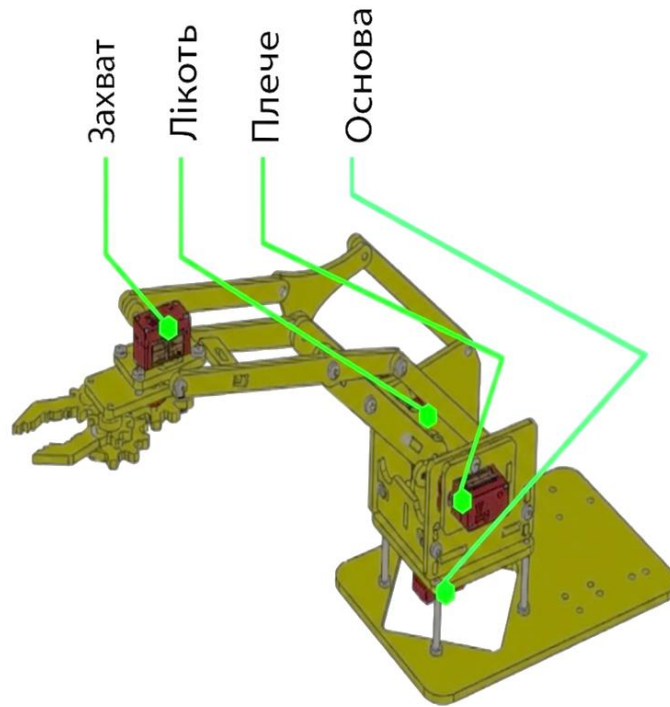


Рис. 2.3. Конфігурація чотирьохланкового робота

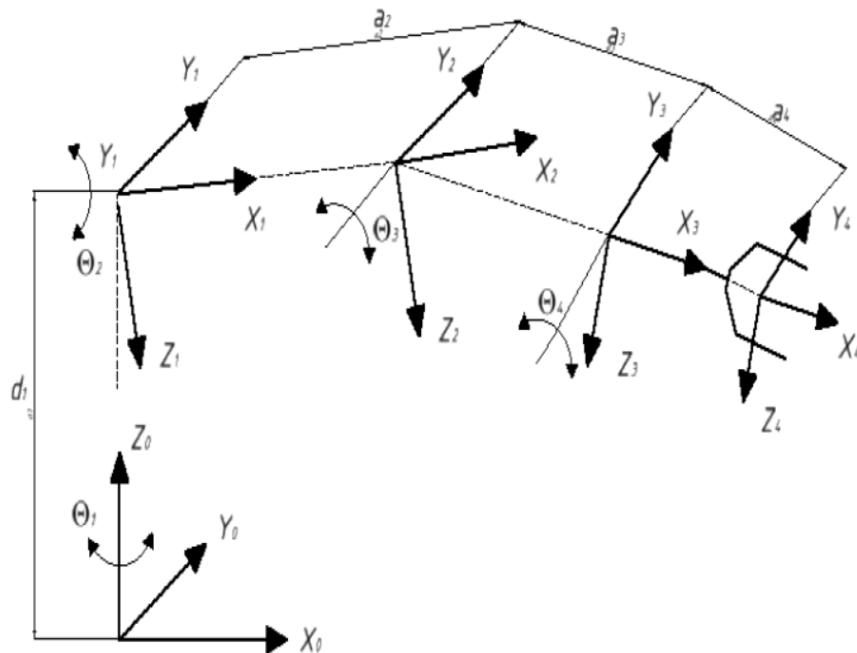


Рис. 2.4. Кінематична схема 4-х ланкового робота

Згідно з кінематичною схемою, наведеною на рис. 2.4, визначено основні параметри систем координат ланок, які подано в таблиці 2.2 [21].

Таблиця 2.2

Параметри системи координат

№ ланки	θ_i	α_i	a_i	d_i
1	θ_1	90°	0	d_1
2	θ_2	0	a_1	0
3	θ_3	0	a_2	0
4	θ_4	0	a_3	0

Де: a_i – довжини ланок [21]; α_i – кут між осями z_i та y_i навколо осі x_i [21]; θ_i – кут обертання навколо осі z_i [21]; d_i – відстань вздовж осі z_i [21].

Для опису динамічного рівняння використаємо матрицю однорідного перетворення, що відповідає однорідному руху [21].

$$A_i^{i-1} = \begin{vmatrix} \cos \theta_i & -\cos \alpha_i \sin \theta_i & \sin \alpha_i \sin \theta_i & \alpha_i \cos \theta_i \\ \sin \theta_i & \cos \alpha_i \cos \theta_i & -\sin \alpha_i \cos \theta_i & \alpha_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{vmatrix} \quad (2.1)$$

Якщо координати точки r_i відомі в i -й системі координат, то за допомогою матриці перетворення (1) можна обчислити координати цієї точки в системі координат, пов'язаній з попередньою, тобто в $(i-1)$ -й системі координат [21]:

$$r_i^{i-1} = A_i^{i-1} \cdot r_i^i \quad (2.2)$$

Де:

$r_i^{i-1} = |x_{i-1} \ y_{i-1} \ z_{i-1} \ 1|^T$ – координати точки r_i в $(i-1)$ -й системі координат [21];

$r_i^i = |x_i \ y_i \ z_i \ 1|^T$ – координати точки r_i в i -й системі координат [21].

Перемноження матриць перетворення для суміжних систем координат дозволяє отримати узагальнену матрицю перетворення, яка встановлює зв'язок між системою координат i -тої ланки та базовою системою координат маніпулятора [21].

Положення i -тої ланки відносно основи маніпулятора визначається таким чином [21]:

$$A_i^0 = A_1^0 \cdot A_2^1 \cdot A_3^2 \cdot A_4^3 \cdot \dots \cdot A_i^{i-1} \quad (2.3)$$

Для прийнятих систем координат, на основі даних таблиці 1 та матриці A_i^{i-1} (1), визначимо однорідні матриці перетворення для маніпулятора робота [21].

$$A_1^0 = \begin{vmatrix} \cos \theta_1 & 0 & \sin \theta_1 & 0 \\ \sin \theta_1 & 0 & \cos \theta_1 & 0 \\ 0 & -1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{vmatrix} \quad (2.4)$$

$$A_2^1 = \begin{vmatrix} \cos \theta_2 & -\sin \theta_2 & 0 & \alpha_2 \cos \theta_2 \\ \sin \theta_2 & \cos \theta_2 & 0 & \alpha_2 \sin \theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} \quad (2.5)$$

$$A_3^2 = \begin{vmatrix} \cos \theta_3 & -\sin \theta_3 & 0 & \alpha_3 \cos \theta_3 \\ \sin \theta_3 & \cos \theta_3 & 0 & \alpha_3 \sin \theta_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} \quad (2.6)$$

$$A_4^3 = \begin{vmatrix} \cos \theta_4 & -\sin \theta_4 & 0 & \alpha_4 \cos \theta_4 \\ \sin \theta_4 & \cos \theta_4 & 0 & \alpha_4 \sin \theta_4 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} \quad (2.7)$$

Після визначення матриць $A_1^0, A_2^1, A_3^2, A_4^3$ наступним етапом є їх поетапне перемноження з метою спрощення загального перетворення, що виконується у такій послідовності [21]:

$$A_4^2 = A_3^2 \cdot A_4^3; \quad A_4^1 = A_2^1 \cdot A_4^2; \quad A_4^0 = A_1^0 \cdot A_4^1 \quad (2.8)$$

Для формування диференціальних рівнянь руху маніпулятора скористаємося рівняннями Лагранжа у векторно-матричному вигляді, які можна записати таким чином [21]:

$$\tau(t) = D(q(t)) \cdot \ddot{q}(t) + h(q(t), \dot{q}(t)) + C(q(t)) \quad (2.9)$$

де $\tau(t)$ — вектор узагальнених сил (моментів), що створюються силовими приводами у зчленуваннях маніпулятора [21];

$q(t)$ — вектор узагальнених координат [21];

$\dot{q}(t)$ — вектор узагальнених швидкостей [21];

$\ddot{q}(t)$ — вектор узагальнених прискорень [21];

$D(q)$ — симетрична матриця інерції розміром $n \times n$ [21];

$h(q, \dot{q})$ — вектор коріолісових та відцентрових сил [21];

$C(q)$ — вектор гравітаційних сил (моментів) [21].

Відповідно до наведеного опису, векторно-матрична модель чотириланкового робота має такий вигляд [21]:

$$\begin{bmatrix} D_{11} & D_{12} & D_{13} & D_{14} \\ D_{21} & D_{22} & D_{23} & D_{24} \\ D_{31} & D_{32} & D_{33} & D_{34} \\ D_{41} & D_{42} & D_{43} & D_{44} \end{bmatrix} \cdot \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \ddot{\theta}_3 \\ \ddot{\theta}_4 \end{bmatrix} + \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \end{bmatrix} + \begin{bmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \\ \tau_4 \end{bmatrix} \quad (2.10)$$

де D_{ii} — коефіцієнти, що визначають залежність сил і моментів, які діють у зчленуваннях маніпулятора, від прискорень узагальнених змінних [21];

h_i — коефіцієнти, які характеризують вплив сил і моментів у зчленуваннях на швидкості зміни узагальнених змінних [21];

C_i — коефіцієнти, що враховують дію сили тяжіння на кожну ланку маніпулятора [21].

Елементи матриці DDD визначаються відповідно до виразу (2.11) [21].

$$D_{ik} = \sum_{j=\max(i,k)}^n \text{Tr}(U_{jk} J_j U_{ji})^T \quad (2.11)$$

де $i, k = 1, 2, 3, 4$; $n = 4$ — кількість ланок [21].

Для кінематичної системи, що розглядається, рівняння (3.11) має вигляд [21]:

$$\begin{aligned} D_{11} &= \text{Tr}(U_{11} J_1 U_{11})^T + \text{Tr}(U_{21} J_2 U_{21})^T + \text{Tr}(U_{31} J_3 U_{31})^T + \text{Tr}(U_{41} J_4 U_{41})^T \\ D_{12} &= D_{21} = \text{Tr}(U_{21} J_2 U_{21})^T + \text{Tr}(U_{32} J_3 U_{31})^T; \\ D_{13} &= D_{31} = \text{Tr}(U_{33} J_3 U_{31})^T; \\ D_{22} &= \text{Tr}(U_{22} J_2 U_{22})^T + \text{Tr}(U_{32} J_3 U_{31})^T; \\ D_{23} &= D_{32} = \text{Tr}(U_{33} J_3 U_{32})^T; \\ D_{33} &= \text{Tr}(U_{33} J_3 U_{33})^T; \end{aligned} \quad (2.12)$$

Матриця U_{ji} характеризує зміну положення i -тої ланки, зумовлену рухом у j -му зчленуванні, та визначається відповідно до виразу (2.13) [21]:

$$U_{ji} = \frac{\partial A_i}{\partial q_j} = \{A_{j-1}^0 \cdot Q_j \cdot A_i^{j-1}, \text{ якщо } j \leq i\} \quad (2.13)$$

де Q_i – матриця для обертового та поступального зчленувань [21]. Звідси отримуємо, що:

$$\begin{aligned} U_{11} &= Q_1 \cdot A_1^0; \\ U_{21} &= Q_1 \cdot A_2^0; \\ U_{31} &= Q_1 \cdot A_3^0; \\ U_{22} &= A_1^0 \cdot Q_1 \cdot A_2^0; \\ U_{32} &= A_1^0 \cdot Q_2 \cdot A_3^1; \\ U_{33} &= A_2^0 \cdot Q_3 \cdot A_3^2; \end{aligned} \quad (2.14)$$

Елементи матриці J_i визначаються за відомими співвідношеннями, зокрема шляхом обчислення лінійної та кутової швидкостей кінцевого ефектора робота [21].

Для знаходження складових, що характеризують відцентрові та коріолісові сили, скористаємося виразом (2.15) [21].

$$h_i = \sum_{k=1}^n \sum_{m=1}^n h_{ikm} \dot{q}_k \dot{q}_m \quad (2.15)$$

де $i = 1, 2, 3, 4$; $n = 4$ [21].

Тоді вираз (2.16) матиме наступний вигляд [21]:

$$\begin{aligned} h_1 &= \sum_{k=1}^4 \sum_{m=1}^4 h_{1km} \dot{q}_k \dot{q}_m \\ &= h_{111} \dot{\theta}_1 \dot{\theta}_1 + h_{112} \dot{\theta}_1 \dot{\theta}_2 + h_{113} \dot{\theta}_1 \dot{\theta}_3 + h_{114} \dot{\theta}_1 \dot{\theta}_4 + h_{121} \dot{\theta}_2 \dot{\theta}_1 \\ &+ h_{122} \dot{\theta}_2 \dot{\theta}_2 + h_{123} \dot{\theta}_2 \dot{\theta}_3 + h_{124} \dot{\theta}_2 \dot{\theta}_4 + h_{131} \dot{\theta}_3 \dot{\theta}_1 + h_{132} \dot{\theta}_3 \dot{\theta}_2 \\ &+ h_{133} \dot{\theta}_3 \dot{\theta}_3 + h_{134} \dot{\theta}_3 \dot{\theta}_4 + h_{141} \dot{\theta}_4 \dot{\theta}_1 + h_{142} \dot{\theta}_4 \dot{\theta}_2 + h_{143} \dot{\theta}_4 \dot{\theta}_3 \\ &+ h_{144} \dot{\theta}_4 \dot{\theta}_4; \end{aligned}$$

$$\begin{aligned}
h_2 &= \sum_{k=1}^4 \sum_{m=1}^4 h_{2km} \dot{q}_k \dot{q}_m \\
&= h_{211} \dot{\theta}_1 \dot{\theta}_1 + h_{212} \dot{\theta}_1 \dot{\theta}_2 + h_{213} \dot{\theta}_1 \dot{\theta}_3 + h_{214} \dot{\theta}_1 \dot{\theta}_4 + h_{221} \dot{\theta}_2 \dot{\theta}_1 \\
&+ h_{222} \dot{\theta}_2 \dot{\theta}_2 + h_{223} \dot{\theta}_2 \dot{\theta}_3 + h_{224} \dot{\theta}_2 \dot{\theta}_4 + h_{231} \dot{\theta}_3 \dot{\theta}_1 + h_{232} \dot{\theta}_3 \dot{\theta}_2 \\
&+ h_{233} \dot{\theta}_3 \dot{\theta}_3 + h_{234} \dot{\theta}_3 \dot{\theta}_4 + h_{241} \dot{\theta}_4 \dot{\theta}_1 + h_{242} \dot{\theta}_4 \dot{\theta}_2 + h_{243} \dot{\theta}_4 \dot{\theta}_3 \\
&+ h_{244} \dot{\theta}_4 \dot{\theta}_4; \\
h_3 &= \sum_{k=1}^4 \sum_{m=1}^4 h_{3km} \dot{q}_k \dot{q}_m \\
&= h_{311} \dot{\theta}_1 \dot{\theta}_1 + h_{312} \dot{\theta}_1 \dot{\theta}_2 + h_{313} \dot{\theta}_1 \dot{\theta}_3 + h_{314} \dot{\theta}_1 \dot{\theta}_4 + h_{321} \dot{\theta}_2 \dot{\theta}_1 + h_{322} \dot{\theta}_2 \dot{\theta}_2 \\
&+ h_{323} \dot{\theta}_2 \dot{\theta}_3 + h_{324} \dot{\theta}_2 \dot{\theta}_4 + h_{331} \dot{\theta}_3 \dot{\theta}_1 + h_{332} \dot{\theta}_3 \dot{\theta}_2 + h_{333} \dot{\theta}_3 \dot{\theta}_3 + h_{334} \dot{\theta}_3 \dot{\theta}_4 \\
&+ h_{341} \dot{\theta}_4 \dot{\theta}_1 + h_{342} \dot{\theta}_4 \dot{\theta}_2 + h_{343} \dot{\theta}_4 \dot{\theta}_3 + h_{344} \dot{\theta}_4 \dot{\theta}_4; \\
h_4 &= \sum_{k=1}^4 \sum_{m=1}^4 h_{4km} \dot{q}_k \dot{q}_m = h_{411} \dot{\theta}_1 \dot{\theta}_1 + h_{412} \dot{\theta}_1 \dot{\theta}_2 + h_{413} \dot{\theta}_1 \dot{\theta}_3 + \\
&h_{414} \dot{\theta}_1 \dot{\theta}_4 + h_{421} \dot{\theta}_2 \dot{\theta}_1 + h_{422} \dot{\theta}_2 \dot{\theta}_2 + h_{423} \dot{\theta}_2 \dot{\theta}_3 + \\
&h_{424} \dot{\theta}_2 \dot{\theta}_4 + h_{431} \dot{\theta}_3 \dot{\theta}_1 + h_{432} \dot{\theta}_3 \dot{\theta}_2 + h_{433} \dot{\theta}_3 \dot{\theta}_3 + \\
&h_{434} \dot{\theta}_3 \dot{\theta}_4 + h_{441} \dot{\theta}_4 \dot{\theta}_1 + h_{442} \dot{\theta}_4 \dot{\theta}_2 + h_{443} \dot{\theta}_4 \dot{\theta}_3 + h_{444} \dot{\theta}_4 \dot{\theta}_4; \tag{2.16}
\end{aligned}$$

Коефіцієнти h_{ikm} знаходимо з виразу [21]:

$$h_{ikm} = \sum_{j=\max(i,k)}^n \text{Tr}(U_{jk} J_j U_{ji})^T \tag{2.17}$$

де $i, k, m = 1, 2, 3, 4$; $n = 4$ [21].

Після, того визначаємо значення U_{ikm} за виразом (2.18) [21]:

$$U_{ikm} = \begin{cases} A_{k-1}^0 \cdot Q_k \cdot A_{m-1}^{k-1} \cdot Q_m \cdot A_m^{m-1}, \text{ якщо } j \geq m \geq k \\ A_{m-1}^0 \cdot Q_m \cdot A_{k-1}^{m-1} \cdot Q_k \cdot A_j^{k-1}, \text{ якщо } j \geq k \geq m \\ 0, \text{ якщо } j < k \text{ або } j < m \end{cases} \tag{2.18}$$

Вплив гравітаційних сил (моментів), що діють на маніпулятор, визначається за формулою (2.19) [21]:

$$C_i = \sum_{j=1}^n (-m_j g U_{ji} \bar{r}_j^j) \quad (3.19)$$

де $i=1,2,3,4$; $n=4$ [21]; g — вектор-гравітація у базовій системі координат [21]; \bar{r}_j^j — радіус-вектор центра мас i -тої ланки [21].

У нашому випадку вирази (2.19) для обчислення гравітаційних моментів набудуть наступного вигляду [21]:

$$\left. \begin{aligned} C_1 &= -(m_1 g U_{11} \bar{r}_1^1 + m_2 g U_{21} \bar{r}_2^2 + m_3 g U_{31} \bar{r}_3^3 + m_4 g U_{41} \bar{r}_4^4) \\ C_2 &= -(m_2 g U_{22} \bar{r}_2^2 + m_3 g U_{32} \bar{r}_3^3 + m_4 g U_{42} \bar{r}_4^4) \\ C_3 &= -(m_3 g U_{33} \bar{r}_3^3 + m_4 g U_{43} \bar{r}_4^4) \\ C_4 &= -(m_4 g U_{44} \bar{r}_4^4) \end{aligned} \right\} \quad (2.20)$$

Вектор узагальнених моментів $\tau(t)$, які створюються зовнішніми впливами в зчленуваннях, має вигляд [21]:

$$\tau(t) = \begin{bmatrix} M_1 - M_{f1} \\ M_2 - M_{f2} \\ M_3 - M_{f3} \\ M_4 - M_{f4} \end{bmatrix} \quad (2.21)$$

де M_1, M_2, M_3, M_4 — моменти, що діють на ланки механізму з боку приводів переміщення [21]; $M_{f1}, M_{f2}, M_{f3}, M_{f4}$ — моменти зовнішнього в'язкого тертя у відповідних зчленуваннях [21].

Підставивши знайдені вирази (2.12), (2.16), (2.20) та (2.21) у рівняння (2.10) та здійснивши необхідні перетворення, отримуємо математичну модель, що описує динаміку чотириланкового маніпулятора робота [21]. На основі цієї моделі та з використанням методів теорії автоматичного керування можна здійснити синтез системи управління роботом-маніпулятором [21].

2.3. Проектування 3-D моделі робота маніпулятора

Набір 4DOF Arm Kit (чотири ступені свободи) є зменшеною моделлю промислового робота-маніпулятора [15]. Даний робототехнічний комплект належить до опенсорних проєктів, а вся інформація щодо його конструкції та програмного забезпечення знаходиться у відкритому доступі в мережі Інтернет.

Це забезпечує зручні умови для розроблення й апробації програмних рішень, які в подальшому можуть бути адаптовані та застосовані на промислових роботах, зокрема типу IRB 140.

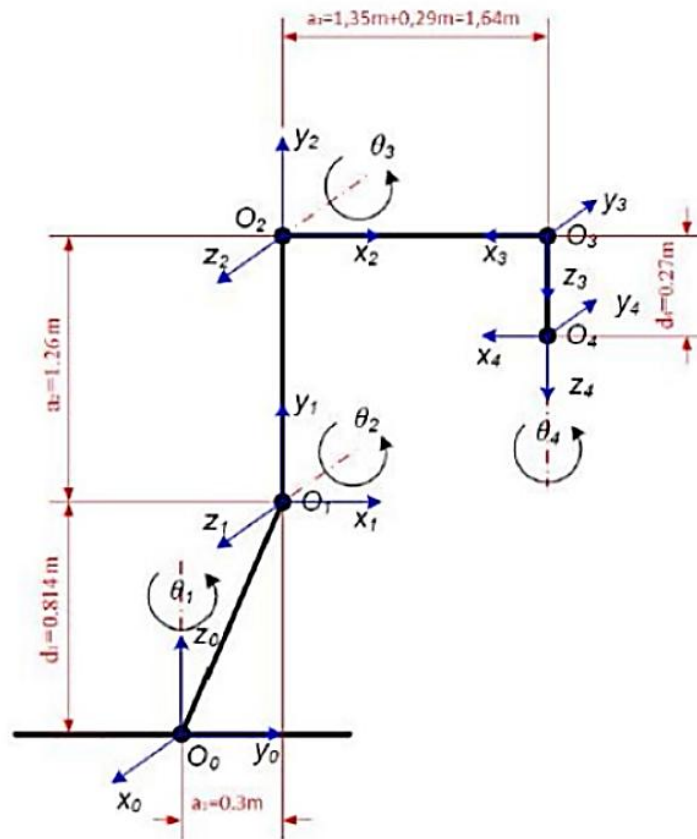


Рис. 2.5. Параметри a_i і d_i

Для глибшого розуміння конструкції робота доцільно проаналізувати можливі типи побудови аналогічних маніпуляторів. На цій основі була сформована структурна схема, наведена на рисунку 2.6, яка відображає основні функціональні вузли робота, їх призначення та характер взаємодії між ними. Згідно зі схемою, маніпулятор має два «суглоби», рух яких здійснюється в одній площині. Кінематичний рух цих «суглобів» реалізується шляхом повороту відповідних ланок на кути φ_1 і φ_2 . Переміщення маніпулятора в тривимірному просторі забезпечується обертанням механічної руки навколо осі y на кут φ_3 .

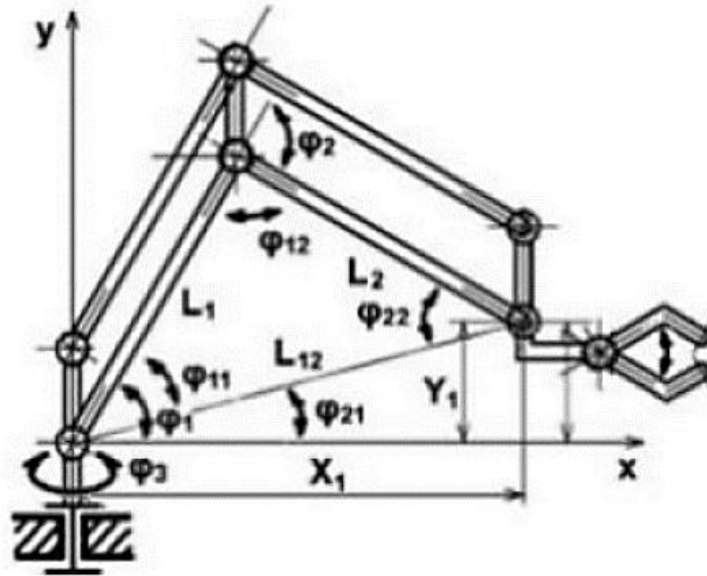


Рис. 2.6. Структурна схема маніпулятора [15]

Для розроблення тривимірної моделі маніпулятора (рис. 2.7) було застосовано програмний САПР-комплекс Autodesk Inventor (Autodesk, Inc., США), який призначений для автоматизації процесів конструкторської та технологічної підготовки виробництва на промислових підприємствах. Даний програмний продукт надає можливість виконувати анімацію рухів моделі, що дозволяє на етапі проєктування виявляти та усувати потенційні конструктивні недоліки. Окрім цього, вже під час моделювання можна підібрати необхідні кріпильні елементи, використовуючи вбудовану бібліотеку стандартних компонентів Inventor. Усі механічні частини лабораторного маніпулятора, за винятком кріпильних елементів, були виготовлені методом 3D-друку.

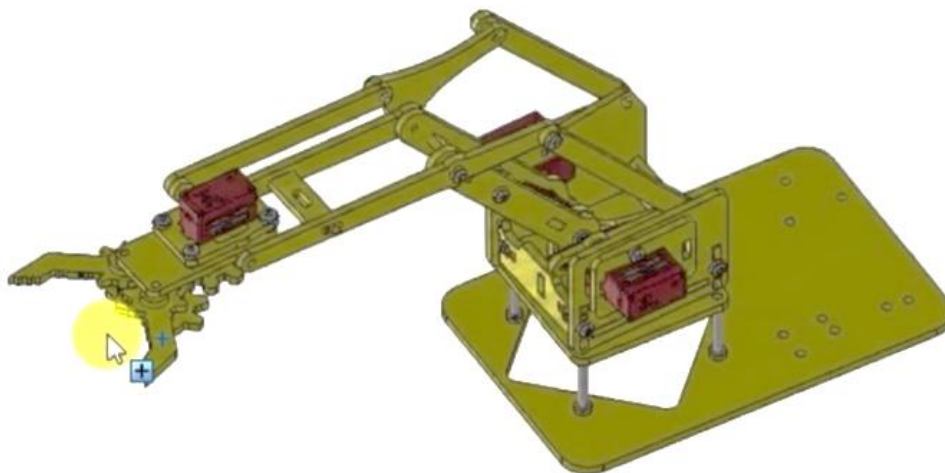


Рис. 2.7. 3-D модель прототипу маніпулятора

На рисунку 2.8 наведено каркас деталей, призначених для складання фізичної моделі робота.

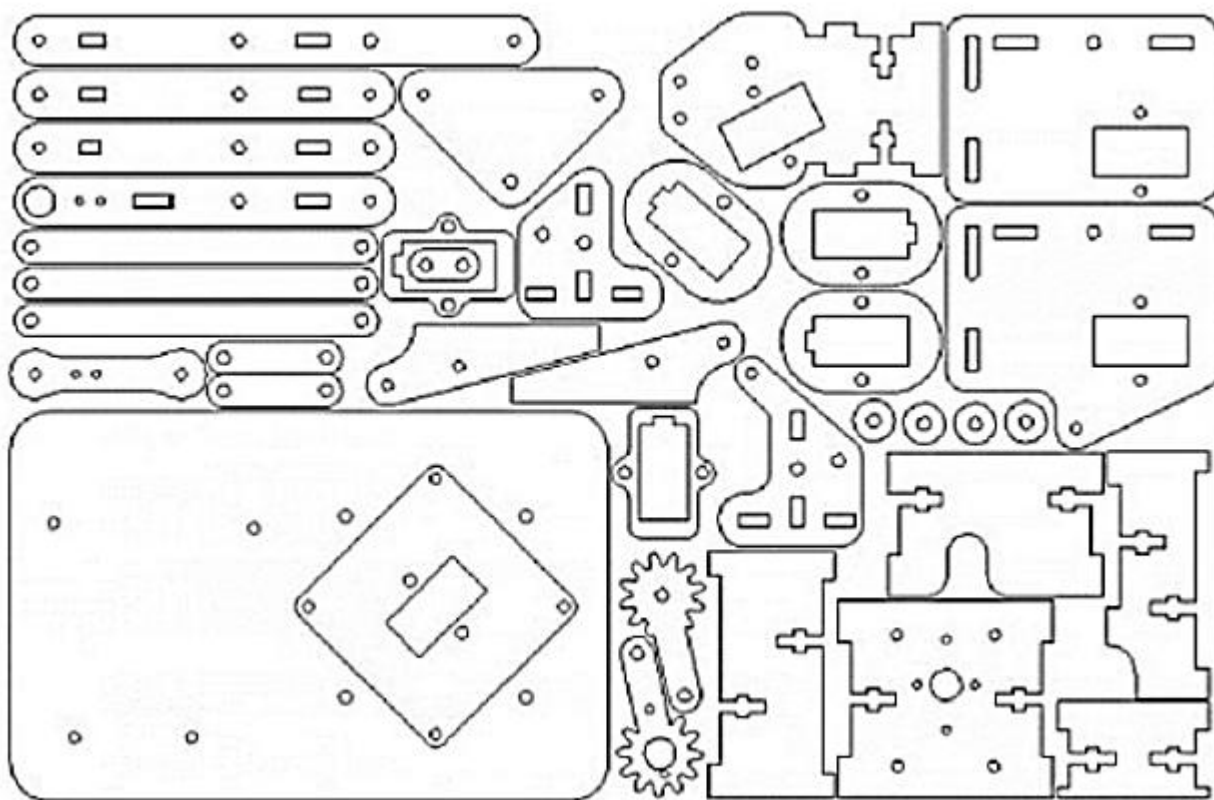


Рис. 2.8. Набір деталей маніпулятора

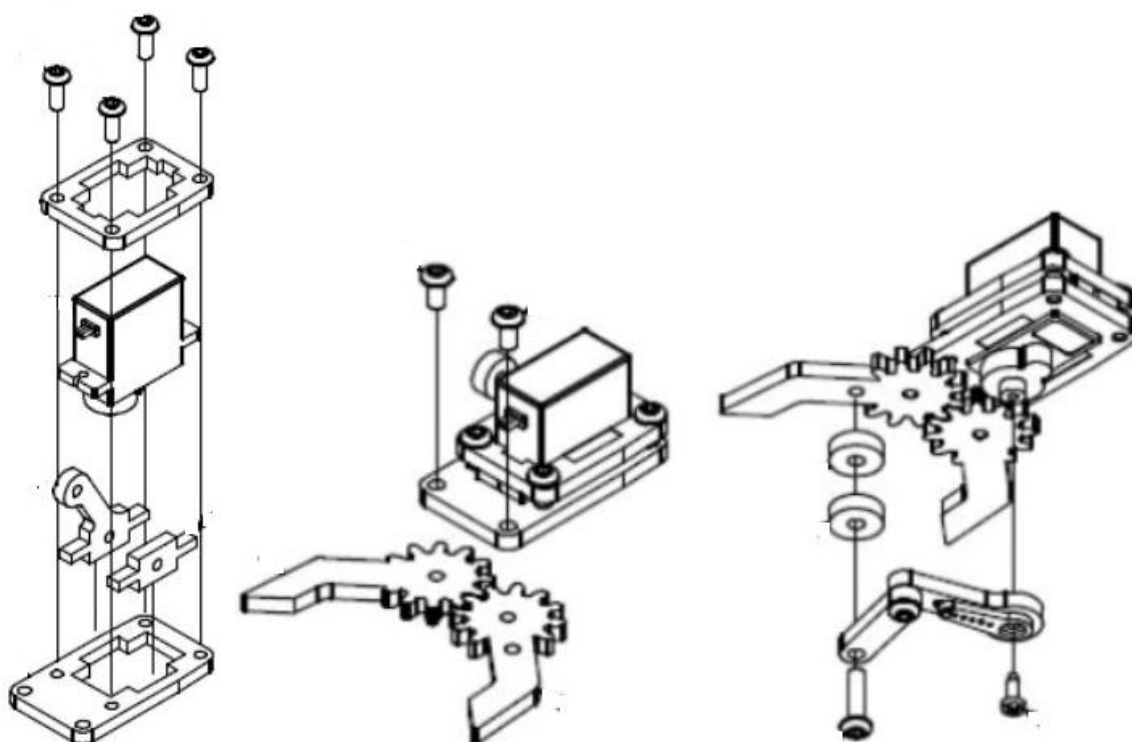


Рис. 2.9. Етапи збору маніпулятора

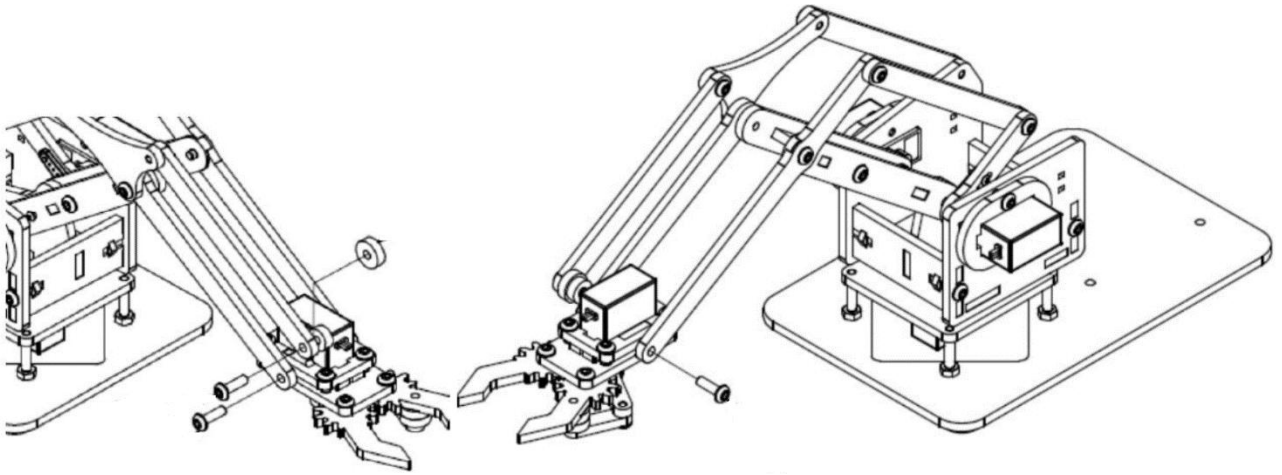


Рис. 2.10. Етапи збору маніпулятора

2.4. Керування приводом

Привод призначений для перетворення енергії, що надходить від джерела живлення, у механічний рух виконавчих ланок робота відповідно до керувальних сигналів системи управління. У загальному випадку привод складається з двигуна та трансмісії, яка включає передавальні механізми (рис. 2.11).

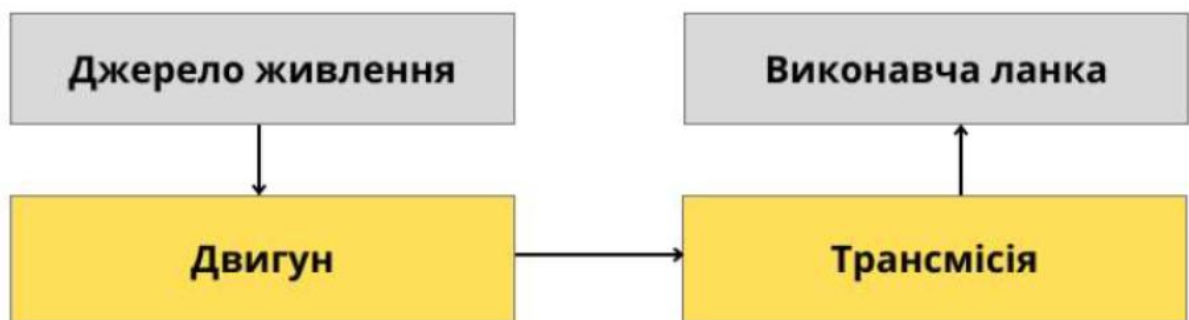


Рис. 2.11. Загальна схема приводу

Тип приводу визначається видом використовуваного енергоносія. У робототехнічних системах застосовуються пневматичні, гідравлічні, електричні та комбіновані приводи. На ранніх етапах розвитку робототехніки домінували пневматичні та гідравлічні приводи, однак у сучасних конструкціях перевага надається електроприводам.

Вибір типу приводу зумовлюється функціональним призначенням робота, технологічними вимогами, умовами експлуатації, особливостями виробничого середовища, а також економічними чинниками, зокрема вартістю реалізації, складністю технічного обслуговування та експлуатаційними витратами.

У маніпуляторах можуть використовуватися всі типи приводів, тоді як у механізмах пересування мобільних роботів переважно застосовується електричний, а значно рідше — комбінований привод.

Для забезпечення рухливості робота було прийнято рішення застосувати комплект сервоприводів, які повністю відповідають вимогам за габаритами та вантажопідйомністю, а також вирізняються відносно невисокою вартістю.

Структура електричного сервоприводу полягає у перетворенні заданого кута повороту ланки маніпулятора φ_n на відповідну електричну напругу, яка подається на сервопідсилювач і далі на сервомотор. Механічний вплив мотора через редуктор повертає ланку на кут φ_1 , близький до заданого. Поточний кут φ_1 контролюється датчиком зворотного зв'язку, який формує сигнал для компаратора. Різниця між заданим та фактичним значенням автоматично коригує роботу мотора, забезпечуючи швидке та точне досягнення необхідного положення ланки маніпулятора.

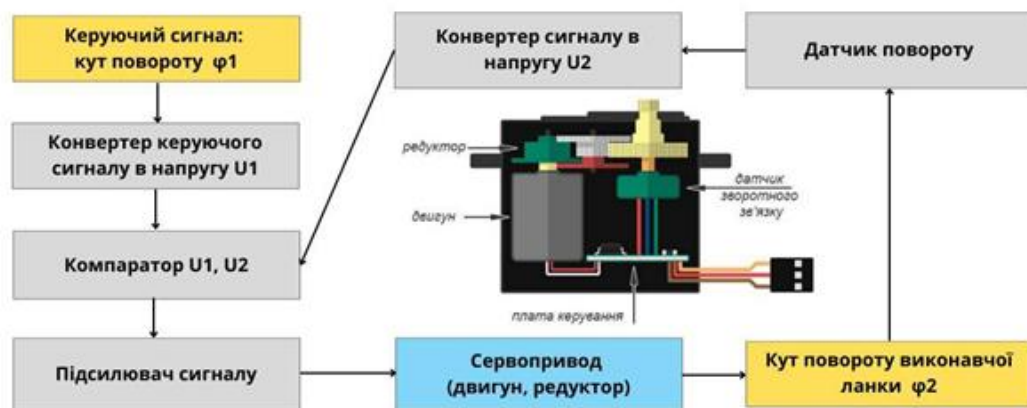


Рис. 2.12. Схема електричної сервосистеми


Вихідний сигнал φ_n подається на сервопривід від керуючого пристрою (комп'ютера), безпосередньо підключеного до контуру управління приводом. У багатьох електричних сервосистемах для підвищення точності використовується подвійний зворотний зв'язок: позиційний сигнал (наприклад, кут повороту) надходить на комп'ютер, а сигнал швидкості обертання вихідної ланки — від тахогенератора на сервопідсилювач. Така система забезпечує плавне наближення виконавчої ланки до заданої позиції і називається системою з

м'яким, або плавним керуванням. Система з одним контуром зворотного зв'язку називається системою жорсткого керування.

Порівняльні характеристики систем розімкненого типу (кроковий двигун) і замкненого типу (сервопривод) подано в таблиці 2.3.

Таблиця 2.3

Порівняльна характеристика крокових двигунів та сервоприводів

Характеристика	Кроковий двигун	Сервопривод
		
Призначення	Позиціонування вихідної ланки в залежності від керуючого сигналу	
Особливості	Розімкнений контур (без зворотного зв'язку)	Замкнений контур за рахунок системи зворотного зв'язку
Точність позиціонування	середня	висока
Переваги	Менша вартість; простота керування; більш висока надійність; постійний крутний момент на низьких швидкостях	Висока точність; ефективне енергоспоживання; постійний крутний момент на різних швидкостях; відносно тиха робота
Недоліки	Генерує більше тепла; відсутність зворотного зв'язку; зниження крутного моменту на високій швидкості; відносно шумна робота	Більш висока ціна; складна конструкція; більш високі витрати на експлуатацію
Типові сфери застосування	3D-принтери; верстати з ЧПК; сканери; цифрові фотоапарати	Робототехніка (маніпулятори, дрони); обладнання з ЧПК; приводи конвеєрів

Порівняно з розімкненими системами на базі крокових двигунів, замкнені сервоприводи забезпечують значно більшу точність позиціонування, ефективніше енергоспоживання та стабільний крутний момент на різних швидкостях, хоча мають складнішу конструкцію та вищу вартість. Крокові двигуни простіші та дешевші, підходять для задач, де не потрібен зворотний зв'язок, але при високих швидкостях їх крутний момент знижується, а робота стає більш шумною. Основна сфера застосування крокових двигунів — 3D-принтери, ЧПК-станки,

сканери, цифрові камери, а сервоприводів — робототехніка (маніпулятори, дрони), ЧПК-обладнання та конвеєрні приводи.

Для роботи з виробами складної геометрії маніпулятор повинен забезпечувати поступовий рух робочого органу по трьох координатах, а також можливість обертання навколо однієї, двох або трьох осей. Крім того, сучасні роботи-маніпулятори здатні одночасно виконувати комбіновані рухи: обертання навколо однієї осі при поступовому переміщенні вздовж інших двох, або два обертальних рухи й одне поступове переміщення у радіальному напрямку.

Робочий орган маніпулятора безпосередньо взаємодіє з об'єктом контролю. У промислових роботах він може бути реалізований як спеціалізований захватний пристрій або комплексний робочий модуль, оснащений датчиками контролю та захоплення [11].

Складність конструкції робота-маніпулятора визначається кількістю кінематичних пар, які він містить. Маніпулятор n -го порядку формується кінцевою множиною кінематичних пар, які можуть бути одного типу або комбінованими різних видів. Важливою характеристикою такої структури є її однорідність, яка безпосередньо впливає на маневреність і гнучкість маніпуляційної системи [12].

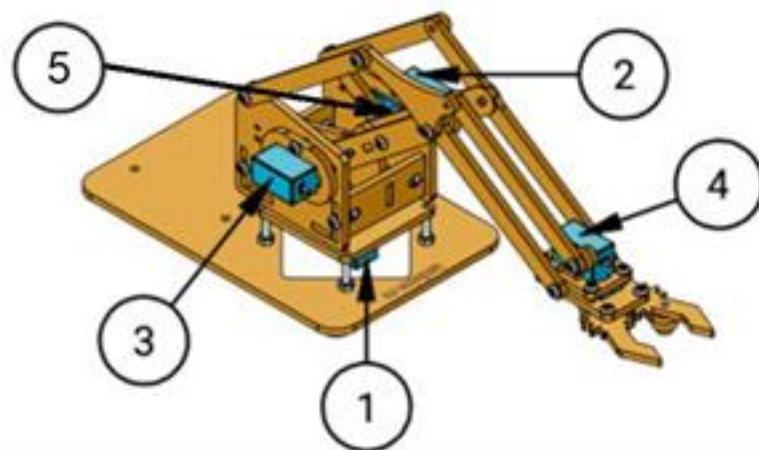


Рис. 2.13. Макетна схема розташування серводвигунів робота-маніпулятора

На рис. 2.13 показано макетну схему розташування серводвигунів, що передбачено для реалізації в даному проєкті. На схемі відзначені серводвигуни у позиціях 1–5, які забезпечують привід відповідних ланок маніпулятора.

Висновки до розділу

У другому розділі проведено теоретичне обґрунтування та практичне проектування механічної складової робота-маніпулятора. Розглянуто типові кінематичні структури маніпуляторів та їх базові системи координат (декартова, циліндрична, сферична, ангулярна, SCARA). Встановлено, що для проектування універсального лабораторного маніпулятора найбільш доцільною є схема з кутовою (ангулярною) системою координат, яка забезпечує компактність конструкції при достатньому робочому просторі. Розроблено математичну модель чотириланкового маніпулятора для подальшого синтезу системи керування. Для опису кінематики використано метод Денавіта–Хартенберга, що дозволило систематизувати параметри ланок (довжини, кути повороту, зміщення). Динаміку системи описано за допомогою рівнянь Лагранжа другого роду, що враховують інерційні навантаження, коріолісові та відцентрові сили, а також гравітаційні моменти. Отримана модель є основою для точного розрахунку керуючих впливів на приводи.

Спроектовано 3D-модель маніпулятора у середовищі САПР Autodesk Inventor на базі прототипу 4DOF Arm Kit. Це дозволило перевірити кінематичні зв'язки, виявити потенційні колізії деталей та підготувати елементи конструкції до виготовлення методом 3D-друку. Використання адитивних технологій забезпечує можливість швидкої модифікації вузлів та здешевлює виробництво дослідного зразка.

Проведено порівняльний аналіз приводів для забезпечення рухливості ланок. Визначено, що для навчально-лабораторних цілей оптимальним є використання електричних сервоприводів із замкненим контуром керування. На відміну від крокових двигунів, сервоприводи забезпечують вищу точність позиціонування завдяки вбудованому датчику зворотного зв'язку та мають кращі масогабаритні показники, що є критичним для легких маніпуляторів.

РОЗДІЛ 3. ПРОЄКТУВАННЯ ТА ПРАКТИЧНА РЕАЛІЗАЦІЯ СИСТЕМИ АВТОМАТИЗОВАНОГО КЕРУВАННЯ РОБОТОМ-МАНІПУЛЯТОРОМ НА БАЗІ ПЛАТФОРМИ ARDUINO

3.1 Структура та принципи роботи системи автоматизованого керування

На початковому етапі проєктування автоматизованої системи керування маніпуляційним роботом із використанням голосового управління на базі платформи Android необхідно окреслити основні функціональні можливості системи відповідно до визначених завдань. З метою впорядкування та наочного подання цих можливостей було розроблено структурну схему, представлену на рисунку 3.1.



Рис. 3.1. Можливості системи управління

До основних функціональних можливостей такої системи належать такі:

- керування маніпуляційним роботом із використанням голосових команд;

- розпізнавання, обробка та інтерпретація голосових команд оператора;
- визначення та контроль поточного положення маніпулятора під час виконання заданих операцій;
- організація бездротового обміну даними між роботом і мобільним пристроєм на платформі Android;
- забезпечення безпечної роботи системи, зокрема автоматичне зупинення руху при виявленні перешкод;
- регулювання швидкості рухів і вибір режимів функціонування маніпулятора;
- реалізація механізмів зворотного зв'язку для інформування користувача про стан системи та результати виконання команд.

Для розв'язання поставленої задачі необхідно розробити систему, здатну реалізувати визначений набір функцій. Початковий етап передбачає формування архітектури системи, добір відповідних апаратних і програмних компонентів, проектування електричної схеми, розроблення програмного забезпечення, а також виконання верифікації й тестування моделі. Після успішного завершення цих етапів здійснюється створення та збирання робочого прототипу.

Загальну структуру автоматизованої системи керування наведено на рисунку 3.2.



Рис. 3.2. Загальна схема системи автоматизованого управління

Архітектура автоматизованої системи керування маніпуляційним роботом ґрунтується на використанні голосового управління за допомогою платформи Android. Маніпулятор містить декілька рухомих ланок, привід яких здійснюється сервоприводами або іншими електродвигунами, керованими мікроконтролером. Мікроконтролер забезпечує обмін даними з мобільним пристроєм на базі Android через Bluetooth-модуль з метою приймання та обробки голосових команд.

Структурну схему автоматизованої системи керування роботом-маніпулятором наведено на рисунку 3.3.

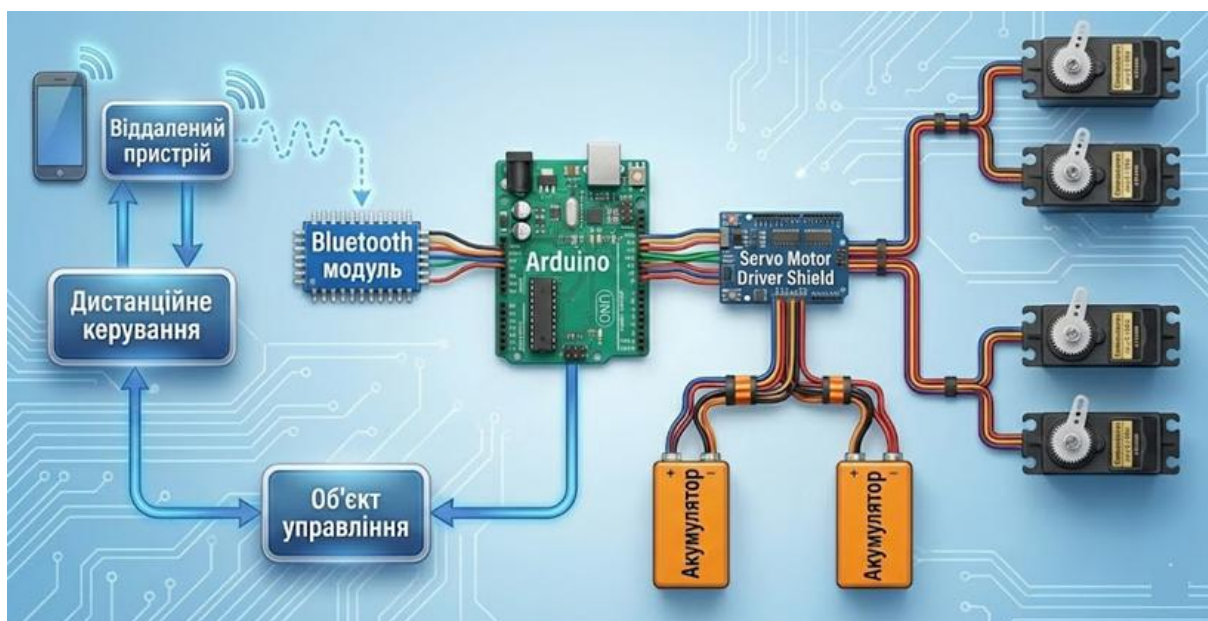


Рис. 3.3. Структурна схема системи управління на основі ARDUINO

Наступним етапом є формування логіки керування системою. Він передбачає виконання низки послідовних дій, зокрема: визначення умов, за яких мають реалізовуватися відповідні операції (наприклад, у разі виявлення об'єкта датчиком ініціюється процес захоплення); застосування логічних конструкцій (if, else, while, for) для побудови алгоритмів керування; виокремлення основних станів системи (наприклад, режим очікування, захоплення об'єкта, його переміщення); а також встановлення умов переходу між різними станами функціонування системи (рис. 3.4).

Розглянемо приклад послідовності виконання дій:



Рис. 3.4. Блок схема алгоритм послідовності дій системи

Ідентифікація ключових компонентів є одним із визначальних етапів проектування, оскільки вона дає змогу окреслити основні модулі системи,

встановити їх функціональне призначення та визначити необхідні зв'язки між окремими елементами.

Процес ідентифікації ключових компонентів передбачає виконання таких послідовних кроків:

Визначення основних модулів системи. До них належать контролери, які виконують роль центрального блоку керування, здійснюють обробку сигналів від датчиків і формують керувальні впливи на виконавчі механізми (наприклад, Arduino Uno R3); датчики, призначені для зчитування інформації про стан системи та параметри зовнішнього середовища (зокрема датчики положення, тиску, температури); виконавчі механізми, які реалізують команди контролера у вигляді фізичних дій (серводвигуни тощо).

Визначення функціональних ролей компонентів. Контролери забезпечують обробку вхідних даних, виконання алгоритмів керування та передачу команд виконавчим пристроям; датчики здійснюють збір і передавання інформації про стан системи та навколишнє середовище; виконавчі механізми безпосередньо виконують керувальні команди, забезпечуючи захоплення, переміщення та позиціонування об'єктів.

Визначення з'єднань і взаємодії між компонентами. Це включає електричні з'єднання для підключення датчиків і виконавчих механізмів до контролера за допомогою проводів і роз'ємів; сигнальні канали для обміну інформацією між контролером і датчиками з метою контролю стану системи; а також канали передавання керувальних команд від контролера до виконавчих механізмів для реалізації необхідних дій.

До складу даної системи входять такі основні компоненти:

1. **Мікроконтролер Keyestudio UNO (ATmega328).** Даний пристрій використовується для приймання команд із мобільного застосунку на платформі Android через Bluetooth-з'єднання, обробки отриманих інструкцій та формування ШІМ-сигналів (PWM), які подаються на відповідні сервоприводи для виконання операцій підйому та позиціонування ланок маніпулятора [21].

2. **Чотири сервомотори SG90.** Ці електромеханічні пристрої виконують функцію приводів суглобів робота та забезпечують по одному ступеню свободи для кожної рухомої ланки. Застосування чотирьох сервомоторів надає маніпулятору чотири ступені свободи, що дозволяє реалізувати необхідні маніпуляційні рухи [21].
3. **Bluetooth-модуль HC-05.** Енергоефективний модуль, призначений для двостороннього послідовного обміну даними через відповідний комунікаційний інтерфейс. Він використовується для організації бездротового зв'язку між мобільним застосунком Android та мікроконтролером ATmega328 [21].
4. **Блок живлення (літій-іонні акумулятори або джерело напруги 5–9 В).** Живлення маніпулятора забезпечується літій-іонними акумуляторами, з'єднаними у послідовно-паралельній конфігурації, що дозволяє отримати необхідні електричні параметри. Мінімальна потужність джерела живлення повинна становити не менше 28 Вт (7 В, 4 А) [21].

Запропонована структура автоматизованої системи керування є ефективним і надійним рішенням для реалізації голосового управління маніпуляційним роботом. Використання платформи Android як засобу введення команд забезпечує зручність експлуатації та розширює функціональні можливості системи за рахунок застосування мобільного пристрою. Розглянута архітектура повністю відповідає поставленим вимогам і забезпечує коректне виконання заданих функцій керування маніпулятором.

Наведені елементи дають змогу сформуванню цілісного уявлення про архітектуру та принципи функціонування системи керування роботом-маніпулятором, що сприяє ефективному плануванню й успішній реалізації проєкту. Далі розглянемо детальніший алгоритм роботи робота-маніпулятора, поданий на рисунку 3.5.

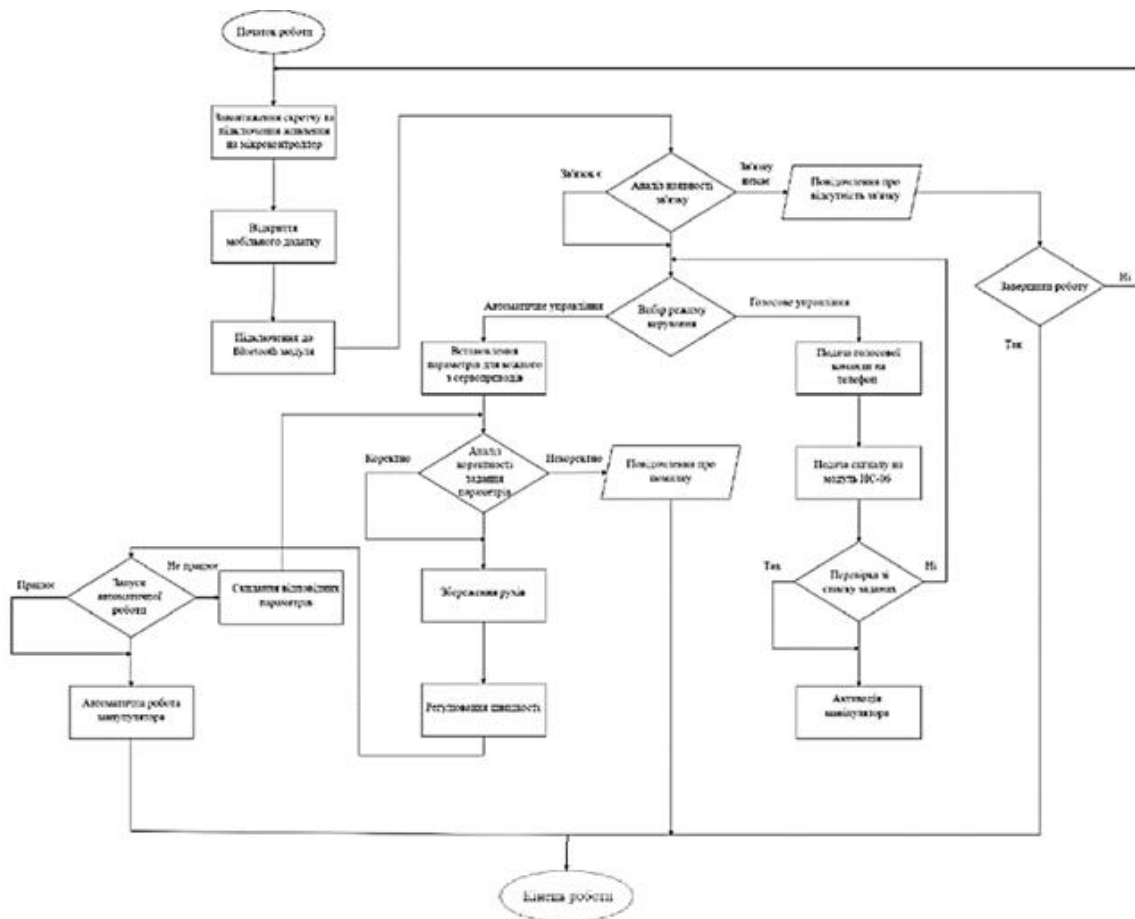


Рис. 3.5. Детальний алгоритм роботи робота-маніпулятора

Процес запуску системи розпочинається із завантаження програмного коду та підключення джерела живлення, яким у даному випадку є літій-іонні батареї. Перед початком роботи в мобільному застосунку задаються необхідні параметри, що відповідають налаштуванням системи. Після цього здійснюється з'єднання з Bluetooth-модулем.

Після цього користувач обирає режим керування, при цьому передбачено два варіанти роботи системи: автоматичний та голосовий. У голосовому режимі керування переміщення окремих ланок маніпулятора здійснюється за допомогою голосових команд, які заздалегідь закладені у програмний код. У разі вибору автоматичного режиму задаються відповідні параметри для кожного сервоприводу. За умови коректного введення координат відбувається збереження траєкторій руху та налаштування швидкості виконання. Якщо ж введені координати є некоректними, система формує повідомлення про помилку та припиняє виконання програми.

Після успішного задання параметрів запускається автоматичний режим роботи, і за відсутності збоїв процес виконується безперервно. У випадку виникнення помилки параметри керування скидаються до початкових значень. За необхідності керування роботом може бути тимчасово призупинене за допомогою кнопки «Пауза».

3.2. Розробка системи керування роботом маніпулятором на основі ARDUINO

Щоб оживити прототип і запрограмувати його рухи, необхідно підібрати найбільш відповідну мікропроцесорну платформу.

Платформа Arduino є дуже популярним рішенням у сфері мікропроцесорної техніки та зарекомендувала себе як надійний і зручний інструмент для реалізації проєктів різного спрямування (рис. 3.6.).

Хоча Arduino виготовлена з якісних компонентів і демонструє стабільну роботу, її вартість значно вища порівняно з аналогами, що може бути економічно складним варіантом для студентів і новачків. Тому для реалізації даного проєкту було обрано платформу Keyestudio Uno у поєднанні з доповненням Keyestudio Servo Motor Driver Shield, яке забезпечує зручне керування сервомоторами.

Платформа Keyestudio Uno включає мікроконтролер та інтерфейсну плату, яка забезпечує зручний доступ до його контактів і дозволяє підключати різноманітні модулі та датчики. Відкрита архітектура цієї платформи дає можливість розробляти власне програмне забезпечення та розширювати функціональність за допомогою численних модулів і бібліотек, підтримуваних активною спільнотою розробників.

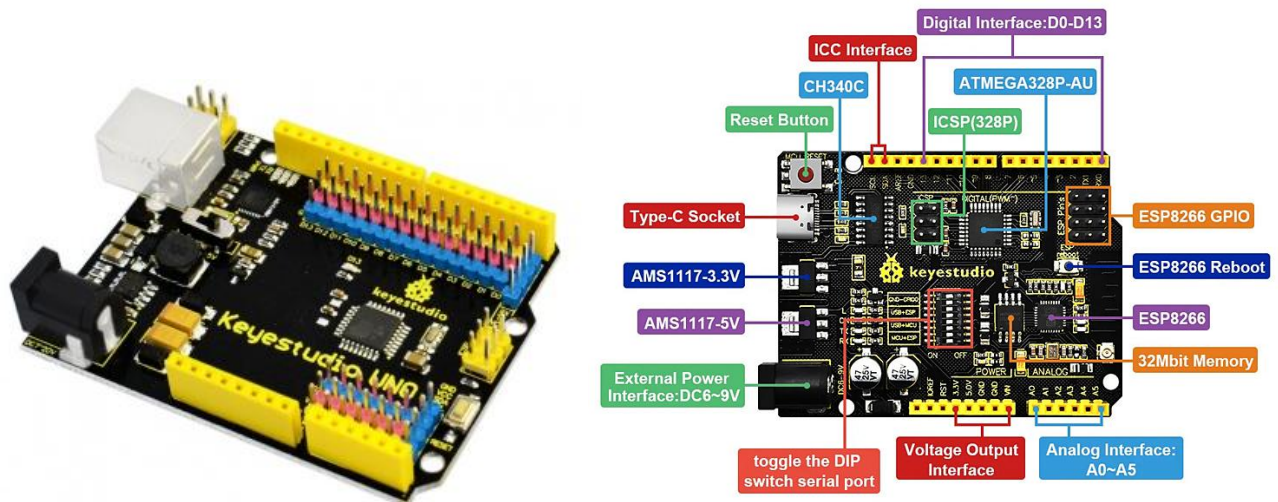


Рис. 3.6. Зовнішній вигляд Keyestudio Uno

На початковому етапі ознайомимося з платою керування Arduino Uno, зокрема з її основними входами та виходами (рис. 3.7).

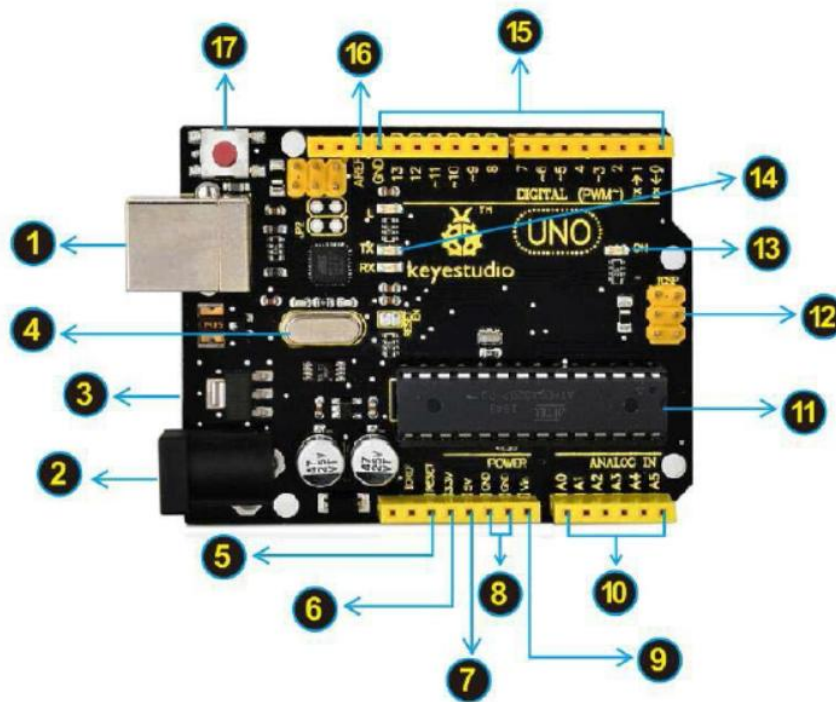


Рис. 3.7. Плата керування ARDUINO UNO з позначеними входами та виходами

На рисунку 3.7 наведено умовні позначення основних елементів плати Arduino Uno:

- 1) USB-підключення – дозволяє жити плату та програмувати її через USB-порт.
- 2) Гніздо живлення постійного струму – плата може отримувати енергію від зовнішнього джерела постійного струму.

3) Регулятор напруги – контролює та стабілізує напругу, що подається на мікроконтролер та інші компоненти плати.

4) Кристалічний генератор – забезпечує точне відлічування часу; на кристалі вказано 16.000H9H, що відповідає частоті 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, мікропроцесор Arduino, що складається з MOSI, MISO, SCK, RESET, VCC і GND.

13) Світлодіодний індикатор живлення. Під час живлення Arduino світлодіод горить, що 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.

Технічні характеристики Keyestudio Uno R3

Параметр	Значення
Мікроконтролер	ATmega328P-PU (у DIP корпусі або SMD, залежно від версії)
Робоча напруга	5 В
Вхідна напруга (рекомендована)	7-12 В (через роз'єм живлення DC)
Вхідна напруга (гранична)	6-20 В
Цифрові входи/виходи	14 (з них 6 підтримують ШІМ/PWM)
Аналогові входи	6
Постійний струм на пін I/O	20 мА (максимум 40 мА)
Постійний струм для піна 3.3V	50 мА
Тактова частота	16 МГц
USB-інтерфейс	Зазвичай ATmega16U2 (як в оригіналі) або CP2102 (у деяких ревізіях для стабільності)

Принципова електрична схема плати наведена на рисунку 3.8.

Bluetooth-інтерфейс: використовується для бездротового зв'язку з іншими пристроями, наприклад, смартфонами або комп'ютерами, забезпечуючи передачу команд і отримання даних.

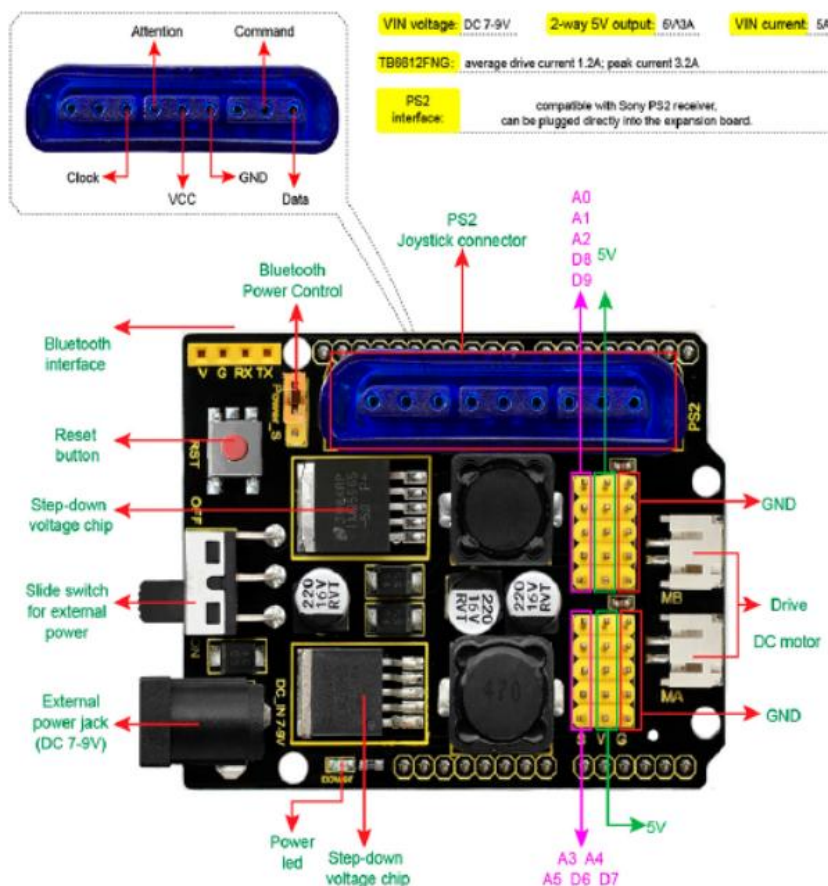


Рис. 3.9. Плата розширення двигуна/сервоприводу keyestudio TB6612FNG

Роз'єм для підключення двигунів MA та MB: виходи для керування двигунами постійного струму, що забезпечують подачу живлення та сигналів управління для двигуна.

Штекер для зовнішнього живлення: дозволяє підключати зовнішнє джерело живлення в діапазоні 7–9 В для стабільної роботи плати та підключених пристроїв.

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

Слайдер (перемикач живлення): вмикає та вимикає живлення плати, може використовуватися для тестування системи.

Кнопка "Reset": призначена для перезапуску плати, корисна при налагодженні та відладці системи.

Індикатори живлення: показують наявність електроживлення та стан роботи плати.

Аналогові та цифрові контакти (A0–A5, D8–D7):

- **Аналогові входи (A0–A5):** використовуються для підключення сенсорів та пристроїв, що передають аналогові сигнали (наприклад, датчики температури).
- **Цифрові входи/виходи (D8–D7):** призначені для обробки цифрових сигналів, підключення кнопок, світлодіодів та інших цифрових компонентів.

Платформа Keyestudio підтримує широкий спектр вбудованих модулів і компонентів, включно з датчиками, моторами, актуаторами та іншими пристроями, що робить процес розробки роботів більш гнучким та оперативним. Завдяки відкритій архітектурі та наявності численних бібліотек, створення робототехнічних проектів на базі Keyestudio Uno потребує менше часу порівняно з іншими платформами [18].

Серед основних переваг використання Keyestudio у робототехніці варто виділити простий інтерфейс і зручність програмування на C/C++, що робить платформу доступною для новачків і дозволяє швидко освоювати основи роботи з мікроконтролерами. Крім того, активна спільнота користувачів надає великий обсяг навчальних матеріалів, включно з інструкціями, відеоуроками та прикладами проектів, що значно спрощує процес навчання.

Підтримка концепції «шільдів» (shields) у Keyestudio Uno дозволяє легко підключати додаткові плати для розширення функціоналу, забезпечуючи модульність та адаптивність платформи до різних проектів у сфері робототехніки та автоматизації.

Завдяки простоті програмування та можливості використання готових моделей, розробка роботів на базі Keyestudio зазвичай займає менше часу та

ресурсів. Важливою перевагою є також доступність платформи, що дозволяє економити на обладнанні, особливо для студентів та початківців.

Для забезпечення рухомості робота було обрано комплект сервоприводів SG-90 (рис. 3.10), які відповідають вимогам за габаритами та вантажопідйомністю, а також відрізняються доступною вартістю.



Рис. 3.10. Мікро сервопривід Tower Pro SG90

Сервоприводи є електромеханічними пристроями, що поєднують у собі датчик для вимірювання заданих параметрів і блок керування, який автоматично підтримує ці параметри в установлених межах. Перетворення електричної енергії на механічний рух здійснюється за допомогою електродвигуна.

Приводний механізм сервопривода складається з електродвигуна та редуктора. Редуктор призначений для зменшення швидкості обертання двигуна, оскільки його початкова частота обертів зазвичай є надто високою для практичного застосування. Він містить корпус із системою валів і шестерень, які забезпечують передачу та перетворення крутного моменту.

Окрім електродвигуна, редуктора та потенціометра, сервопривід оснащений електронною платою керування. Вона обробляє сигнал, що надходить від потенціометра, порівнює його із заданим значенням і на основі результатів цього порівняння регулює роботу електродвигуна, здійснюючи його увімкнення або вимкнення.

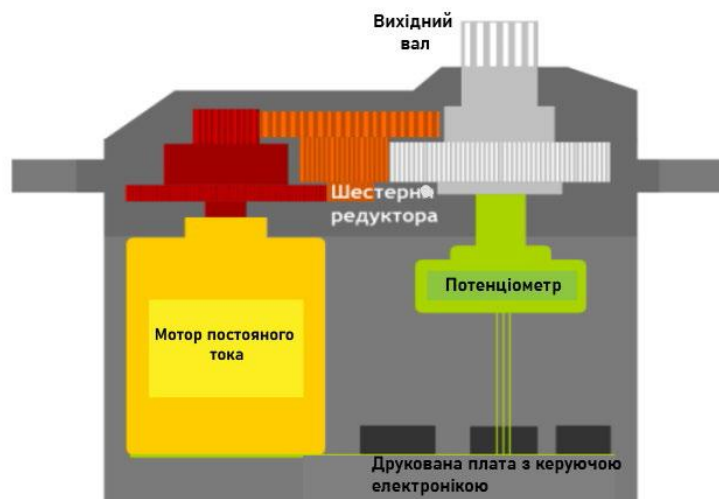


Рис. 3.11. Будова сервоприводу SG-90

Дана модель сервоприводу є якісним і компактним мікросервоприводом, що відрізняється доступною вартістю та малою масою. Вага редуктора може бути віднесена до ультралегкої, оскільки становить лише 9 г.

Таблиця 3.2

Технічні характеристики мікро сервоприводу Tower Pro SG90

Характеристика	Показник
Вага, г	14,7
Частота, мкс	1520
Тип двигуна	Колекторний
Габаритні розміри, мм	23 x 12 x 28,5
Матеріал корпусу та редуктора	Пластик
Швидкість, сек/60°	0,1 – 0,12
Живлення, В	4,8 – 6
Зусилля, кг/см	1,2 – 2,5

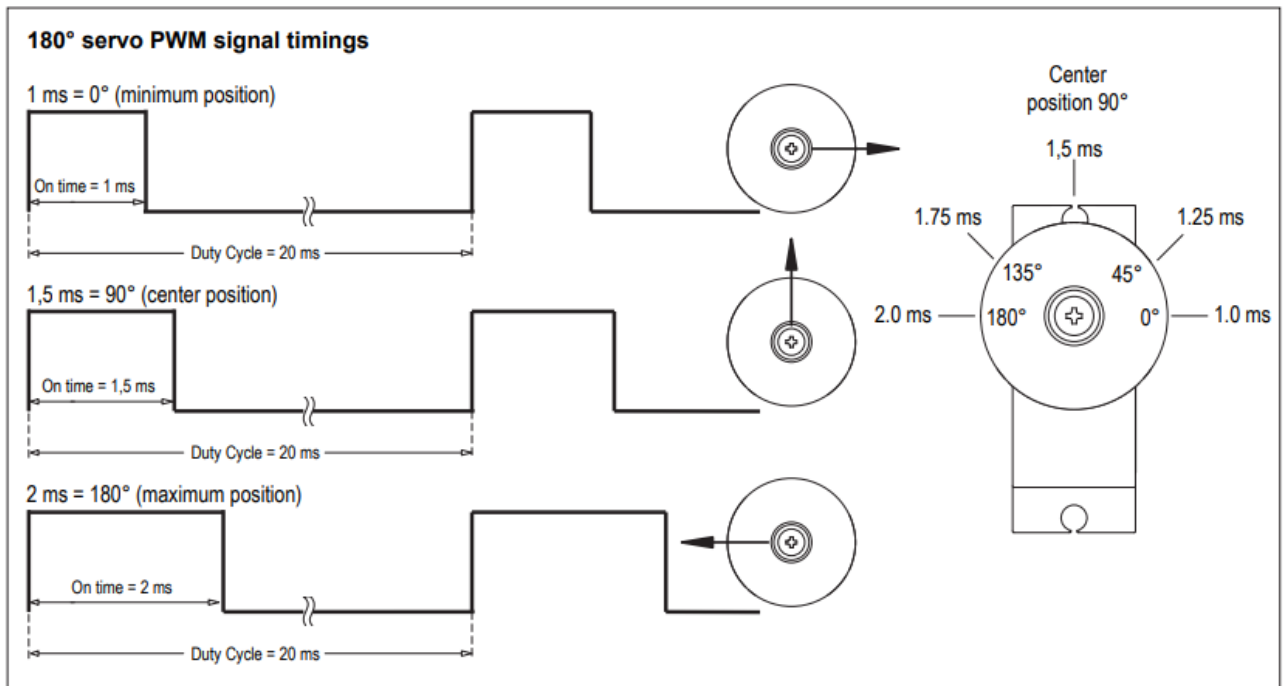


Рис. 3.12. Схема сервоприводу Tower Pro SG90

Для оцінювання складності структури маніпулятора необхідно виконати її розрахунок за відповідним співвідношенням:

$${}^{(P)}\frac{1}{3}\left(\sum_{i=1}^P \rho_i + \sum_{i=1}^P n_i + \omega\right)$$

Де $\sum_{i=1}^P \rho_i$ - кількість кінематичних пар у структурі ланцюга, $\sum_{i=1}^P n_i$ - кількість рухомих частин, ω - ступені свободи в кінематичному ланцюгу.

Структура маніпулятора характеризується кількістю кінематичних пар, що входять до його складу. Маніпулятор n-го порядку включає визначену, обмежену кількість таких пар. Кінематичний ланцюг робота-маніпулятора може бути сформований як з пар одного типу, так і з комбінації різних типів кінематичних пар. Важливим є визначення показника, що відображає однорідність структури, оскільки саме він суттєво впливає на маневреність маніпуляційної системи. Для однорідного кінематичного ланцюга маневреність залишається сталою, тоді як у випадку неоднорідної структури вона може змінюватися залежно від взаємного розташування кінематичних пар.

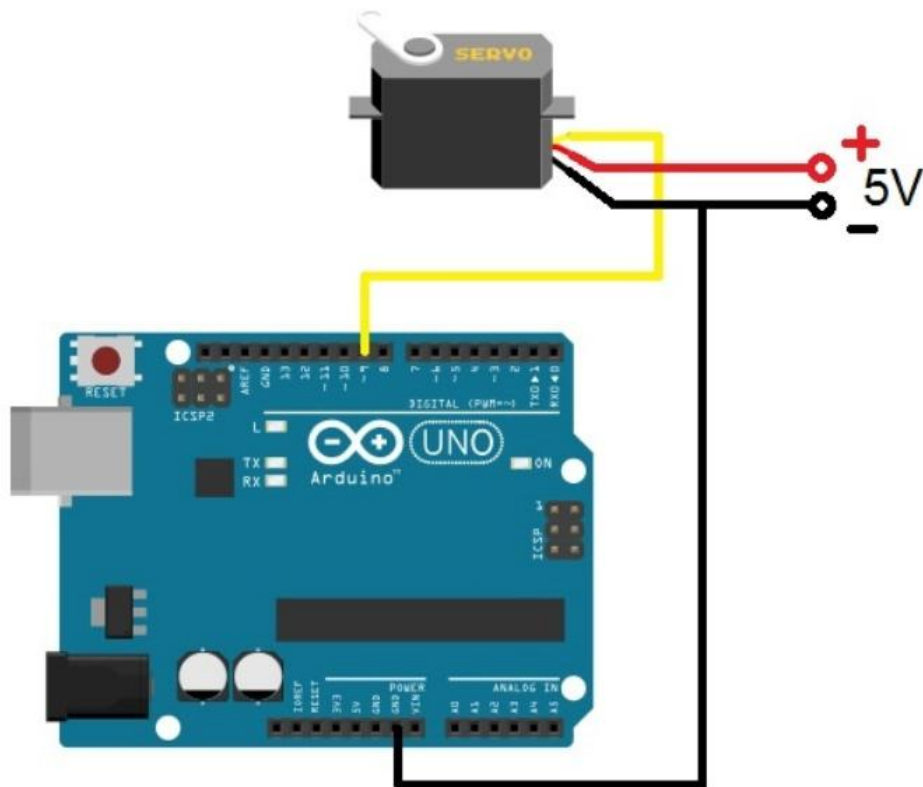


Рис. 3.13. Схема підключення сервоприводу Tower Pro SG90 до Arduino Uno R3



Рис. 3.14. Зовнішній вигляд Keyestudio Servo Motor Driver Shield

Технологія Bluetooth є одним із найпоширеніших стандартів бездротового обміну даними між електронними пристроями на невеликій відстані. Розроблена консорціумом Bluetooth Special Interest Group, ця технологія знайшла застосування в різноманітних пристроях і системах — від бездротових навушників і акустичних систем до медичного обладнання та автомобільних інформаційно-розважальних систем. У поєднанні з платформами, такими як

Keyestudio Uno та **Keyestudio Servo Motor Driver Shield**, Bluetooth відкриває нові можливості для автоматизованих проєктів і дистанційного керування пристроями.

Основні характеристики Bluetooth:

Bluetooth працює на коротких відстанях, забезпечуючи надійний зв'язок між пристроями на відстані до 100 метрів. Однією з ключових переваг цієї технології є низьке енергоспоживання при достатній швидкості передачі даних, що робить її оптимальним вибором для енергоефективних застосунків, таких як датчики, фітнес-трекери та системи автоматизації на базі Keyestudio Uno.

Взаємодія мікроконтролерів Keyestudio з Bluetooth:

Використання модулів Bluetooth разом з платформою **Keyestudio V4.0 Control Board** дозволяє легко організувати бездротовий зв'язок між мікроконтролером та іншими пристроями, такими як смартфони або комп'ютери. Це особливо корисно для проєктів із дистанційним керуванням або моніторингом системи.

HC-05 — це універсальний Bluetooth-модуль, що забезпечує двосторонній бездротовий зв'язок у ваших проєктах [26]. Він ідеально підходить для комунікації між мікроконтролерами, такими як Arduino, або для зв'язку з будь-яким пристроєм, що підтримує Bluetooth, наприклад, смартфоном чи ноутбуком. Існує безліч доступних Android-додатків, що значно спрощують цей процес. Важливо зазначити, що HC-05 не призначений для передачі мультимедійних даних, як-от фотографій чи музики, але для потреб нашого пристрою це й не є необхідним.

Технологія Bluetooth дозволяє передавати дані між двома пристроями, що знаходяться в безпосередній близькості, без вимоги прямої видимості. Вона забезпечує високу стійкість до широкутругових перешкод, що дозволяє багатьом пристроям одночасно взаємодіяти в одному просторі, не заважаючи один одному. Ця технологія широко використовується в телефонах, планшетах та ноутбуках [27].

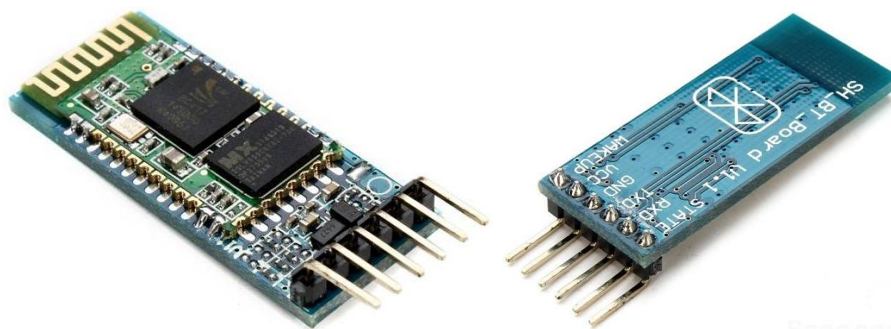


Рис. 3.15. Bluetooth модуль HC-05

Таблиця 3.3

Технічні характеристики Bluetooth-модуля HC-05 для Arduino

Показник	Значення
Діапазон частот радіозв'язку	2,4–2,48 ГГц
Потужність передавача	0,25–2,5 мВт
Чутливість	-80 dBm
Напруга живлення	3,3–5 В
Споживаний струм	50 мА
Радіус дії	до 10 метрів
Інтерфейс	послідовний порт
Режими	master, slave
Робочий діапазон температур	-25...+75 °C
Габарити	27 × 13 × 2,2 мм

Принципова схема плати показана на рисунку 3.16.

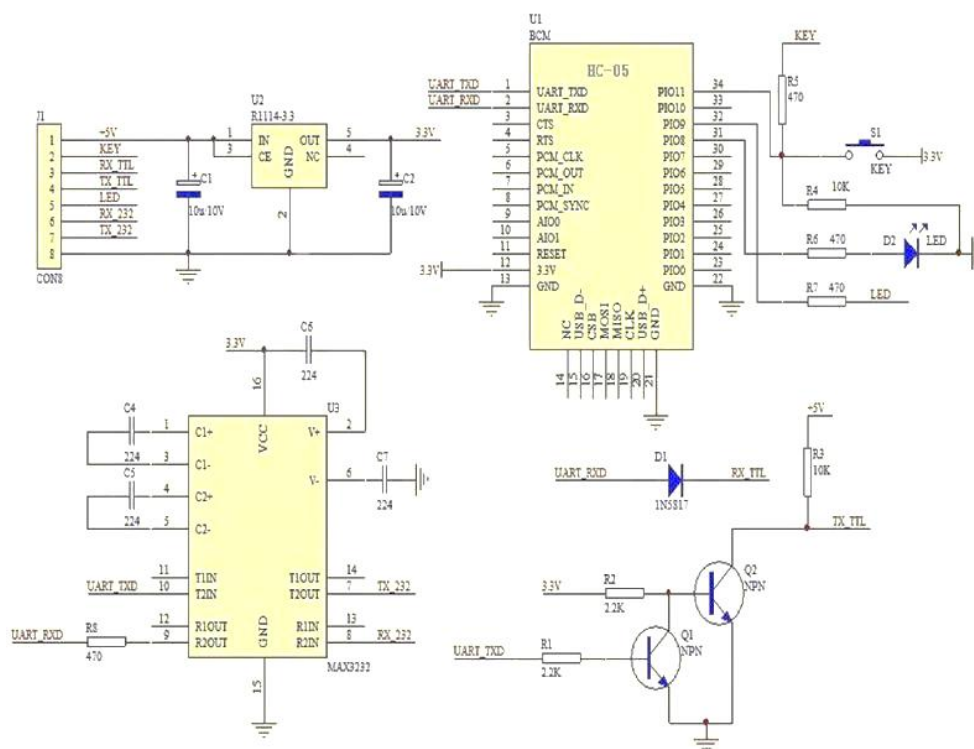


Рис. 3.16. Принципова схема Bluetooth модуля HC-05

Bluetooth-модуль забезпечує бездротове з'єднання між Arduino-модулями та сумісними пристроями (наприклад, КПК, смартфонами), функціонуючи в режимі "master". Взаємодія з контролером Arduino реалізується за допомогою послідовного інтерфейсу RS-232.

Наприклад, модуль **Bluetooth HC-05** (рис. 3.17) можна інтегрувати з Keyestudio Uno, що дозволяє бездротово керувати підключеними сервоприводами через Keyestudio Servo Motor Driver Shield.

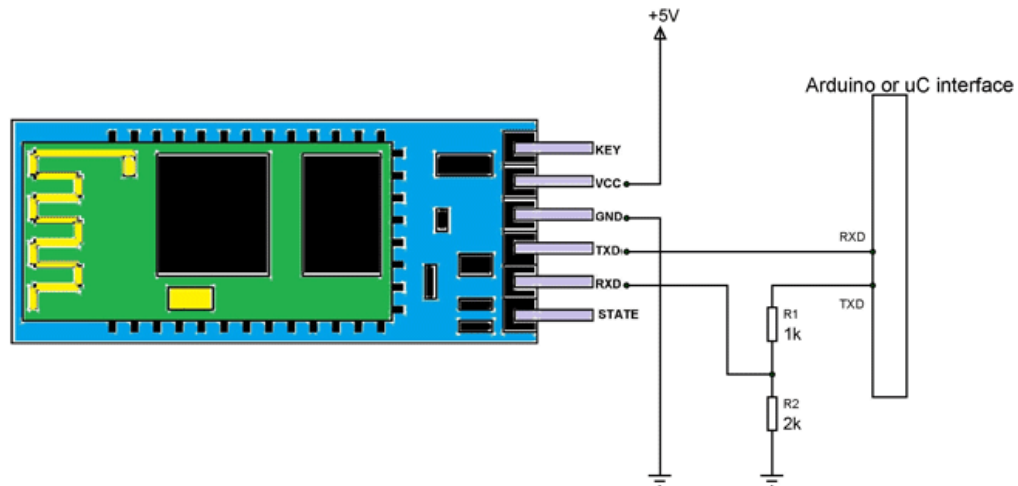


Рис. 3.17. Взаємодія з контролером Arduino

Опис пінів Bluetooth модуля HC-05 показаний на рисунку 3.18.

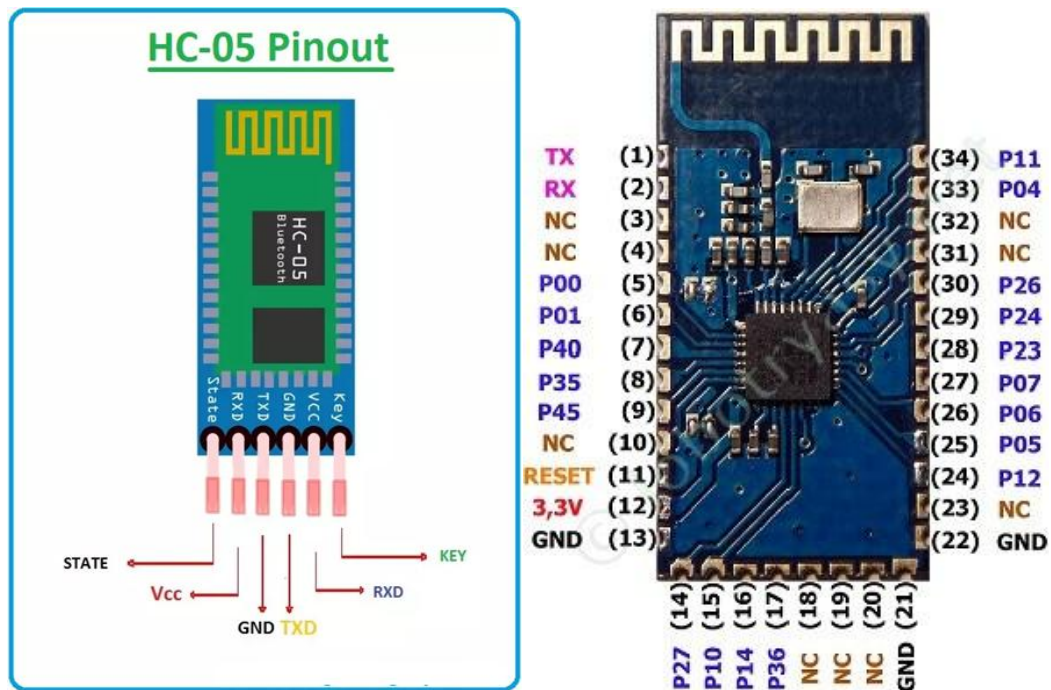


Рис. 3.18. Опис пінів Bluetooth модуля HC-05

Схему підключення Bluetooth модуля HC-05 до Arduino Uno R3 наведено на рисунку 3.19.

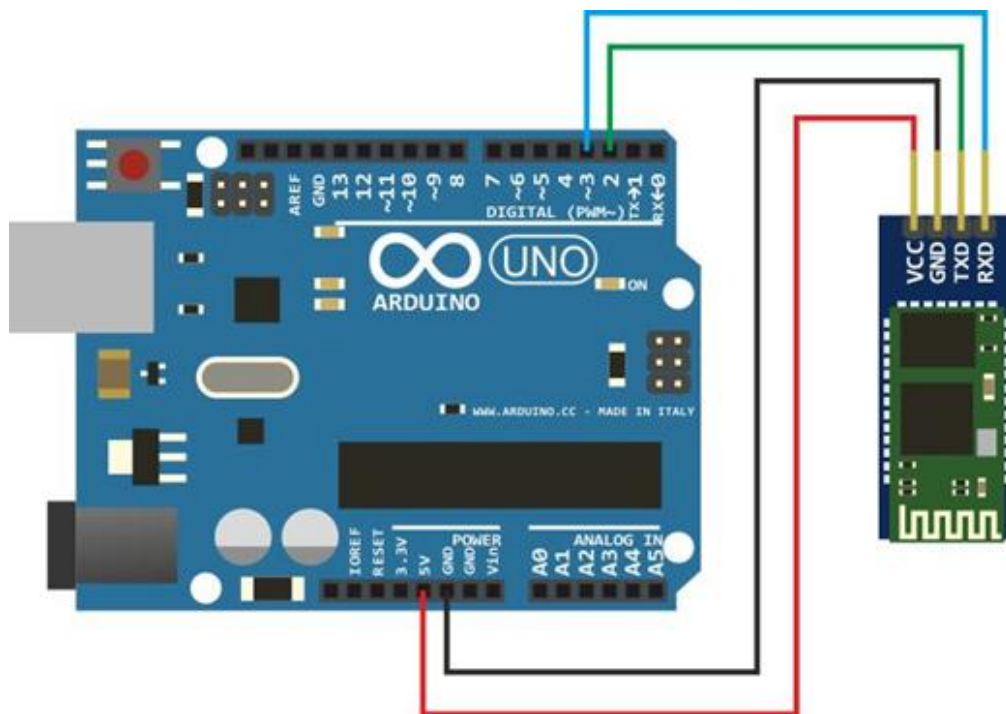


Рис. 3.19. Схема підключення Bluetooth модуля HC-05 до Arduino Uno R3

HC-05 може функціонувати як у режимі "master" (ініціатор з'єднання), так і "slave" (приймач з'єднання), тоді як HC-06 підтримує лише режим "slave".

Оскільки для передачі даних з пристрою на телефон необхідна можливість ініціювати з'єднання, було обрано саме HC-05.

3.3. Розробка електричної схеми системи керування маніпулятором

До складу системи входять такі основні компоненти:

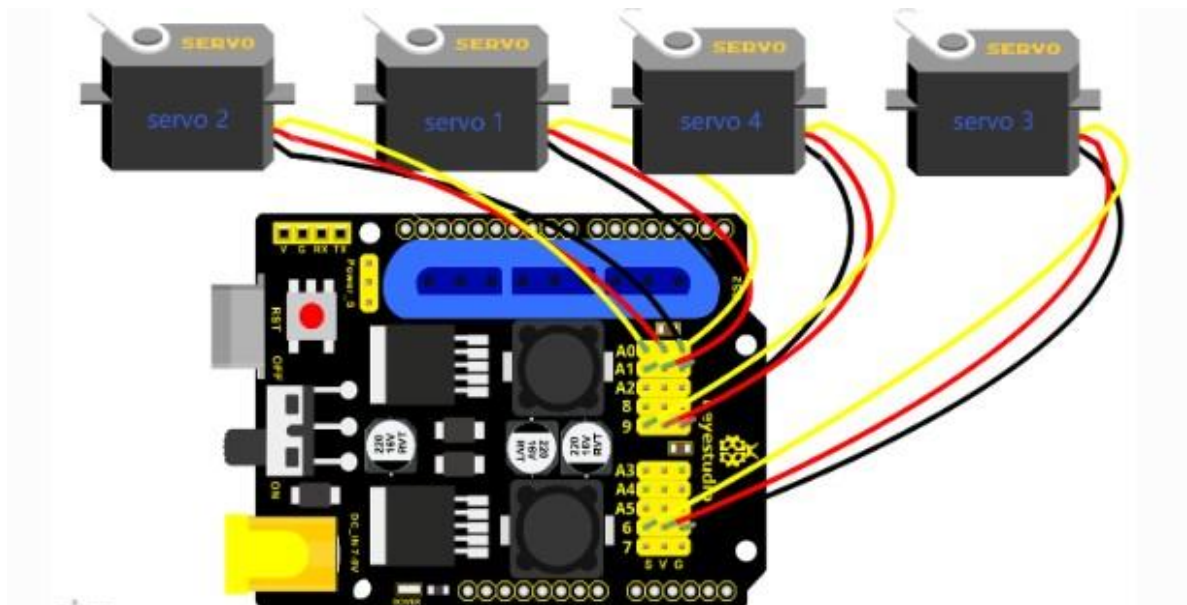
1. **Мікроконтролер Keystudio UNO (ATmega328).** Пристрій призначений для приймання команд із мобільного застосунку на платформі Android через Bluetooth-з'єднання, обробки отриманих інструкцій та формування ШІМ-сигналів (PWM), які подаються на відповідні сервоприводи для виконання операцій підйому й позиціонування маніпулятора [21].
2. **Чотири сервомотори SG90.** Дані приводи виконують роль рухомих з'єднань між ланками робота та є електромеханічними пристроями, кожен з яких забезпечує один ступінь свободи (DOF). Використання чотирьох сервомоторів надає маніпулятору чотири ступені свободи [21].

3. **Bluetooth-модуль HC-05.** Енергоефективний модуль, що забезпечує двосторонній послідовний обмін даними через відповідний інтерфейс. Він використовується для організації зв'язку між мобільним Android-додатком і мікроконтролером ATmega328 [21].

4. **Блок живлення (літій-іонні акумулятори або джерело напруги 5–9 В).** Живлення маніпулятора забезпечується літій-іонними батареями, з'єднаними у послідовно-паралельній конфігурації, що дозволяє отримати необхідні електричні параметри. Мінімальна потужність джерела живлення повинна становити не менше 28 Вт (7 В, 4 А) [21].

Запропонована структура системи є ефективним і надійним рішенням для реалізації голосового керування маніпуляційним роботом. Застосування платформи Android як засобу введення команд підвищує зручність користування та розширює функціональні можливості керування за рахунок використання мобільного пристрою. Розглянута автоматизована система повністю виконує покладені на неї функції та успішно справляється з поставленими завданнями.

Перед початком проєктування електричної схеми було створено попередню схему підключення сервоприводів до мікроконтролера Arduino (рис. 3.20), а також схему підключення Bluetooth-модуля (рис. 3.21) з метою подальшого використання для інтеграції з пристроями на базі Android.



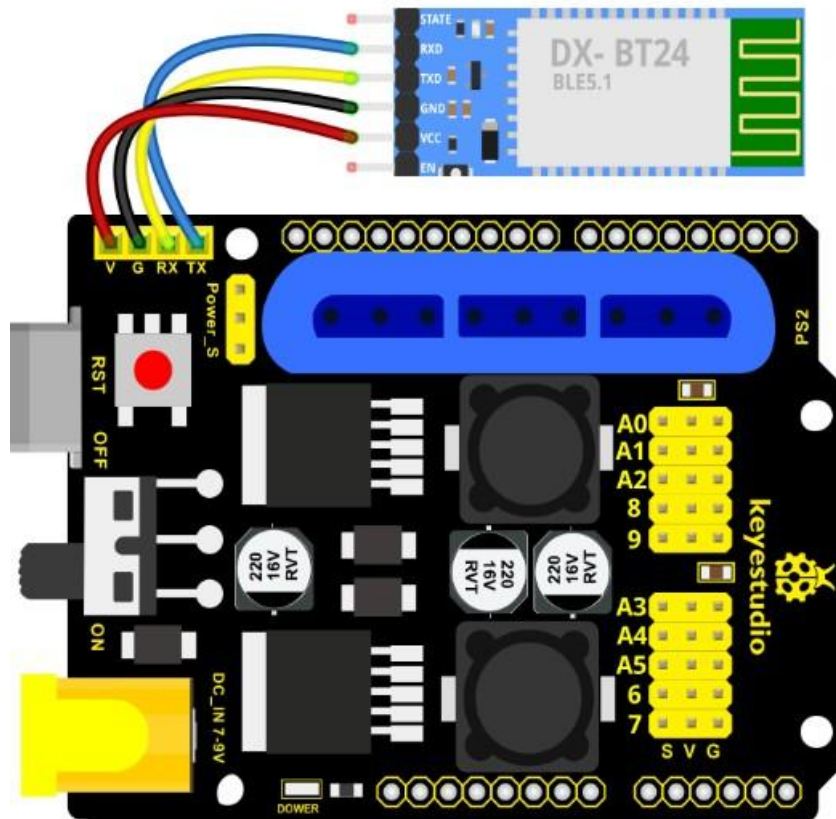


Рис. 3.21. Схема підключення мікроконтролера до Bluetooth-модуля

Для початку моделювання електричної схеми в середовищі Proteus необхідно врахувати, що за замовчуванням плати та модулі Arduino, зокрема пристрої на базі Bluetooth, не входять до стандартної бібліотеки програми. Тому перед початком роботи слід встановити всі необхідні компоненти, які забезпечать коректну симуляцію в середовищі Proteus.

Додавання потрібних елементів, таких як плата Arduino Nano або Bluetooth-модуль, здійснюється шляхом завантаження їх із сторонніх ресурсів, що спеціалізуються на моделюванні електронних пристроїв у Proteus. Для цього використовується механізм підключення додаткових бібліотек через меню Libraries, після чого обирається опція Import Parts. За допомогою графічного інтерфейсу завантажується архів із відповідним компонентом, отриманим з одного з таких ресурсів. Після завершення встановлення додані елементи стають доступними у стандартній бібліотеці Proteus.

Слід зазначити, що застосування сторонніх компонентів може спричиняти певні ускладнення, зокрема некоректну роботу окремих елементів або всієї моделі загалом, а в окремих випадках — повну непрацездатність схеми.

Після успішного встановлення всіх необхідних компонентів можна переходити до складання електричної схеми. При цьому особливу увагу необхідно приділити правильності підключення елементів до плати Arduino. Детальні відомості щодо схем підключення наведені в офіційній документації Arduino, а також на профільних сторонніх ресурсах. Після збору всієї необхідної інформації можна повернутися до середовища Proteus та продовжити моделювання системи.

На рисунку 3.22 наведено електричну схему системи керування.

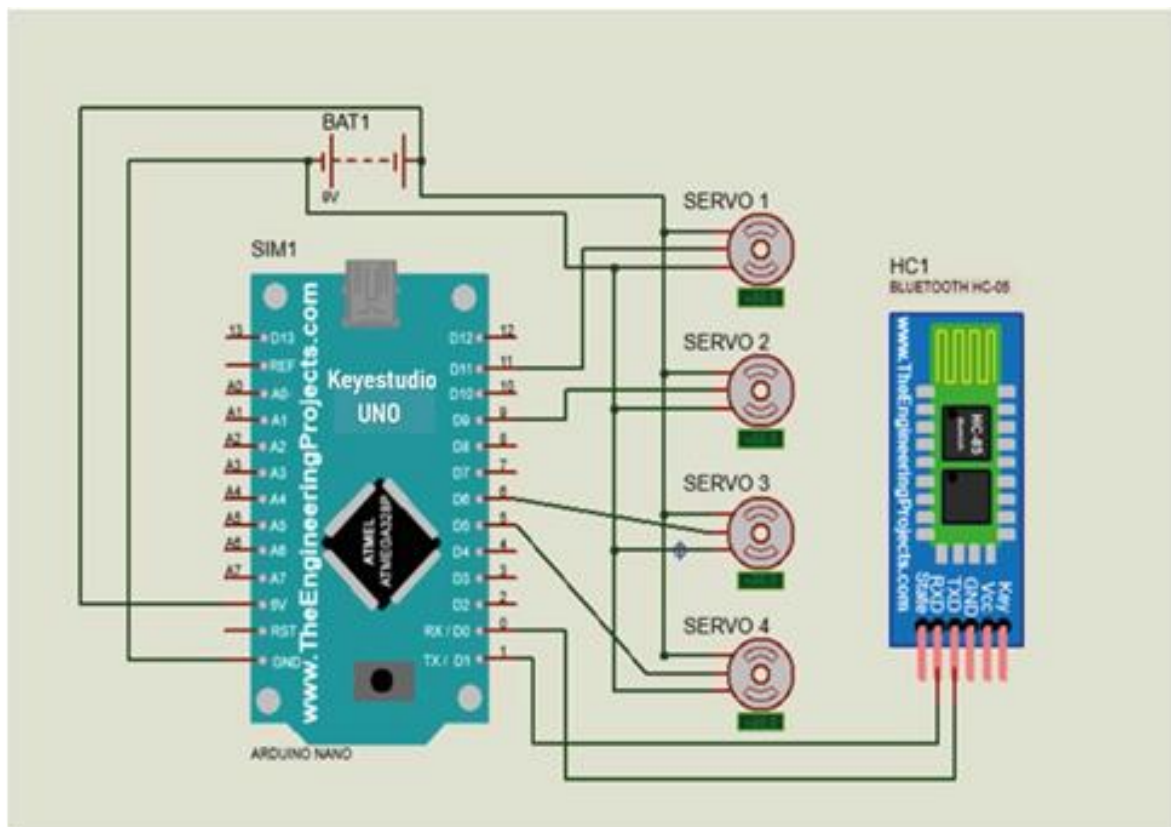


Рис.3.22. Електрична схема пристрою в середовищі Proteus

Для забезпечення працездатності розробленої схеми виконано підключення плати Keyestudio UNO до джерела живлення. Позитивний полюс батареї напругою 6 В під'єднано до виводу 5V плати Keyestudio UNO, а негативний полюс — до виводу GND.

Під час підключення Bluetooth-модуля HC-05 контакт TX з'єднано з контактом RX (A0) плати Keyestudio UNO, а контакт RX — з контактом TX (A1) відповідно.

Наступним етапом є підключення сервоприводів. Ліктьовий сервопривод з'єднано таким чином: контакт VCC підключено до позитивного полюса батареї напругою 6 В, контакт GND — до негативного полюса, а сигнальний провід — до виводу 5 плати Keyestudio UNO. Плечовий сервопривод під'єднано аналогічно: VCC — до позитивного виходу батареї (6 В), GND — до негативного виходу батареї, сигнальний провід — до піну 6 Keyestudio UNO.

Сервопривод основи маніпулятора підключається за такою ж схемою: VCC — до позитивного полюса батареї (6 В), GND — до негативного полюса, сигнальний провід — до піну 9 плати Keyestudio UNO. Для сервоприводу, що виконує функцію захоплення, контакт VCC під'єднано до позитивного виходу батареї (6 В), GND — до негативного виходу, а сигнальний провід — до піну 11 Keyestudio UNO.

Принципову електричну схему системи керування наведено на рисунку 3.23.

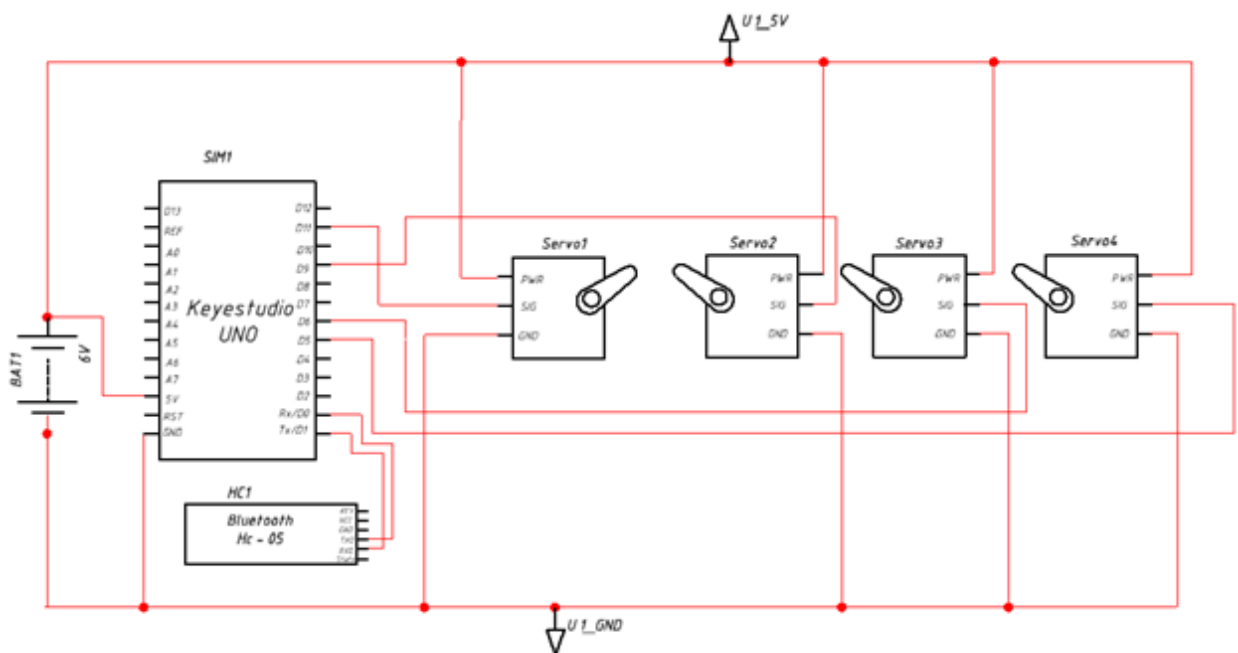


Рис. 3.23. Принципова електрична схема пристрою

3.4. Розробка додатку системи управління для смартфона на базі Android

Після завершення складання маніпулятора та перевірки його працездатності розпочинається етап проєктування системи керування, що передбачає реалізацію голосового управління. Для передавання команд на

платформу Arduino необхідно розробити спеціальний пристрій, до складу якого входять мікрофон для приймання та розпізнавання голосових сигналів, блок пам'яті для зберігання програм, процесор для перетворення звукових хвиль в електричні сигнали, а також Bluetooth-модуль, який забезпечує бездротову передачу команд до Arduino [22].

Мобільний телефон є найбільш доцільною основою для майбутньої розробки, оскільки він володіє всіма необхідними характеристиками, переліченими вище. Суттєвими перевагами створення програмного забезпечення для смартфонів є те, що сучасні користувачі фактично не уявляють свого повсякденного життя без персонального гаджета, використовуючи його як у робочий час, так і під час відпочинку [23]. Оскільки виконавець проєкту користується смартфоном на базі операційної системи Android, було прийнято рішення розробити додаток для керування маніпулятором із застосуванням голосового режиму саме на цій платформі. Результати розробки Android-додатку подано в Додатку Б.

Для реалізації програмного продукту було обрано середовище **MIT App Inventor**. Дане середовище забезпечує простий та наочний інтерфейс для створення мобільних застосунків, що дає змогу розробляти повнофункціональні програми навіть без поглиблених знань у сфері програмування. Воно є доцільним вибором для побудови системи керування, оскільки надає зручні засоби для проєктування користувацького інтерфейсу, а також підтримує інтеграцію з різноманітними сенсорами та API операційної системи Android.

Особливістю цього середовища є можливість створення графічного інтерфейсу для ручного керування, а також реалізація голосового управління з використанням **Google Voice Recognition API** (рис. 3.24).

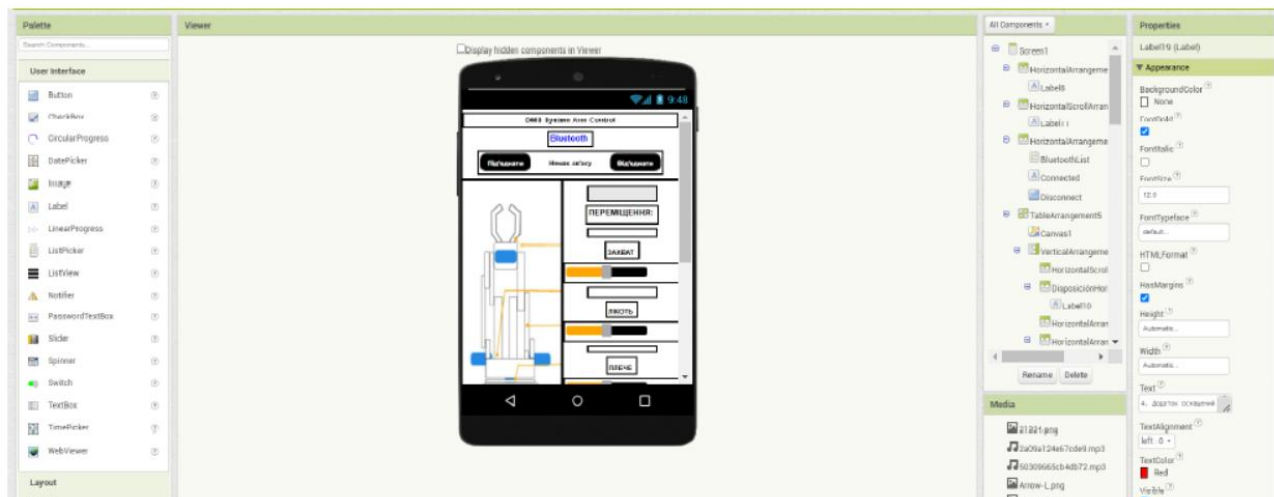


Рис. 3.24. Інтерфейс MIT App Inventor

Інтерфейс застосунку формується шляхом інтуїтивного перетягування необхідних компонентів на віртуальний екран смартфона. Серед доступних елементів можна виокремити:

- кнопки, текстові поля та списки;
- сенсорні модулі (зокрема акселерометр);
- мультимедійні компоненти (вбудоване аудіо та відео);
- інтерактивні елементи керування, такі як перемикачі, повзунки та прапорці (чекбокси).

Мова програмування в цьому середовищі реалізована у вигляді блокової системи, що належить до візуального програмування. Логіка роботи застосунку формується шляхом поєднання графічних блоків, які відображають основні програмні конструкції, зокрема умови, цикли та виклики функцій. Розробка не потребує написання коду традиційними мовами програмування, такими як Java чи Kotlin, натомість використовуються готові блоки, синтаксично подібні до C++, кожен з яких відповідає за виконання конкретної операції або функції. Такий підхід значно спрощує процес програмування, роблячи його наочним і доступним широкому колу користувачів, у тому числі без попереднього досвіду у програмуванні (рис. 3.25).

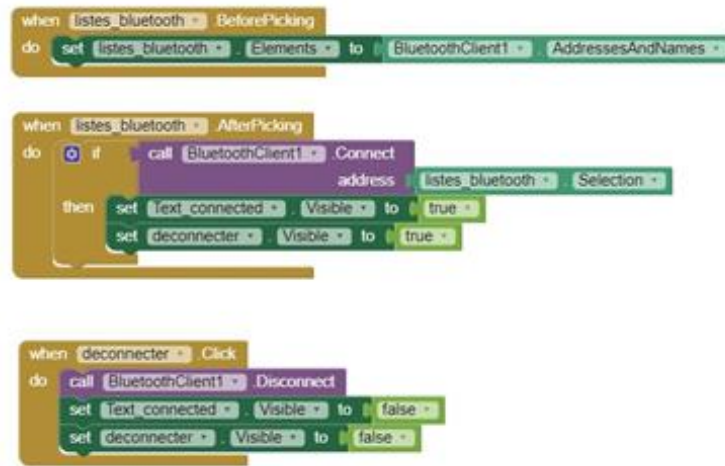


Рис. 3.25. Приклад коду

Програма ґрунтується на використанні графічних блоків, подібних до елементів пазлу, які легко з'єднуються між собою. Кожен такий блок відповідає певній програмній конструкції або окремій функції, зокрема:

- **логічні оператори** — блоки для реалізації умовних конструкцій («якщо», «інакше») та роботи з булевими значеннями (істина/хиба);
- **цикли** — блоки, призначені для організації повторюваних дій, наприклад циклів типу *for* або *while*;
- **події** — блоки, що реагують на дії користувача (натискання кнопок, взаємодія з сенсорним екраном) або на події пристрою (отримання SMS, вхідний дзвінок тощо);
- **функції та змінні** — блоки для створення функцій, а також для оголошення, збереження та опрацювання змінних.

Відповідно до вказаних особливостей і характеристик, була створена програма, інтерфейс якої показаний на рисунку 3. 26.

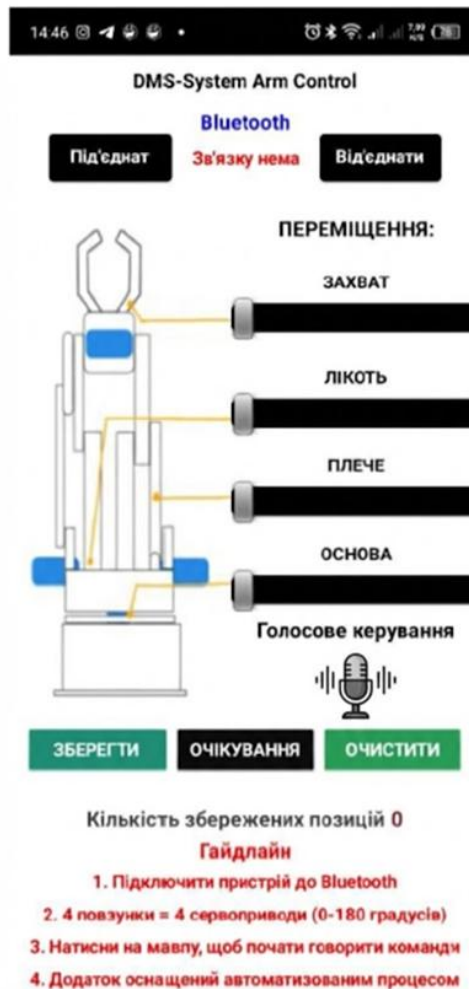


Рис. 3.26. Пристрій керування роботизованим маніпулятором

Для того, щоб почати роботу, потрібно натиснути кнопку **Під'єднати** і вибрати зі списку пристрій який потрібно, в даному випадку до модуля Bluetooth HC-05. Після успішного контакту, прозвучить звук успішної операції [21]. Сусідня кнопка **Від'єднати** слугує припиненням контакту, яке також супроводжуєть голосовою командою [21]. Потім з лівого боку ми маємо зображення руки робота на основі якого проектуємо дану систему керування, а з правого боку маємо 4 повзунків для керування сервоприводами [21]:

Для початку роботи необхідно натиснути кнопку «**Під'єднати**» та зі списку доступних пристроїв обрати потрібний, у даному випадку Bluetooth-модуль **HC-05**. У разі успішного встановлення з'єднання відтворюється звуковий сигнал, що підтверджує виконання операції [21]. Розташована поруч кнопка «**Від'єднати**» призначена для розриву з'єднання, що також супроводжується голосовим повідомленням [21].

Далі, з лівого боку інтерфейсу відображено зображення роботизованої руки, на основі якого реалізовано дану систему керування, а з правого боку розміщено чотири повзунки, що використовуються для управління сервоприводами маніпулятора.

- **Захват** – рухома частина маніпулятора, яка слугує роль руки
- робота (робочий діапазон $0^{\circ} - 90^{\circ}$) [21];
- **Рухи вперед-назад** – відтворює рухи вперед і назад “клешні”
- робота (робочий діапазон $0^{\circ} - 90^{\circ}$) [21];
- **Рухи вгору-вниз** - слугує для опускання і відповідно піднімання
- “клешні” робота (робочий діапазон $0^{\circ} - 90^{\circ}$) [21];
- **Рухи платформою** – кругові рухи всієї конструкції (робочий
- діапазон $0^{\circ} - 180^{\circ}$).

Кожен повзунок має власні початкові, мінімальні та максимальні значення, що відповідають кутовим положенням суглобів руки робота [21].

Наступним елементом є **голосове керування**, яке надсилає відповідні голосові команди у вигляді сигналів на сервоприводи для запуску їх руху. Для активації цієї функції необхідно натиснути на зображення **Мавпи з мікрофоном** [21].

У нижній частині інтерфейсу розташовано три кнопки: **ЗБЕРЕГТИ**, **ЗАПУСК** та **СКИДАННЯ**, за допомогою яких можна запрограмувати автоматичне керування рукою робота. Натискання будь-якої з кнопок супроводжується звуковим підтвердженням [21]. Нижче також розміщено мітку, що відображає кількість збережених кроків [21].

Для детального розуміння взаємодії елементів пристрою слід розглянути код розробки.

На рисунку 3.27 наведено алгоритм роботи Bluetooth-підключення до смартфона.



Рис. 3.27 Алгоритм Bluetooth модуля

На рисунку 3.28 показано алгоритм роботи повзунків для керування кожним із сервоприводів маніпулятора.



Рис. 3.28. Схема роботи повзунків для керування сервоприводами

Отже, при зміні положення повзунка за допомогою функції **Bluetooth.SendText** відповідні дані надсилаються на Arduino. Повідомлення містить префікс, що ідентифікує змінений повзунок, а також його поточне значення.

На рисунку 3.29 наведено алгоритм роботи кнопки для **ГОЛОСОВОГО** керування маніпулятором.

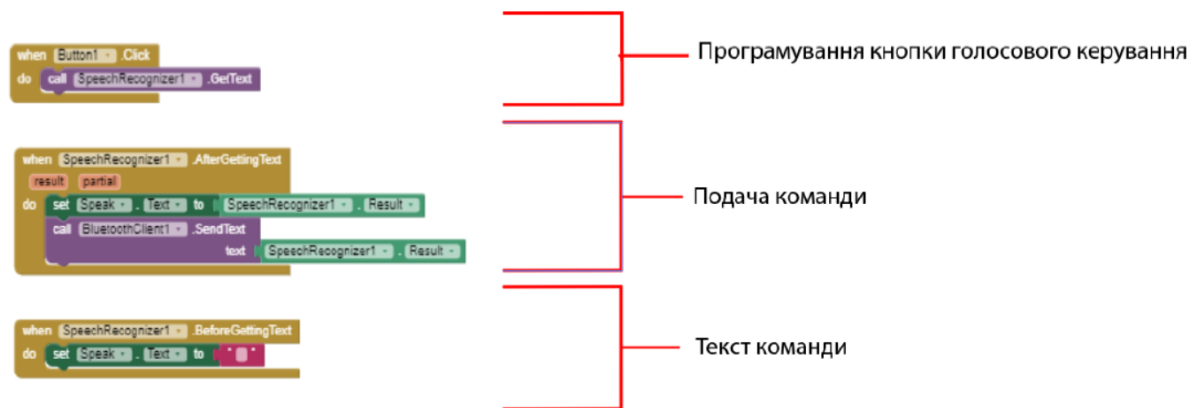


Рис. 3.29. Схема роботи кнопки для голосового керування

Після розпізнавання голосової команди програма звертається до бази даних із заздалегідь встановленими командами, які активуються при вимові певних слів. У разі збігу команди відправляється сигнал через Bluetooth, що приводить у рух відповідні сервоприводи. Схема роботи програми наведена на рисунку 3.30.

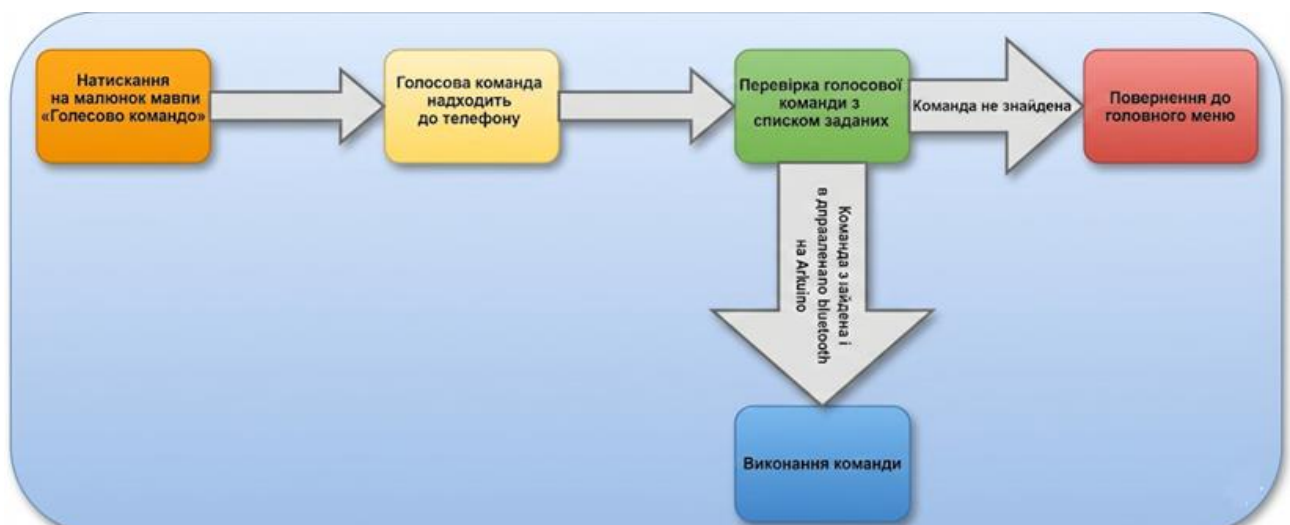


Рис. 3.30. Схема реалізації голосового керування

На рисунку 3.31 наведено алгоритми роботи кнопок **ЗБЕРЕГТИ**, **ЗАПУСК** та **СКИДАННЯ**, а також повзунка для регулювання швидкості.

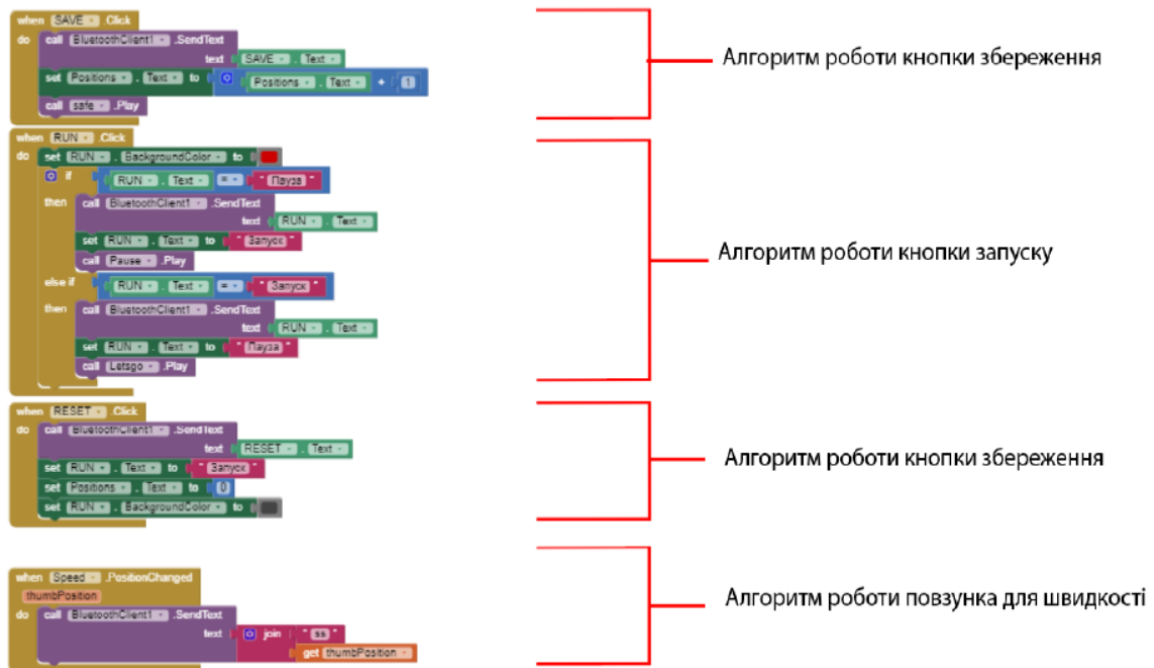


Рис. 3.31. Схема роботи кнопок і повзунка керування швидкістю

Розробка додатку системи керування на платформі Android із використанням **MIT App Inventor** дозволяє поєднати ручне та голосове управління в одному застосунку [21], що значно спрощує процес розробки та слугує прототипом для подальших робототехнічних проєктів.

Підсумкову архітектурну схему роботи системи наведено на рисунку 3.32.

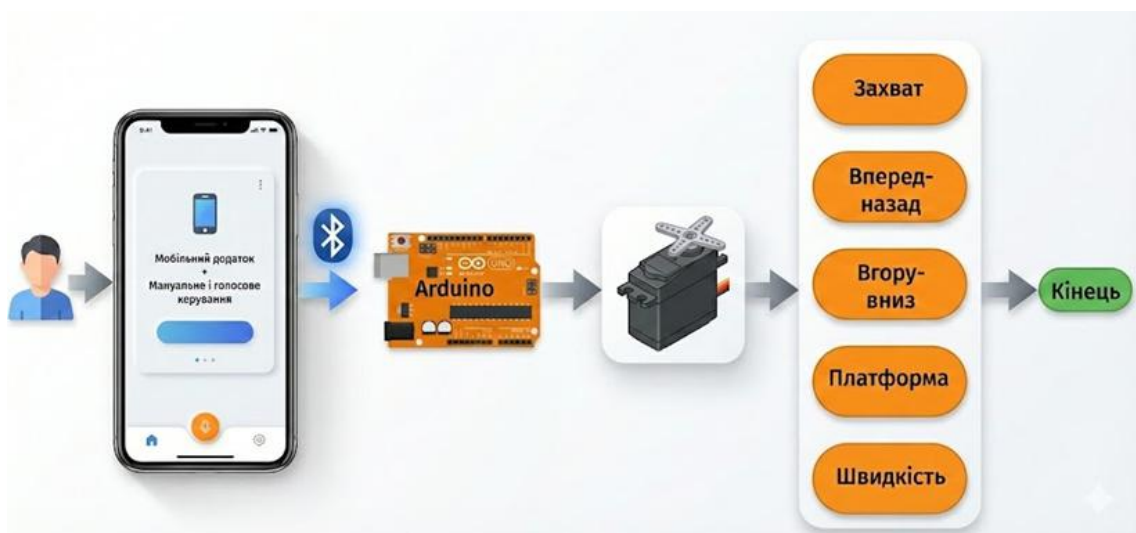


Рис. 3.32. Загальна архітектурна схема

3.5. Програмне забезпечення для автоматизованого керування роботом-маніпулятором

Розробку програмного забезпечення для автоматизованої системи керування маніпуляційним роботом доцільно здійснювати з використанням

інтегрованого середовища розробки **Arduino IDE** (див. рис. 3.33). Дане середовище є базовим інструментом для створення програм для плат Arduino та спеціально розроблене з метою спрощення процесу програмування мікроконтролерів.



Рис. 3.33. Інтерфейс та складові Arduino IDE

Програма, створена в даному програмному середовищі, називається **скетчем**. Його розробка здійснюється у вбудованому текстовому редакторі, який містить набір інструментів для вставки, видалення, заміни та пошуку фрагментів тексту. Під час експорту або збереження проєкту в області повідомлень відображаються пояснення чи повідомлення про помилки, що виникають у процесі роботи.

Для реалізації керування зібраним роботом-маніпулятором необхідно розробити програмний скетч у середовищі Arduino IDE, який забезпечує запуск і коректну роботу мікроконтролера та сервоприводів. Підключення плати до персонального комп'ютера здійснюється безпосередньо через середовище Arduino IDE.

Arduino IDE є програмним забезпеченням, орієнтованим переважно на користувачів операційної системи Windows, та надає можливість створення власних програм для мікроконтролерів сімейства Arduino. Дане середовище призначене як для розробників невеликих проєктів, так і для фахівців у галузі автоматизації та робототехніки. Воно оснащено зручним текстовим редактором, вбудованим компілятором, менеджером проєктів, а також інструментами для завантаження програмного коду безпосередньо на мікроконтролер.

Середовище Arduino IDE реалізоване на основі мови Java та поширюється як програмне забезпечення з відкритим вихідним кодом. Програмування мікроконтролерів у цьому середовищі здійснюється мовою C++ із використанням компілятора AVR-GCC.

Після інсталяції середовища на персональний комп'ютер необхідно під'єднати плату за допомогою USB-кабелю. Коректність підключення можна визначити за індикацією світлодіода на платі, миготіння якого свідчить про подачу живлення та початок виконання стандартної програми Blink, що попередньо встановлена виробником.

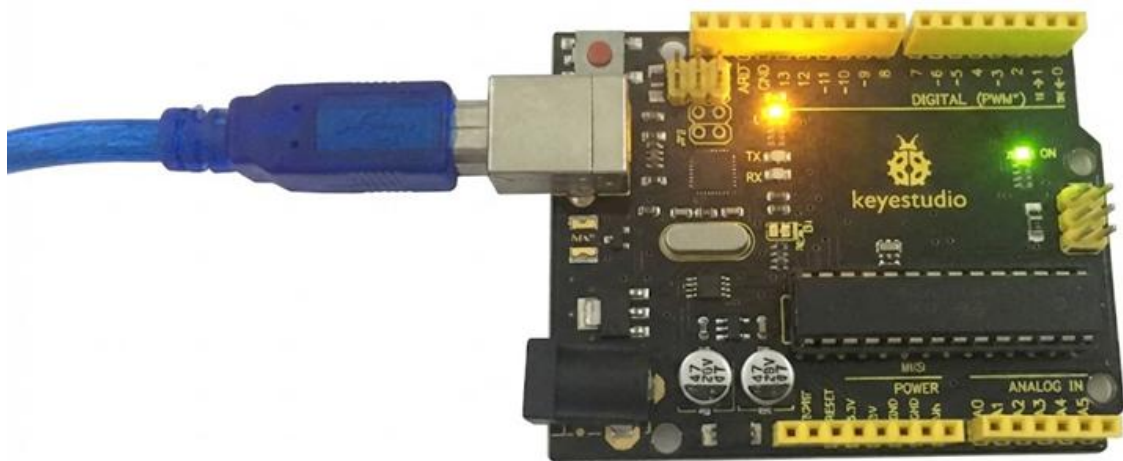


Рис. 3.34. Підключення плати

Для коректної роботи з конкретною моделлю плати необхідно налаштувати середовище розробки. З цією метою в меню «Інструменти» слід обрати пункт «Плата» та вказати модель плати, з якою передбачається робота.

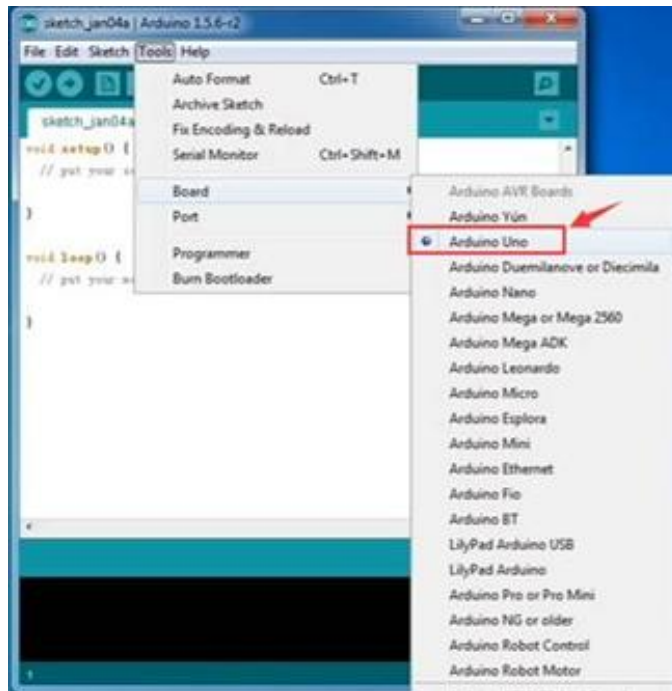


Рис. 3.35. Вибір плати

Керування сервоприводами робота-маніпулятора здійснюється за допомогою бібліотеки Servo, яка надає такі основні функції:

```
uint8_t attach(int pin);
```

```
uint8_t attach(int pin, int min, int max);
```

Зазначені функції дозволяють визначити вивід мікроконтролера, до якого підключений сервопривод. Синтаксис використання команд має такий вигляд:

```
servo.attach(pin);
```

```
servo.attach(pin, min, max);
```

де Servo — об'єкт класу Servo;

pin — номер порту, до якого під'єднано серводвигун;

min — тривалість імпульсу в мікросекундах, що відповідає положенню сервоприводу на 0° (за замовчуванням 544);

max — тривалість імпульсу в мікросекундах, що відповідає положенню сервоприводу на 180° (за замовчуванням 2400).

Калібрування відповідних сервоприводів до заданих кутових положень здійснюється за допомогою програмного коду, наведеного нижче:

```
#include <Servo.h>
```

```
Servo myservo; // Створення відповідної змінної
```

```

void setup()
{
    Serial.begin(9600);
    delay(1000);
}
void loop()
{
    myservo.attach(A1); // Відповідний пін сервоприводу
    myservo.write(0); // Заданий кут
    delay(1000);
}

```

Користувач задає необхідні параметри, і система виконує їх автоматично до завершення роботи.

Фрагмент програмного коду, що реалізує цю функцію, виглядає наступним чином:

```

if (bt.startsWith("SAVE")) {
    if (index < 50) { // Перевірка виходу за межі масиву
        s1[index] = s1Ant; s2[index] = s2Ant; s3[index] = s3Ant;
        s4[index] = s4Ant; index++;
    }
}
if (bt.startsWith("RESET")) {
    memset(s1, 0, sizeof(s1));
    memset(s2, 0, sizeof(s2));
    memset(s3, 0, sizeof(s3));
    memset(s4, 0, sizeof(s4));
    index = 0;
}
if (bt.startsWith("RUN")) {
    while (true) {

```

```

for (int i = 0; i < index; i++) {
    if (Bluetooth.available() > 0) {
        bt = Bluetooth.readString(); bt.trim();
        if (bt.startsWith("PAUSE")) {
            while (true) {
                if (Bluetooth.available() > 0) {
                    bt = Bluetooth.readString(); bt.trim();
                    if (bt.startsWith("RUN")) break;
                    if (bt.startsWith("RESET")) return;
                }
            }
        }
        if (bt.startsWith("RESET")) return;
        if (bt.startsWith("ss")) {
            btS = bt.substring(2); velG = btS.toInt();
        }
    }
    executeSteps(i, velG);
}
}
}

```

Після завершення розробки програмного коду та його завантаження до плати Arduino робот-маніпулятор переходить у робочий режим. Подальша робота здійснюється автономно, без необхідності підключення до персонального комп'ютера, а керування виконується в ручному режимі. Для кращого розуміння принципу дії розглянемо алгоритм функціонування робота-маніпулятора, який відображає логіку процесу керування.

Повний програмний код для керування роботом-маніпулятором наведено в додатку Б.

3.6. Тестування автоматизованої маніпуляційної системи

Тестування автоматизованих робототехнічних систем є одним із ключових етапів їх розроблення та впровадження, оскільки воно забезпечує надійність, точність функціонування та відповідність заданим технічним вимогам. Проведення тестування дає змогу виявити й усунути потенційні недоліки, підвищити безпеку експлуатації та гарантувати ефективне виконання поставлених завдань.

Розроблена система також підлягає обов'язковому тестуванню з метою мінімізації можливих ризиків, серед яких можна виділити:

Апаратні збої:

- несправності окремих компонентів (драйвери, сервоприводи, контролер);
- пошкодження елементів унаслідок перенапруги або неправильної експлуатації.

Програмні помилки:

- некоректне програмування компонентів системи;
- збої в роботі алгоритмів або у процесі обміну даними між мобільним додатком і роботом;
- непередбачувана поведінка системи в реальних умовах експлуатації.

Перший етап має підготовчий характер і передбачає визначення основних критеріїв тестування, зокрема перевірку роботи сервоприводів, стабільності Bluetooth-з'єднання, справності механізму маніпулятора, коректності з'єднання компонентів, а також правильності підключення джерела живлення.

На рисунку 3.36 наведено прототип маніпуляційної системи.

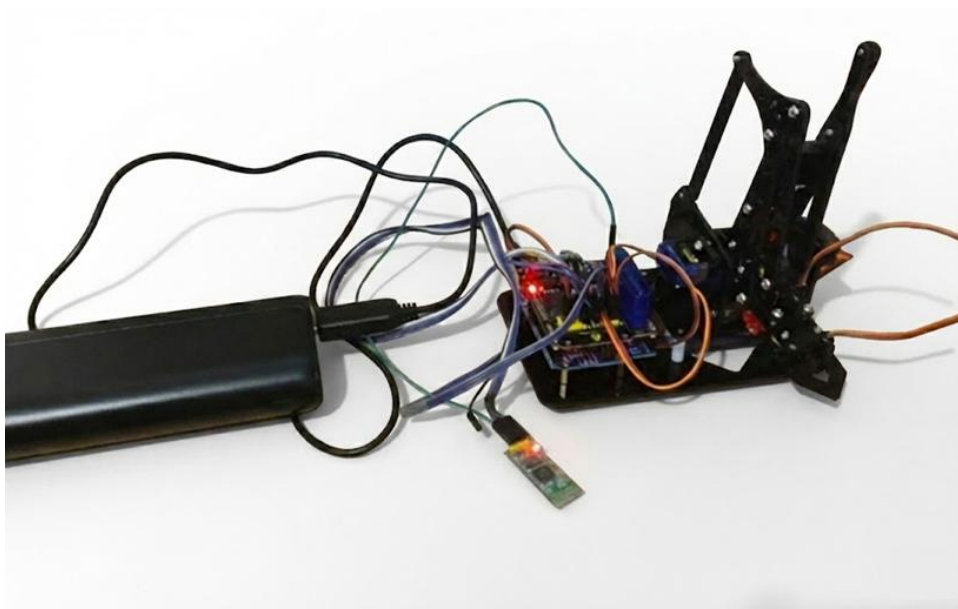


Рис. 3.36. Прототип маніпуляційної системи

Наступним етапом є тестування програмного забезпечення. На цьому етапі виконується перевірка працездатності алгоритмів у віртуальному середовищі без залучення реального обладнання, а також оцінюється коректність виконання кожної функції мобільного додатку. Для цього використовується електрична схема, розроблена в середовищі Proteus та наведена на рисунку 3.22. Емулятор мобільного застосунку на персональному комп'ютері безперешкодно підключається до схеми за допомогою Bluetooth-з'єднання.

Оскільки додаток підтримує два режими керування, обидва режими підлягають обов'язковому тестуванню. Процес перевірки голосового керування представлено на рисунку 3.37.

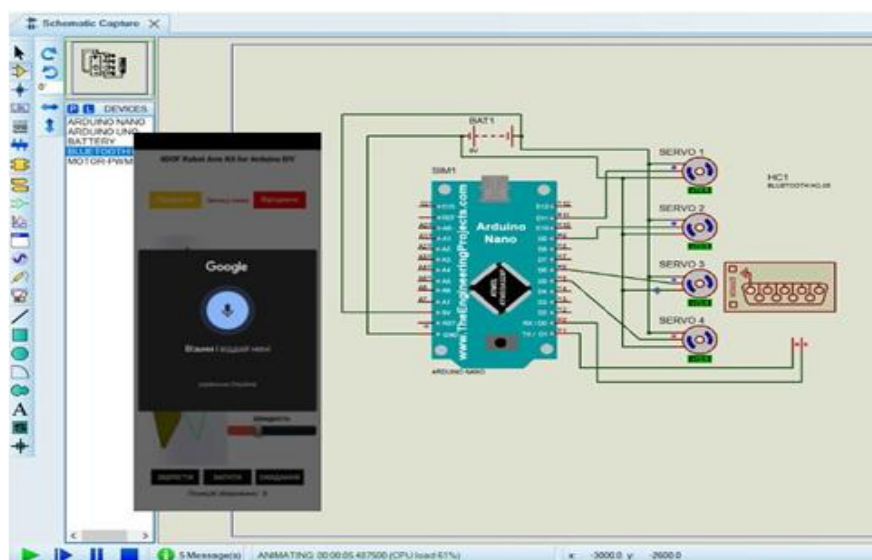


Рис. 3.37. Тестування голосового керування

Після подачі голосової команди «Візьми і віддай мені» сигнал передається на Bluetooth-модуль HC-05, а далі надходить до мікроконтролера, що приводить у дію чотири сервоприводи, відповідальні за переміщення механізму робота на задану траєкторію руху.

Тестування мануального режиму керування, тобто ручного управління маніпулятором, наведено на рисунку 3.38.

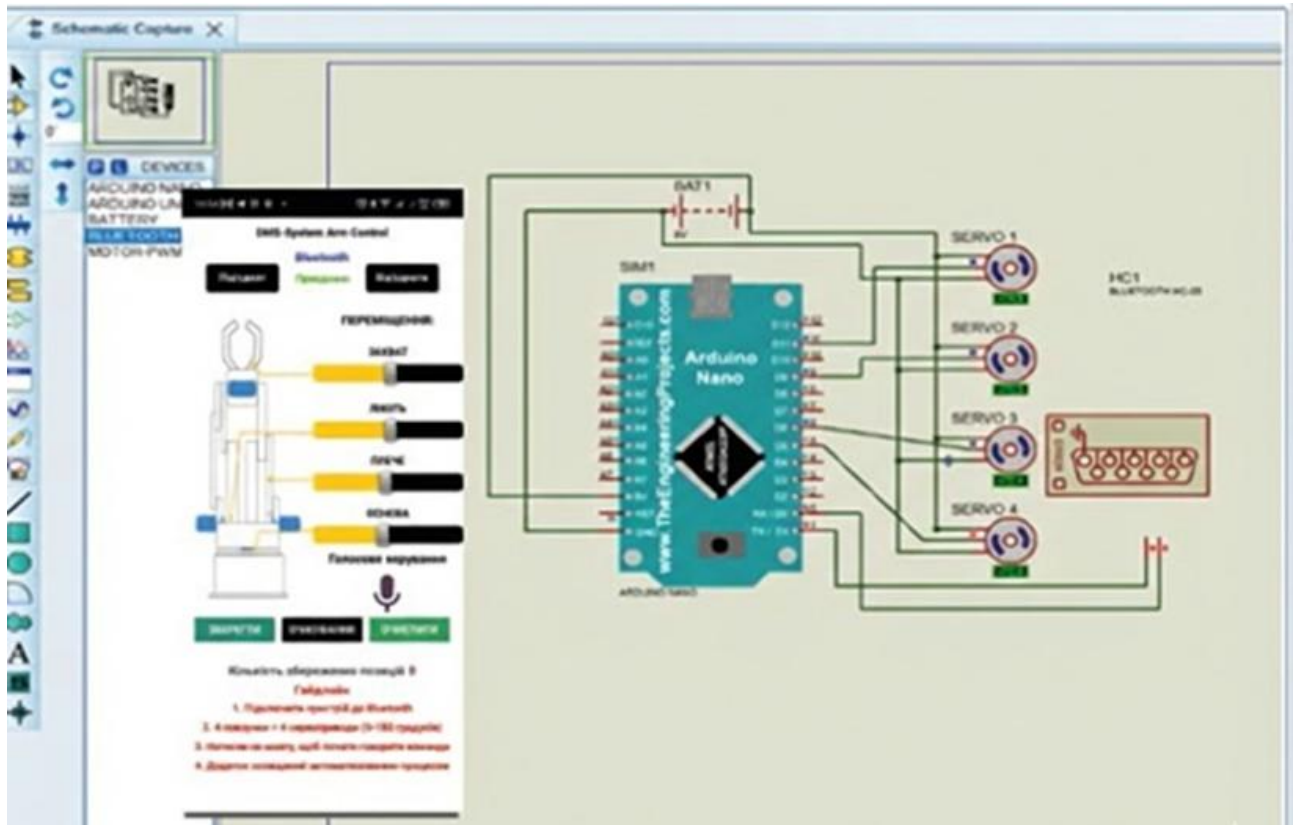


Рис. 3.38. Тестування мануального режиму керування

За результатами випробувань встановлено, що програма функціонує коректно, однак рухи маніпулятора є недостатньо плавними. Сервоприводи реагують на команди, що задаються за допомогою повзунків, та повертаються на відповідні кути. Для більш детального аналізу роботи системи необхідно підключити мобільний додаток до фізичного прототипу маніпулятора, що відповідає третьому етапу тестування.

На рисунку 3.39 показано процес інтеграції мобільного додатку з експериментальним прототипом маніпулятора.

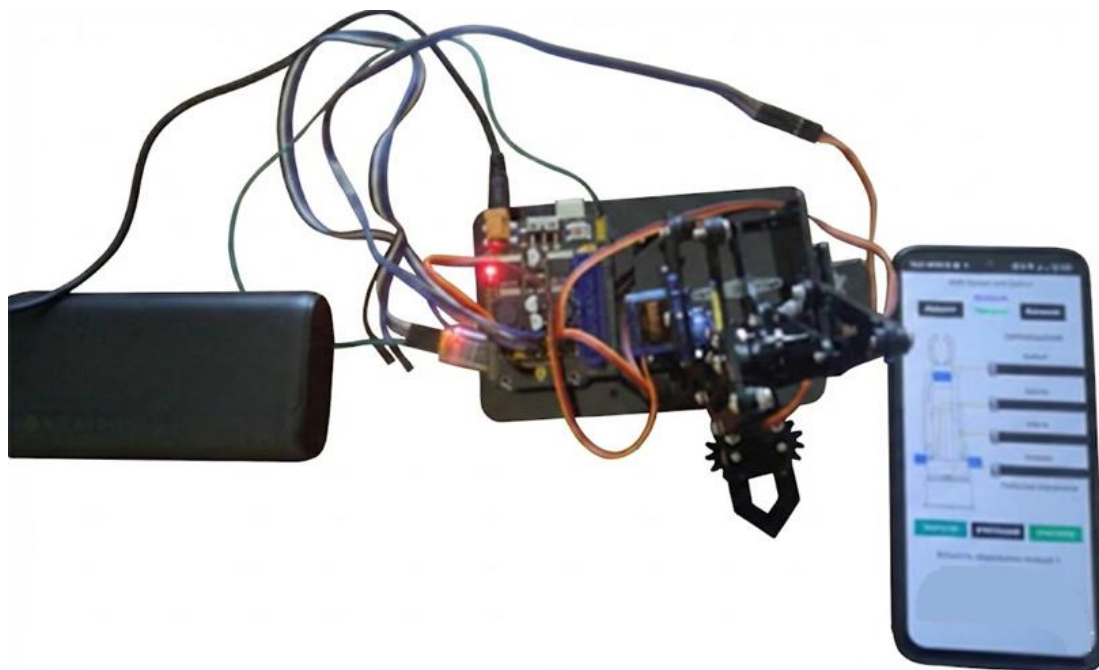


Рис. 3.39. Інтеграція мобільного додатку з прототипом маніпулятора

Під час тестування було виявлено, що команди, які надходять із додатку, спричиняють перегрів і надмірне навантаження на сервоприводи, що може призвести до їхньої швидкої поломки. Усунення проблеми було досягнуто шляхом корекції програмного коду та системи подачі живлення.

Зокрема, для плавнішої роботи сервоприводів значення у команді `Serial.begin(38600);` було змінено на `Serial.begin(9600);`. Щодо живлення, кабель, підключений до входу 5 V, було замінено на вихід 9 V, що забезпечило стабільну роботу системи.

Висновки до розділу

У третьому розділі проведено комплексне проєктування та практичну реалізацію автоматизованої системи керування маніпуляційним роботом із функцією голосового управління. За результатами виконаних робіт можна зробити такі висновки:

1. Розроблено архітектуру та структурну схему системи. Визначено ключові функціональні вузли: мобільний пристрій на базі Android (командний модуль), Bluetooth-модуль HC-05 (канал зв'язку), мікроконтролер Keyestudio UNO (обробка даних) та сервоприводи SG90 (виконавчі механізми). Така структура забезпечує гнучкість керування та надійний зворотний зв'язок.

2. Обґрунтовано вибір елементної бази. Для реалізації апаратної частини обрано платформу Keyestudio Uno R3 та плату розширення Keyestudio Servo Motor Driver Shield, що дозволило спростити схемотехніку та забезпечити стабільне керування чотирма ступенями свободи маніпулятора. Використання мікросервоприводів SG90 забезпечило необхідну кінематику при збереженні компактності та низького енергоспоживання конструкції.

3. Спроектовано та змодельовано електричну схему. Розроблено принципову схему підключення компонентів, працездатність якої попередньо перевірено шляхом моделювання у середовищі Proteus. Це дозволило уникнути помилок при фізичному монтажі та забезпечити коректну комутацію силових і сигнальних ліній.

4. Створено програмне забезпечення для мобільного керування. У середовищі MIT App Inventor розроблено Android-додаток, який поєднує два режими роботи: ручний (через графічні повзунки) та голосовий (з використанням Google Voice Recognition API). Реалізовано функціонал збереження послідовності дій, їх відтворення та екстреного зупинення роботи.

5. Реалізовано алгоритми керування мікроконтролером. У середовищі Arduino IDE написано скетч, що забезпечує приймання та парсинг команд через Bluetooth-інтерфейс, керування ШІМ-сигналами для сервоприводів, а також обробку сценаріїв автоматичного відтворення збережених рухів.

6. Проведено комплексне тестування. Виконано перевірку системи на етапах віртуальної симуляції та на фізичному прототипі. Виявлено та усунено недоліки, пов'язані з плавністю ходу сервоприводів та стабільністю живлення (оптимізовано швидкість передачі даних Serial-порту до 9600 бод та переведено живлення на джерело 9 В).

ВИСНОВКИ

У кваліфікаційній роботі вирішено актуальну науково-прикладну задачу створення системи керування роботом-маніпулятором на базі платформи Arduino. На основі теоретичних досліджень, моделювання та практичних експериментів отримано такі результати:

1. **Проведено аналіз сучасного стану робототехніки.** Розглянуто конструктивні особливості та системи керування провідних промислових роботів (Universal Robots, Fanuc, KUKA, ABB). Встановлено, що для лабораторних та навчальних цілей економічно доцільним є використання маніпуляторів з електричними сервоприводами та відкритою архітектурою керування на базі мікроконтролерів AVR (Arduino).
2. **Розроблено математичну та конструкторську моделі.** Обґрунтовано вибір ангулярної кінематичної схеми, яка забезпечує компактність та достатню робочу зону. За допомогою методу Денавіта–Хартенберга та рівнянь Лагранжа побудовано математичну модель динаміки чотириланкового маніпулятора, що дозволило описати рух ланок під дією керуючих моментів. У середовищі Autodesk Inventor спроєктовано 3D-модель механічної частини, адаптовану для виготовлення методом 3D-друку.
3. **Спроєктовано апаратну частину системи керування.** Розроблено структурну та принципову електричну схеми пристрою. В якості ядра системи обрано платформу Keystudio UNO R3 у поєднанні з драйвером Keystudio Servo Motor Driver Shield, що забезпечило стабільне керування чотирма сервоприводами SG90. Для організації бездротового зв'язку інтегровано модуль Bluetooth HC-05.
4. **Реалізовано програмне забезпечення.**
 - Створено скетч для мікроконтролера в середовищі Arduino IDE, який забезпечує прийом команд, генерацію ШІМ-сигналів для сервоприводів та реалізацію алгоритмів автоматичного відтворення рухів.

- Розроблено мобільний додаток у середовищі MIT App Inventor для ОС Android. Додаток підтримує два режими роботи: ручне керування повзунками та голосове керування (з використанням Google Voice Recognition API), а також функції збереження та циклічного відтворення траєкторій.

5. Проведено комплексне тестування системи. Працездатність електричної схеми попередньо перевірено шляхом моделювання у програмному середовищі Proteus. Фізичні випробування прототипу підтвердили коректність роботи алгоритмів керування. У ході тестування було виявлено та усунено недоліки, пов'язані з живленням сервоприводів (перехід на джерело 9В) та налаштуванням швидкості передачі даних (9600 бод), що забезпечило плавність рухів маніпулятора.

Таким чином, мета роботи досягнута: створено функціональну автоматизовану систему керування роботом-маніпулятором, яка здатна виконувати команди оператора дистанційно, зокрема за допомогою голосового інтерфейсу.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Кльон А. М., Кондратовіч А. С. Використання Arduino-симулятора WOKWI на лабораторних роботах з робототехніки. *Збірник наукових праць Донбаської національної академії будівництва і архітектури*. 2024. № 2 (33). С. 79–83. URL: <https://donnaba.edu.ua/journal/wp-content/uploads/2024/06/79-83.pdf> (дата звернення: 16.11.2025).
2. Кошель С. О., Ковальов Ю. А., Манойленко О. П. Проектування промислових роботів та маніпуляторів : навч. посіб. Київ : Центр навчальної літератури, 2019. 256 с.
3. Ловейкін В. С., Ромасевич Ю. О., Човнюк Ю. В. Мехатроніка : навч. посіб. Київ, 2012. 357 с.
4. Матвієнко М. П. Основи електротехніки та електроніки : підручник. Київ : Ліра-К, 2021. 506 с.
5. Морзе Н. В., Варченко-Троценко Л. О., Гладун М. А. Основи робототехніки : навч. посіб. Кам'янець-Подільський : П. П. Буйницький О. А., 2016. 184 с.
6. STEM-освіта. *Інститут модернізації змісту освіти*. URL: <https://imzo.gov.ua/stem-osvita> (дата звернення: 16.11.2025).
7. Трет'як А. В., Кльон А. М. Основи робототехніки : навч. посіб. Полтава : Нац. ун-т «Полт. політехніка ім. Юрія Кондратюка», 2024. 135 с.
8. Arduino. Open-source electronics platform. URL: <https://www.arduino.cc> (дата звернення: 06.12.2025).
9. AUTODESK Tinkercad. URL: <https://www.tinkercad.com/dashboard> (дата звернення: 06.12.2025).
10. Basics: Electric motor. *Baum Mueller*. URL: <https://www.baumueller.com/en/insights/basics/electric-motor> (дата звернення: 06.12.2025).
11. Bishop O. Robot Builder's Cookbook: Build and Design Your Own Robots. Elsevier Ltd, 2007. 366 p.

12. Boxall J. Arduino Workshop: A Hands-On Introduction with 65 Projects. 2nd ed. No Starch Press, 2021. 432 p.
13. Difference between MCU and SoC. *GeeksforGeeks*. URL: <https://www.geeksforgeeks.org/difference-between-mcu-and-soc> (дата звернення: 07.12.2025).
14. Drive Mechanism. *ERC BPGC Handbook*. URL: https://erc-bpgc.github.io/handbook/mechanical/drive_mechanism (дата звернення: 18.11.2025).
15. Global Robotics Race: Korea, Singapore and Germany in the Lead. *IFR*. URL: <https://ifr.org/ifr-press-releases/news/global-robotics-race-korea-singapore-and-germany-in-the-lead> (дата звернення: 18.11.2025).
16. Henry N. Understanding Industrial Robot Motion Types. *SolisPLC*. URL: <https://www.solisplc.com/tutorials/industrial-robot-motion-types> (дата звернення: 18.11.2025).
17. Holonomic Drive System. Pat Capulong. URL: <https://www.patcapulong.com/work/holonomic-drive> (дата звернення: 19.11.2025).
18. How do you create a communication system for multiple robots. *LinkedIn Advice*. URL: <https://www.linkedin.com/advice/3/how-do-you-create-communication-system-multiple-robots> (дата звернення: 18.11.2025).
19. Linear Motor. *A&M C*. URL: <https://www.a-m-c.com/experience/technologies/motortype/linear> (дата звернення: 16.11.2025).
20. Nussey J. Arduino For Dummies. 2nd ed. John Wiley & Sons, Inc, 2018. 386 p.
21. Автоматизована система управління маніпуляційними роботами / М. Філіппова та ін. *Herald of Khmelnytskyi National University. Technical Sciences*. 2024. Vol. 341, no. 5. P. 466–476. DOI: <https://doi.org/10.31891/2307-5732-2024-341-5-68>

22. PLANOBOT Pick-and-Place Industrial Robot – Joe B. Brown (Planet Corp., 1956). *Cyberneticzoo*. URL: <https://cyberneticzoo.com/early-industrial-robots/1956-planobot-pick-and-place-industrial-robot-joe-b-brown-planet-corp> (дата звернення: 26.11.2025).
23. Raspberry Pi. URL: <https://www.raspberrypi.com> (дата звернення: 26.11.2025).
24. Robot communication methods. *Automate UK*. URL: <https://www.automate-uk.com/our-associations/bara/expert-advice/robots-explained/robot-communication-methods> (дата звернення: 26.11.2025).
25. Stepper Motor vs. Servo Motor: A Comprehensive Comparison. *Lam Motor*. URL: <https://lammotor.com/stepper-motor-vs-servo-motor> (дата звернення: 26.11.2025).
26. Top 5 Robot Trends 2024. *IFR*. URL: <https://ifr.org/ifr-press-releases/news/top-5-robot-trends-2024> (дата звернення: 19.11.2025).
27. What are manipulator robots? Understanding their Design, Types, and Applications. *Wevolver*. URL: <https://www.wevolver.com/article/robot-manipulator> (дата звернення: 19.11.2025).
28. WOKWI: Simulate IoT Projects in Your Browser. URL: <https://wokwi.com> (дата звернення: 19.11.2025).
29. Арі В., Мондада Ф. Елементи робототехніки. Springer Nature, 2018.
30. Мокренко П. В., Ядловська В. В. Огляд розвитку робототехніки. Частина 1. (Робототехніка до XX століття). *Автоматика, вимірювання та керування*. 2020. Т. 2, № 1(2). С. 67–77. DOI: <https://doi.org/10.23939/amm2020.01.067>
31. Robotic colorectal surgery and ergonomics / S. W. Wong et al. *J Robot Surg*. 2022. Vol. 16. P. 241–246.
32. Алешко К. А. Проблема автоматизованого визначення механічної взаємодії колісної робототехнічної платформи із зовнішнім середовищем. *Automation and Development of Electronic Devices*. Харків : ХНУРЕ, 2021. № 2. С. 14–18. URL:

- <https://openarchive.nure.ua/server/api/core/bitstreams/bbf82b13-41c1-442ea4ea-81dbf980fa5e/content> (дата звернення: 17.11.2025).
33. Комп'ютерні технології: інновації, проблеми, рішення – 2017 : тези доповідей II Міжнар. наук.-техн. конф. (Житомир, 17–19 жовт. 2017 р.). Житомир : Вид. О. О. Євенок, 2017. 252 с.
 34. Спосіб і система управління маніпуляційним роботом : пат. US9949877B2 Китай. URL: <https://patents.google.com/patent/CN114174008A/en> (дата звернення: 13.11.2025).
 35. Hartenberg R. S., Denavit J. A kinematic notation for lower pair mechanisms based on matrices. *Journal of Applied Mechanics*. 1955. Vol. 77. P. 215–221.
 36. Barinka L., Berka R. Inverse kinematics-basic methods. 2002. URL: <http://www.cescg.org/CESCG-2002/LBarinka/paper.pdf> (дата звернення: 17.11.2025).
 37. Aristidou A., Lasenby J. Inverse kinematics: a review of existing techniques and introduction of a new fast iterative solver. 2009.
 38. Featherstone R., Orin D. Robot Dynamics: Equations and Algorithms. *Proceedings of the 2000 IEEE International Conference on Robotics & Automation* (San Francisco, April 2000). 2000.
 39. Довідкова інформація IRB 460. URL: <https://library.e.abb.com/public/a7121292272d40a9992a50745fdaa3b2/3HAC041346%20PS%20IRB%20140-en.pdf> (дата звернення: 17.11.2025).
 40. Довідкова інформація Arduino. URL: <https://docs.keyestudio.com/projects/KS0198/en/latest/KS0198X.html> (дата звернення: 17.11.2025).
 41. Murray R. M., Li Z., Sastry S. S. A Mathematical Introduction to Robotic Manipulation. CRC press, 1994.

42. IBM Podcast: Piper, Diaz, Nipper – MQTT. URL: http://www.ibm.com/podcasts/software/websphere/connectivity/piper_diaz_nipper_mqtt_11182011.pdf (дата звернення: 10.11.2025).
43. Verma S. Hand gestures remote controlled robotic arm. *Advance in Electronic and Electric Engineering*. 2013. Vol. 3, no. 5. P. 601–606.
44. Olson C. F. Probabilistic self-localization for mobile robots. *IEEE Transactions on Robotics and Automation*. 2000. Vol. 16, no. 1. P. 55–66. DOI: 10.1109/70.833191
45. Розв'язок оберненої кінематики та верифікація гідравлічного маніпулятора 4-DOF / Ч. Чао та ін. *Journal of Physics: Conference Series*. 2017. Vol. 916, no. 1.
46. Design of a 4-Dof робота-маніпулятора з оптимізованим алгоритмом зворотної кінематики / S. Gomez et al. *International Journal of Mechanism and Machine Theory*.
47. Žlajpah L. Simulation in robotics. *Math. Comput. Simul.* 2008. Vol. 79, no. 4. P. 879–897.
48. Мирза Д. С., Філіпова М. В. Управління маніпуляційними роботами на основі математичного об'єктного моделювання. *Ефективність та автоматизація інженерних рішень у приладобудуванні* : зб. праць XIX наук.-практ. конф. студентів, аспірантів та молодих вчених. Київ : ПБФ КПІ ім. Ігоря Сікорського, 2023.
49. Singh A., Singla A. Kinematic modeling of robotic manipulators. *Proceedings of the National Academy of Sciences, India Section A: Physical Sciences*. 2017. P. 303–319.
50. Kinematic and dynamic behavior of two-link articulated robot manipulators / F. C. Cardoso et al. *ABCM Symposium Series in Mechatronics*. 2012. Vol. 5. P. 1132–1141.
51. Ченг Ф. С. Калібрування опорних систем роботів для підвищення точності позиціонування робота. *Нові технології — тенденції*,

- інновації та дослідження / за ред. С. Volosencu. Рієка : InTech, 2012. Гл. 5.
52. Марич П., Джалич В. Підвищення точності та гнучкості промислових роботів за допомогою комп'ютерного зору. *Нові технології — тенденції, інновації та дослідження* / за ред. С. Volosencu. Рієка : InTech, 2012. Гл. 7.
53. Constraint-combined force/position hybrid control method with Lyapunov stability / F. Yu et al. *Proceedings of the SICE Annual Conference* (Tokyo, 13–18 Sept. 2011). 2011. P. 671–676.
54. Гавриш О. А., Бояринова К. О., Копішинська К. О. Розробка стартап-проектів : навч. посіб. Київ : КПІ ім. Ігоря Сікорського, 2019. 188 с. URL: <http://ela.kpi.ua/handle/123456789/29447> (дата звернення: 17.11.2025).
55. Комп'ютерне моделювання процесів і систем. Методи оптимізації : підручник / С. П. Вислоух та ін. Київ : КПІ ім. Ігоря Сікорського, 2023. 267 с.
56. Комп'ютерне моделювання процесів та систем. Чисельні методи : підручник / С. П. Вислоух та ін. Київ : КПІ ім. Ігоря Сікорського, 2021. 228 с.
57. Simulation-based planning of machining processes with industrial robots / J. Brüning et al. *Procedia Manufacturing*. 2016. Vol. 6. P. 17–24. DOI: <https://doi.org/10.1016/j.promfg.2016.11.003>
58. Здолбіцька Н. В., Ковальчук П. Є. Система керуванням маніпулятором. *Програмовані логічні інтегральні схеми та мікропроцесорна техніка в освіті і виробництві* : матеріали Міжнар. наук.-практ. семінару молодих вчених та студентів (Луцьк, 12–13 трав. 2020 р.) / відп. ред. П. А. Пех. Луцьк : Вежа-Друк, 2020. С. 45.
59. Robotic arc welding sensors and programming in industrial applications / P. Kah et al. *International Journal of Mechanical and Materials*

Engineering. 2015. Vol. 10, no. 1. DOI: <https://doi.org/10.1186/s40712-015-0042-y>

60. Галєб Н. М., Алї А. А. Моделювання та керування 2-DOF маніпулятором робота. *Міжнародний журнал передових інженерних досліджень і технологій*. 2018. Т. 6, № 11. С. 24–31.

ДОДАТОК А

Скетч керування автоматизованою системою

```
#include <Servo.h>
#include <SoftwareSerial.h>

SoftwareSerial Bluetooth(A4, A3); // RX, TX

Servo servo1; // Основа
Servo servo2; // Плече
Servo servo3; // Лікоть
Servo servo4; // Захват

int s1Vel = 20; // Основа
int s2Vel = 20; // Плече
int s3Vel = 20; // Лікоть
int s4Vel = 20; // Захват

int s1Ant = 0;
int s2Ant = 0;
int s3Ant = 0;
int s4Ant = 0;

int velG = 25; // Загальна швидкість
int index = 0; // Індекс запису позицій
int s1[50], s2[50], s3[50], s4[50]; // Масиви позицій
String bt, btS;

void setup() {
    Serial.begin(9600);
    Bluetooth.begin(9600);
    Bluetooth.setTimeout(10);
    servo1.attach(A0); // Основа
    servo2.attach(9); // Плече
    servo3.attach(6); // Лікоть
    servo4.attach(5); // Захват
    servo1.write(s1Ant);
```

```

servo2.write(s2Ant);
servo3.write(s3Ant);
servo4.write(s4Ant);
delay(50);
}
void loop() {
  if (Bluetooth.available() > 0) {
    bt = Bluetooth.readString();
    bt.trim();
    // == ЗВИЧАЙНІ КОМАНДИ ==
    if (bt.startsWith("s2")) {
      btS = bt.substring(2);
      s1Ant = constrain(btS.toInt(), 0, 180);
      moveServo(servo1, s1Ant, s1Vel);
    }
    if (bt.startsWith("s4")) {
      btS = bt.substring(2);
      s2Ant = constrain(btS.toInt(), 0, 90);
      moveServo(servo2, s2Ant, s2Vel);
    }
    if (bt.startsWith("s5")) {
      btS = bt.substring(2);
      s3Ant = constrain(btS.toInt(), 0, 90);
      moveServo(servo3, s3Ant, s3Vel);
    }
    if (bt.startsWith("s6")) {
      btS = bt.substring(2);
      s4Ant = constrain(btS.toInt(), 0, 90);
      moveServo(servo4, s4Ant, s4Vel);
    }
  }
}

```

// ===== ЗАПИС ТА ВІДТВОРЕННЯ ПОСЛІДОВНОСТЕЙ =====

```
if (bt.startsWith("SAVE")) {  
    if (index < 50) {  
        s1[index] = s1Ant;  
        s2[index] = s2Ant;  
        s3[index] = s3Ant;  
        s4[index] = s4Ant;  
        index++;  
    }  
}  
  
if (bt.startsWith("RESET")) {  
    memset(s1, 0, sizeof(s1));  
    memset(s2, 0, sizeof(s2));  
    memset(s3, 0, sizeof(s3));  
    memset(s4, 0, sizeof(s4));  
    index = 0;  
}  
  
if (bt.startsWith("RUN")) {  
    while (true) {  
        for (int i = 0; i < index; i++) {  
            if (Bluetooth.available() > 0) {  
                bt = Bluetooth.readString();  
                bt.trim();  
                if (bt.startsWith("PAUSE")) {  
                    while (true) {  
                        if (Bluetooth.available() > 0) {  
                            bt = Bluetooth.readString();  
                            bt.trim();  
                            if (bt.startsWith("RUN")) break;  
                            if (bt.startsWith("RESET")) return;  
                        }  
                    }  
                }  
            }  
        }  
    }  
}
```

```

        }
    }
}
if (bt.startsWith("RESET")) return;
if (bt.startsWith("ss")) {
    btS = bt.substring(2);
    velG = btS.toInt();
}
}
executeSteps(i, velG);
}
}
}
// ===== ГОЛОСОВІ КОМАНДИ =====
if (bt == "вниз") {
    s2Ant = constrain(s2Ant - 25, 0, 90);
    moveServo(servo2, s2Ant, s2Vel);
}
if (bt == "верх") {
    s2Ant = constrain(s2Ant + 25, 0, 90);
    moveServo(servo2, s2Ant, s2Vel);
}
if (bt == "назад") {
    s3Ant = constrain(s3Ant - 25, 0, 90);
    moveServo(servo3, s3Ant, s3Vel);
}
if (bt == "вперед") {
    s3Ant = constrain(s3Ant + 25, 0, 90);
    moveServo(servo3, s3Ant, s3Vel);
}
}

```

```

if (bt == "Відкрий") {
    s4Ant = constrain(s4Ant + 25, 0, 90);
    moveServo(servo4, s4Ant, s4Vel);
}
if (bt == "закрий") {
    s4Ant = constrain(s4Ant - 25, 0, 90);
    moveServo(servo4, s4Ant, s4Vel);
}
if (bt == "вліво") {
    s1Ant = constrain(s1Ant - 25, 0, 180);
    moveServo(servo1, s1Ant, s1Vel);
}
if (bt == "вправо") {
    s1Ant = constrain(s1Ant + 25, 0, 180);
    moveServo(servo1, s1Ant, s1Vel);
}
}
}
// ===== ФУНКЦІЇ РУХУ =====
void moveServo(Servo& servo, int to, int speed) {
    int from = servo.read();
    if (from > to) {
        for (int pos = from; pos >= to; pos--) {
            servo.write(pos);
            delay(speed);
        }
    } else {
        for (int pos = from; pos <= to; pos++) {
            servo.write(pos);
            delay(speed);
        }
    }
}

```



```
    }  
  }  
}  
void executeSteps(int i, int speed) {  
  moveServo(servo1, s1[i], speed);  
  moveServo(servo2, s2[i], speed);  
  moveServo(servo3, s3[i], speed);  
  moveServo(servo4, s4[i], speed);  
}
```

ДОДАТОК Б

Код додатку системи керування

```
{
  "$components": [
    {
      "$Name": "RESET",
      "$Type": "Button",
      "$Version": "7",
      "BackgroundColor": "&HFF1F9950",
      "FontBold": "True",
      "FontSize": "12",
      "Height": "35",
      "Width": "100",
      "Text": "ОЧИСТИТИ",
      "TextColor": "&HFFFFFFF",
      "Uuid": "-2136887197"
    },
    {
      "$Name": "Label9",
      "$Type": "Label",
      "$Version": "5",
      "FontBold": "True",
      "Text": "Кількість збережених позицій",
      "TextColor": "&HFF444444",
      "Uuid": "818781082"
    },
    {
      "$Name": "Label14",
      "$Type": "Label",
      "$Version": "5",
```

```
"FontBold": "True",
"Text": "Гайдлайн",
"TextColor": "&HFFFF0000",
"Uuid": "-1345450484"
},
{
"$Name": "Label15",
"$Type": "Label",
"$Version": "5",
"FontBold": "True",
"FontSize": "12.0",
"Text": "1. Підключити пристрій до Bluetooth",
"TextColor": "&HFFFF0000",
"Uuid": "-40247243"
},
{
"$Name": "Label16",
"$Type": "Label",
"$Version": "5",
"FontBold": "True",
"FontSize": "12.0",
"Text": "2. 4 повзунки = 4 сервоприводи (0–180 градусів)",
"TextColor": "&HFFFF0000",
"Uuid": "-1466203638"
},
{
"$Name": "Label17",
"$Type": "Label",
"$Version": "5",
"FontBold": "True",
```

```

    "FontSize": "12.0",
    "Text": "3. Натисни на мікрофон, щоб почати говорити команди",
    "TextColor": "&HFFFF0000",
    "Uuid": "1745577763"
  },
  {
    "$Name": "Label19",
    "$Type": "Label",
    "$Version": "5",
    "FontBold": "True",
    "FontSize": "12.0",
    "Text": "4. Додаток оснащений автоматизованим процесом",
    "TextColor": "&HFFFF0000",
    "Uuid": "-1541256205"
  }
],
"$blocks": [
  "<xml xmlns=\"http://www.w3.org/1999/xhtml\">
    <block type=\"component_event\" id=\"AU!Hwb[zU)n{:08?A=O6\" x=\"-
361\" y=\"-40\">
      <mutation          component_type=\"Button\"          is_generic=\"false\"
instance_name=\"RESET\" event_name=\"Click\"/>
      <field name=\"COMPONENT_SELECTOR\">RESET</field>
      <statement name=\"DO\">
        <block type=\"component_method\" id=\"DEd=|BwV549%:OnEjsvq\">
          <mutation          component_type=\"BluetoothClient\"
method_name=\"SendText\"          is_generic=\"false\"
instance_name=\"BluetoothClient1\"/>
          <field
name=\"COMPONENT_SELECTOR\">BluetoothClient1</field>

```

```

<value name=\"ARG0\">
  <block type=\"text\" id=\"4Kh,5VP?D=F84n++#K8\">
    <field name=\"TEXT\">RESET</field>
  </block>
</value>
<next>
  <block
                                type=\"component_set_get\"
id=\"fKm^wd2`dbO~vCTz^e.C\">
    <mutation      component_type=\"Button\"      set_or_get=\"set\"
property_name=\"BackgroundColor\" is_generic=\"false\" instance_name=\"RUN\"/>
    <field name=\"COMPONENT_SELECTOR\">RUN</field>
    <field name=\"PROP\">BackgroundColor</field>
    <value name=\"VALUE\">
      <block type=\"color_red\" id=\"{y-:+}_j~+9b2aq?)i=d\">
        <field name=\"COLOR\">#33ff33</field>
      </block>
    </value>
    <next>
      <block
                                type=\"component_set_get\"
id=\"ny/c#|3DJizpYj7yw6BZ\">
        <mutation      component_type=\"Button\"      set_or_get=\"set\"
property_name=\"Text\" is_generic=\"false\" instance_name=\"RUN\"/>
        <field name=\"COMPONENT_SELECTOR\">RUN</field>
        <field name=\"PROP\">Text</field>
        <value name=\"VALUE\">
          <block type=\"text\" id=\"Ef6C[]xnz=N0Eu$qP=SY\">
            <field name=\"TEXT\">RUN</field>
          </block>
        </value>
        <next>

```

```

        <block                                type="component_set_get"
id="X`sdJjJe$C.+l~oH:r}u">
            <mutation    component_type="Label"    set_or_get="set"
property_name="Text" is_generic="false" instance_name="Positions"/>
            <field
name="COMPONENT_SELECTOR">Positions</field>
            <field name="PROP">Text</field>
            <value name="VALUE">
                <block                                type="math_number"
id="@s$4J]=LIwHn,4es_5Eq">
                    <field name="NUM">0</field>
                </block>
            </value>
        </block>
    </next>
</block>
</next>
</block>
</next>
</block>
</statement>
</block>
</xml>"
]
}

```