

Івано-Франківський національний технічний університет нафти і газу

Інститут інженерної механіки та робототехніки

Кафедра комп'ютеризованого машинобудування

Голіней Михайло Михайлович

(прізвище, ім'я, по батькові)

УДК 621.865:004.896

(індекс)

БАКАЛАВРСЬКА РОБОТА

Робот-павук підвищеної безвідмовності

(назва роботи)

Інженерія мехатронних систем

(назва освітньої програми)

131- Прикладна механіка

(шифр і назва спеціальності)

Робота містить результати власних досліджень, використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело:

Здобувач освітнього ступеня _____ Голіней М.М.

(підпис, ініціали та прізвище здобувача)

Науковий керівник _____ Копей В.Б. доктор техн. наук, професор

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

Допущено до захисту

Завідувач кафедри

Професор _____ Панчук В.Г

(посада) (підпис) (дата) (ініціали та прізвище)

Рецензент

(посада) (підпис) (дата) (ініціали та прізвище)

Івано-Франківськ
2024

Івано-Франківський національний технічний університет нафти і газу

(повне найменування закладу вищої освіти)

Інститут Інженерної механіки та робототехніки

Кафедра Комп'ютеризованого машинобудування

Освітній рівень Бакалавр

Спеціальність 131- Прикладна механіка

(шифр і назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри _____

« ____ » _____ 20__ року

**З А В Д А Н Н Я
НА БАКАЛАВРСЬКУ РОБОТУ СТУДЕНТОВІ**

Голінею Михайлу Михайловичу

(прізвище, ім'я, по батькові)

1. Тема роботи Робот-павук підвищеної безвідмовності

керівник роботи Копей В.Б. доктор техн. наук, професор _____,

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

затверджені наказом закладу вищої освіти від " ____ " _____ 20__ року № _____

2. Строк подання студентом роботи _____

3. Вихідні дані до роботи: література з питань проектування мобільних роботів, оглядові матеріали роботів-павуків, офіційна документація Python, Micropython, esp32.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) Огляд конструкцій мобільних роботів та роботів-павуків. Огляд засобів резервування надійності. Тривимірна параметрична модель робота. Розрахунок на міцність найбільш навантажених деталей. Принципова електрична схема. Клієнт-серверна програма керування роботом через Wi-Fi для середовища Micropython мікроконтролера esp32.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) _____
Тривимірні моделі робота і деталей, складальні креслення, принципова електрична схема, алгоритми керування механізмами робота (5 аркушів формату A1)

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1-3	Копей В.Б. доктор техн. наук, професор		

7. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів магістерської роботи	Термін виконання етапів роботи	Примітка
1	Видача завдання		
2	Збір матеріалу, підготовка оглядової частини		
3	Проектування конструкції робота-павука		
4	Розробка програми для керування роботом		
5	Симуляція і відлагодження програм робота на wokwi.com		
6	Оформлення пояснювальної записки і графічної частини		
7	Відгук керівника дипломного проекту		
8	Рецензування і перевірка на плагіат		
9	Захист дипломної роботи		

Студент _____
(підпис)Голіней М.М.
(прізвище та ініціали)Керівник роботи _____
(підпис)Копей В.Б.
(прізвище та ініціали)

Анотація

Бакалаврська робота на здобуття освітньо-кваліфікаційного рівня бакалавра на тему: «Робот-павук підвищеної безвідмовності».

Бакалаврська робота складається з пояснювальної записки на 52 аркушах формату А4 з додатками, містить 14 рисунків. Графічна частина проєкту містить 5 креслень формату А1.

Проаналізовано переваги різних платформ мобільних роботів та виконано огляд конструкцій роботів-гексаподів з квазіколесами і механізмами резервування надійності. Розроблено конструкцію робота-гексапода з шістьма робочими органами, кожен з яких має можливість автономної гарячої заміни та має механізми резервування надійності, які можна використати у випадку відмови двох сервоприводів. Параметричну модель робота розроблено за допомогою САПР SOLIDWORKS. Міцність храпових коліс розраховано в SOLIDWORKS Simulation. Створено функції керуючої програми для середовища Micropython мікроконтролера esp32 робота. Реалізовано можливість віддаленого виклику цих функцій через Wi-Fi.

Ключові слова: робот-павук, гексапод, мобільний робот, надійність, автономний ремонт, SOLIDWORKS, Python, Micropython.

Abstract

Bachelor's thesis for obtaining a bachelor's educational and qualification level on the topic: " Spider robot with increased reliability ".

The bachelor thesis consists of an explanatory note on 52 sheets of A4 format with attachments, contains 14 figures. The graphic part of the project contains 5 drawings in A1 format.

The advantages of various platforms of mobile robots were analyzed and a review of the designs of hexapod robots with quasi-wheels and reliability backup mechanisms was performed. A hexapod robot design has been developed with six actuators, each of which is autonomously hot-swappable and has fail-safe mechanisms that can be used in the event of failure of two servos. The parametric model of the robot was developed using SOLIDWORKS CAD. The strength of ratchet wheels is calculated in SOLIDWORKS Simulation. Control program functions for the Micropython environment of the esp32 robot microcontroller have been created. The possibility of remote calling of these functions via Wi-Fi has been implemented.

Keywords: spider robot, hexapod, mobile robot, reliability, autonomous repair, SOLIDWORKS, Python, Micropython.

ЗМІСТ

ВСТУП.....	7
1 ОГЛЯД КОНСТРУКЦІЙ МОБІЛЬНИХ РОБОТІВ	9
1.1 Аналіз переваг різних платформ мобільних роботів.....	9
1.2 Огляд конструкцій роботів-павуків.....	10
2 ПРОЕКТУВАННЯ РОБОТА.....	16
2.1 Технічне завдання	16
2.2 Параметрична модель робота у SOLIDWORKS	16
2.3 Параметрична модель робочого органу у SOLIDWORKS	19
2.4 Механізм автономної «гарячої» заміни та резервування надійності.....	25
2.5 Розрахунок храпового колеса на міцність	34
3 ЕЛЕКТРИЧНА СХЕМА ТА ПРОГРАМА КЕРУВАННЯ РОБОТОМ	37
3.1 Електрична схема	37
3.2 Функції для керування механізмами.....	37
3.3 Віддалене виконання Python-команд через Wi-Fi	41
ВИСНОВКИ.....	44
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	45
ДОДАТКИ.....	47

ВСТУП

Мобільні роботи стають все більш популярними у багатьох сферах. Завдяки цьому, вимоги до їх функціональності збільшуються. Сучасні вимоги до мобільних роботів вимагають підвищення їхньої ефективності для широкого спектру застосувань, включаючи промислову автоматизацію, дослідження космосу, роботу з ризикованими матеріалами, оборонні цілі, та демінування території.

Основною проблемою гексаподів є невисока надійність, що пов'язано зі збільшеною кількістю двигунів та деталей приводів, а також складними умовами роботи приводів, які включають високі статичні і ударні навантаження. Додатковою проблемою є те, що більшість недорогих сервоприводів sg90, які планується використовувати в цьому проекті володіють низькою надійністю. Роботи-гексаподи, які працюють віддалено, у першу чергу такі як марсоходи, повинні мати засоби забезпечення і багатократного резервування надійності. Як правило відмовлені деталі може замінити тільки людина, яка присутня біля робота. Або інший спеціалізований робот-ремонтник. Проте існує можливість розробити конструкції з автономною заміною відмовлених деталей. В конструкції потрібно передбачити не тільки засоби заміни вузлів на нові, але й додаткові засоби, які дозволятимуть функціонувати роботу до ремонту навіть з відмовленими сервоприводами.

Метою роботи є створення недорогого робота-гексапода з механізмами резервування надійності приводів та можливістю автономної заміни відмовлених вузлів. Для досягнення мети потрібно розв'язати наступні задачі:

1. Проаналізувати переваги і недоліки різних платформ мобільних роботів – колісної, гусеничної, крокуючої та комбінованої. Проаналізувати способи резервування надійності в роботах-гексаподах.

2. Розробити конструкцію робота-гексапода, який має можливість автономної «гарячої» заміни вузлів, що відмовили, та механізми резервування надійності сервоприводів.

3. Розробити тривимірну параметричну модель цього робота в SOLIDWORKS та виконати оцінку міцності найбільш навантажених деталей.

4. Розробити принципову електричну схему на основі мікроконтролера esp32 та створити керуючу програму для середовища Micropython з можливістю віддаленого виконання Python-команд через Wi-Fi.

1 ОГЛЯД КОНСТРУКЦІЙ МОБІЛЬНИХ РОБОТІВ

1.1 Аналіз переваг різних платформ мобільних роботів

Мобільні роботи можуть бути побудовані на основі різних платформ: колісних, гусеничних, крокуючих або комбінованих. Класичні мобільні роботи на колісній платформі володіють високою швидкістю переміщення, надійністю і простотою конструкції [1] (рис. 1.1). Проте вони мають низьку маневреність та обмежену прохідність [2]. Для підвищення маневреності застосовують платформи на основі шведських коліс (всенаправлених коліс) [3]. Гусенична платформа застосовується для підвищення зчеплення з поверхнею. Вона забезпечує достатню швидкість переміщення робота, високу прохідність і вантажопідйомність [4]. Крокуюча платформа володіє найвищою гнучкістю і прохідністю [5]. Робот може не тільки ходити, але й стрибати, захоплювати предмети і робити інші дії, які неможливі для інших платформ. Проте такі роботи володіють складною конструкцією і обмеженою надійністю. Також складно запрограмувати алгоритм керування таким роботом. Комбіновані платформи [6, 7] можуть поєднувати вказані переваги різних платформ. В цій роботі поставлено завдання розробити крокуючого робота-гексапода, який має шість робочих органів (ніг) і володіє вбудованими засобами підвищення його надійності.



Рисунок 1.1 – Переваги різних платформ мобільних роботів

Традиційні шестиногі роботи, відомі як «hexapods», оснащені шістьма ногами і можуть переміщатися лише ходьбою. Кожна нога має три сервоприводи: один для обертання навколо вертикальної осі і два для обертання ланок навколо горизонтальних осей. Використовується спеціальний алгоритм для координації руху ланок, що забезпечує високу прохідність робота, хоча його швидкість не може конкурувати з колісними роботами [6, 7]. Додаткові сервоприводи збільшують вагу та вартість робота. Роботи-павуки із шістьма ногами мають ряд переваг і мають низку застосувань. Переваги:

1.Стійкість. Шість ніжок забезпечують більшу стійкість на нерівній місцевості.

2.Маневреність. Вони можуть легко повертатися і рухатися в будь-якому напрямку.

3.Прохідність. Здатність переступати через перешкоди, а не обходити їх.

Області їхнього застосування:

1.Пошук і порятунок. Може переміщатися через завали або небезпечне середовище.

2.Дослідження космосу. Ідеально підходить для дослідження нерівних поверхонь планет.

3.Сільське господарство. Використовується для моніторингу врожаю та аналізу ґрунту.

1.2 Огляд конструкцій роботів-павуків

Ось кілька прикладів шестиногих роботів:

Робот RiSE [8] розроблений у Карнегі-Меллон, це робот, натхненний комахами, з боковою симетрією, призначений для лазіння по вертикальних поверхнях.

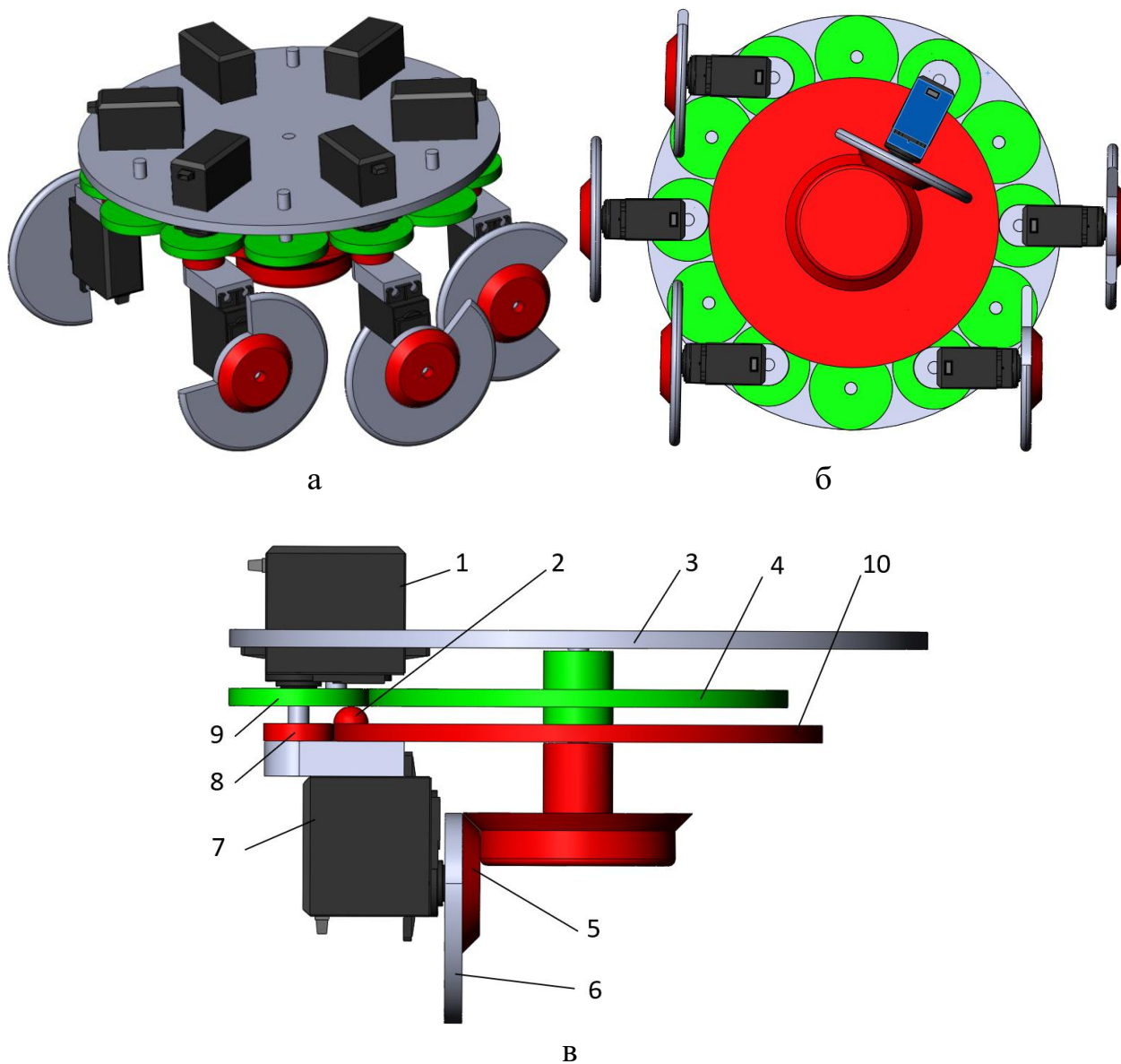
ATHLETE [9] (All-Terrain Hex-Legged Extra-Terrestrial Explorer) (всюдихідний шестиногий позаземний дослідник). Це радіально-симетричний гексапод, розроблений у JPL для дослідження космосу.

Ці роботи демонструють різноманітність дизайну та застосування шестиногих роботів.

В праці [6] запропонована конструкція мобільного робота з квазіколесами та системою резервного забезпечення надійності. Розробники [6] поставили перед собою завдання створити мобільного робота, який би відповідав високим стандартам гнучкості, проходимості, автономності, надійності та доступності. Вони склали матрицю морфологічного аналізу, що включає такі параметри як тип колеса, конструкція ніг, захоплення, трансмісія, приводи, кількість та ступені свободи робочих органів. Аналіз цієї матриці сприяв створенню декількох інноваційних концепцій, зокрема, інтригуючою була ідея поєднання колеса, ноги, маніпулятора та підвищеної надійності [6].

Відомо, що роботи-гексаподи мають високу прохідність, але обмежені у швидкості, тоді як колісні роботи швидкі, але менш прохідні. Часто використовуються комбінації коліс та ніг, як у марсоходів, або квазіколеса, що забезпечують і швидкість, і прохідність. Розробники [6] вирішили об'єднати переваги колеса, ноги та маніпулятора в єдиній конструкції, яка також включає систему резервного забезпечення надійності з п'ятикратним резервуванням та можливістю спільної роботи основних та резервних елементів.

Робот складається (рис. 1.2) з шести однакових робочих органів, кожен з яких містить 2 сервоприводи MG996R з можливістю обертання їхнього валу на 360 градусів [6]. Квазіколесо може обертатися навколо вертикальної та горизонтальної осі завдяки сервоприводам. У режимі переміщення колеса орієнтуються паралельно напрямку руху та обертаються для швидкого пересування та подолання перешкод [6]. Два робочі органи можуть виконувати функції маніпулятора для захоплення предметів. У режимі резервування, при відмові деяких сервоприводів, робочий орган може переключатися для створення зачеплення конічної передачі, дозволяючи обертати робочий орган з несправним сервоприводом [6]. Шестерні можуть передавати рух на додаткові механізми, такі як додатковий маніпулятор. Муфта забезпечує перемикання передачі між сервоприводом та шестернею, активуючись лише при обертанні диску в одному напрямку [6].



1, 7 – сервоприводи; 3 – корпус; 4, 9 – циліндрична передача з постійним колесом 4 і змінною шестернею 9; 5 – конічна передача з постійним колесом і змінною шестернею; 10 – диск з храповиком і кулачком 2; 8 - муфта

Рисунок 1.2 – SOLIDWORKS-модель робота [6]

а – вид збоку, б – вид знизу, в – робочий орган в режимі резервування

Колеса робота можуть мати різні кути вирізу, що розширює його функціональні можливості [6]. Технологічна конструкція робота дозволяє виготовляти більшість деталей за допомогою лазерного різання або 3D-друку, що знижує вартість виробництва [6].

В праці [7] було запропоновано розширити функціональність робота-павука. Автори проекту [7] мали за мету скоротити кількість сервоприводів, але при цьому

розширити можливості робота, додавши колеса. Робот складається з шести ідентичних модулів (див. Рисунок 1.3) і має тільки 12 сервоприводів MG996R, кожен з яких може обертати свій вал на 360 градусів. Сервопривід 5 прикріплений до основи робота і через муфту 4 приводить в рух ланку 2 з маленькими колесами 1 і 6, які можна використовувати за бажанням [7]. Сервопривід 3, прикріплений до ланки 2, через муфту 7 приводить в рух ланку 8 з колісною базою 10 і маленькими колесами 9 і 11 [7].

Робот може переміщатися в двох режимах: ходьбою та коченням. У режимі ходьби сервопривід 3 відштовхує робота від підлоги за допомогою ланки 8, яка може мати гумову «підшву» [7]. Сервопривід 5 піднімає ланку 8 над підлогою. У режимі кочення ланки 8 декількох модулів розташовуються паралельно, забезпечуючи контакт коліс з підлогою, а інші модулі відштовхують робота від підлоги [7]. Колеса 1 і 6 також можуть використовуватися для кочення, одне з яких може бути гумовою «підшвою» для розширення функціональності. Робот також може захоплювати та переносити предмети, а його конструкція дозволяє продовжити роботу навіть після випадкового перевертання на іншу сторону [7].

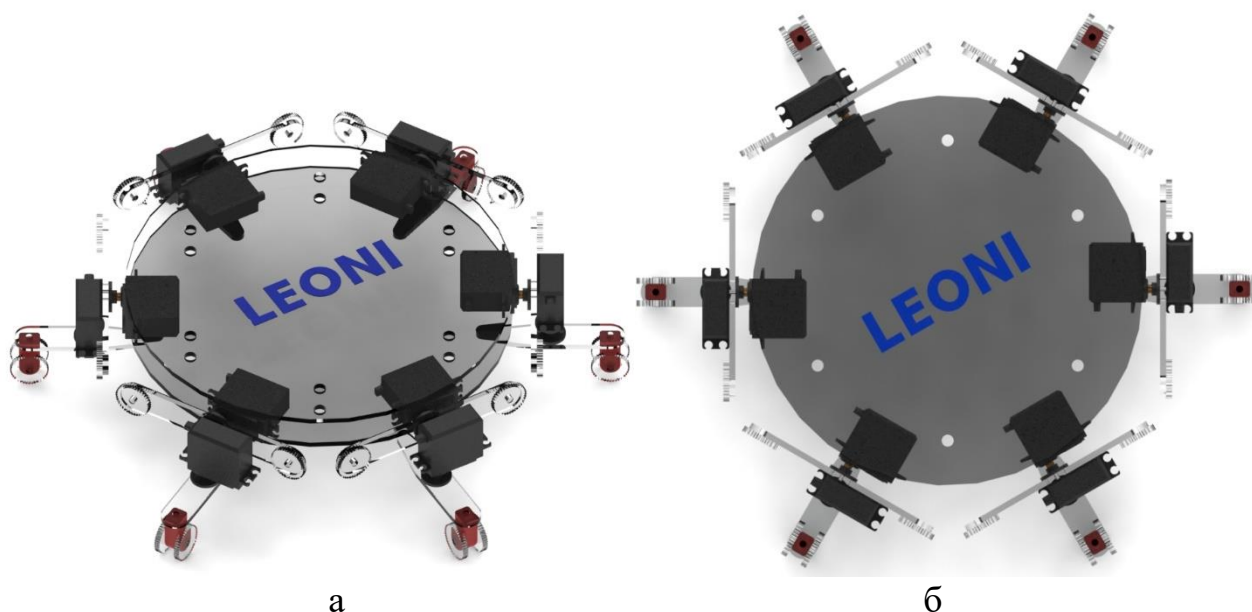


Рисунок 1.3 – SOLIDWORKS модель робота [7]

сформувати нову концепцію такого робота з додатковою можливістю автономної заміни відмовлених вузлів.

2 ПРОЕКТУВАННЯ РОБОТА

2.1 Технічне завдання

Робот-гексапод повинен бути призначений для роботи в важкодоступних місцях, володіти механізмом резервування надійності та мати можливість автономної гарячої заміни робочих органів.

Керувати роботом повинен мікроконтролер esp32 [10] з прошивкою MicroPython [11, 12]. Мікроконтролер повинен підтримувати віддалене керування через wi-fi або Bluetooth. Для живлення використовується 2 літєві батареї 18650. Проте кількість батарей може бути збільшена для покращення автономності.

Робот повинен містити шість робочих органів, які підтримують автономну гарячу заміну без участі людини. Повинна бути можливість замість робочого органу використовувати додаткові інструменти для виконання роботом певних спеціальних завдань (наприклад захват предметів, буріння, взяття проб ґрунту тощо).

2.2 Параметрична модель робота у SOLIDWORKS

Параметрична модель робота-павука розроблено у SOLIDWORKS 2020. SOLIDWORKS – це сучасний і зручний САПР для автоматизації проектних робіт на етапах конструкторської та технологічної підготовки виробництва [13, 14].

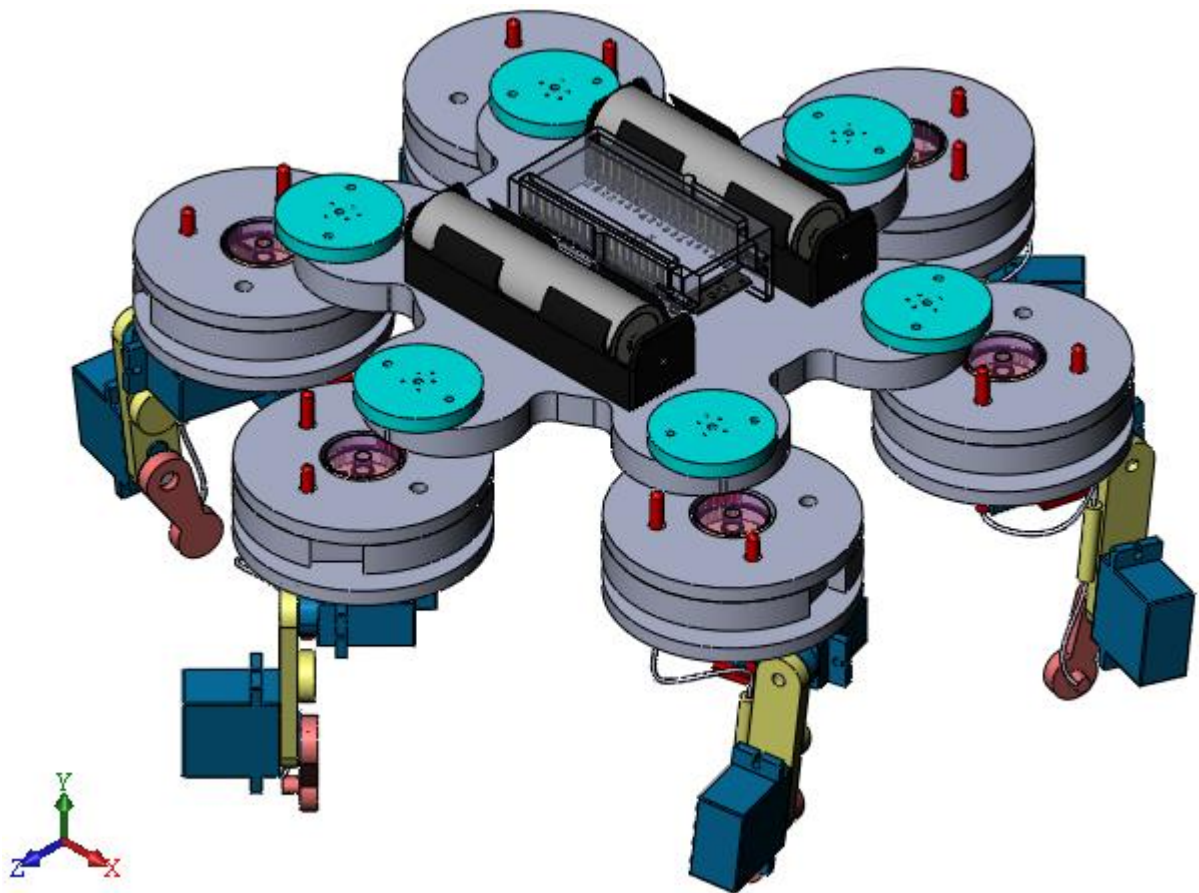
На рисунку 2.1 показано тривимірну модель запропонованого робота.

Складальне креслення робота-павука наведено в додатках. Специфікація (рис. 2.2):

- 1 - Робочий орган в зборі, 6 шт.
- 2 - Плита основи (акрил).
- 3 - Батарея 18650 з тримачем, 2 шт. Модель завантажили з Grabcad [15, 16].
- 4 - Роз'єм з магнітом, 6 шт. (акрил, неодимовий магніт).
- 5 - esp32. Модель завантажили з Grabcad [17].

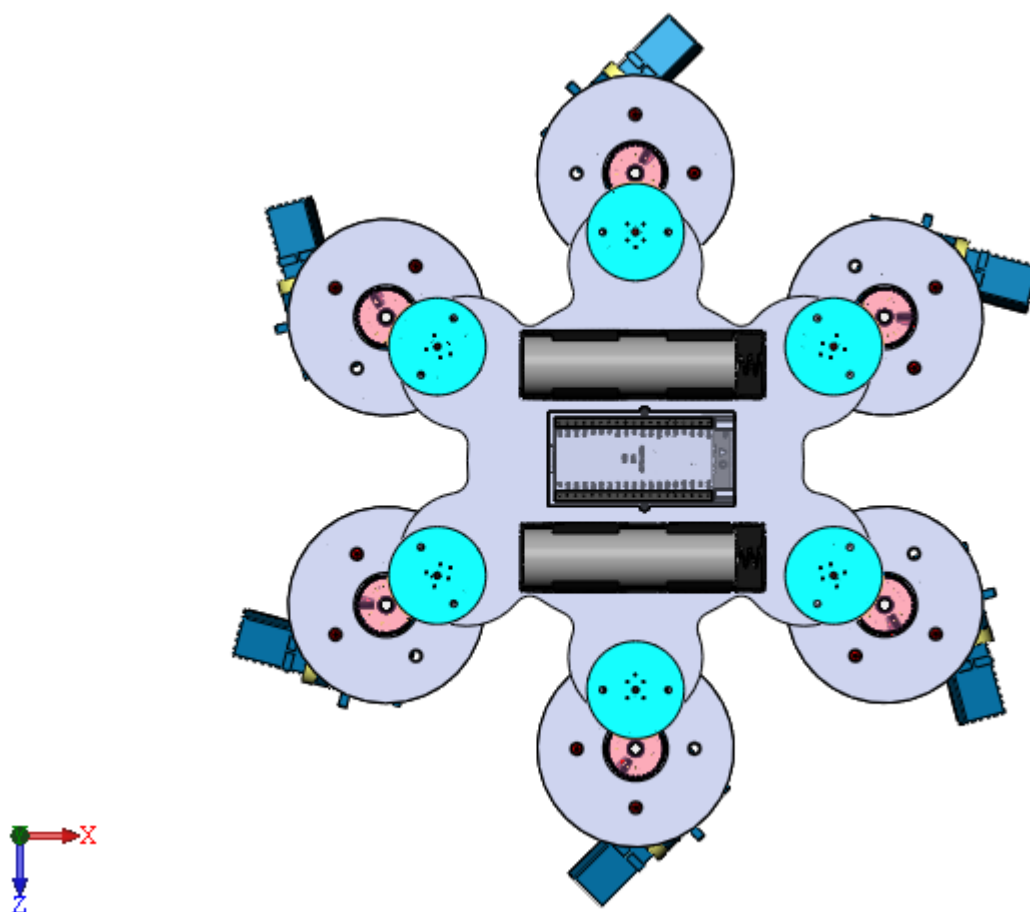
- ▶ (ф) RobotBody<1> (По умолч
- ▶ Connector<1> -> (По умолча
- ▶ (-) main_assy<1> (Default<Di
- ▶ (-) 18650 Battery Holder v5.st
- ▶ Asm<1> (По умолчанию<По
- ▶ Группа сопряжений1
- ▶ Местный линейный массив1
- ▶ Местный круговой массив1
- ▶ Местный круговой массив2

а



б

Рисунок 2.1 – Тривимірна модель робота-павука



В

Продовження рисунка 2.1

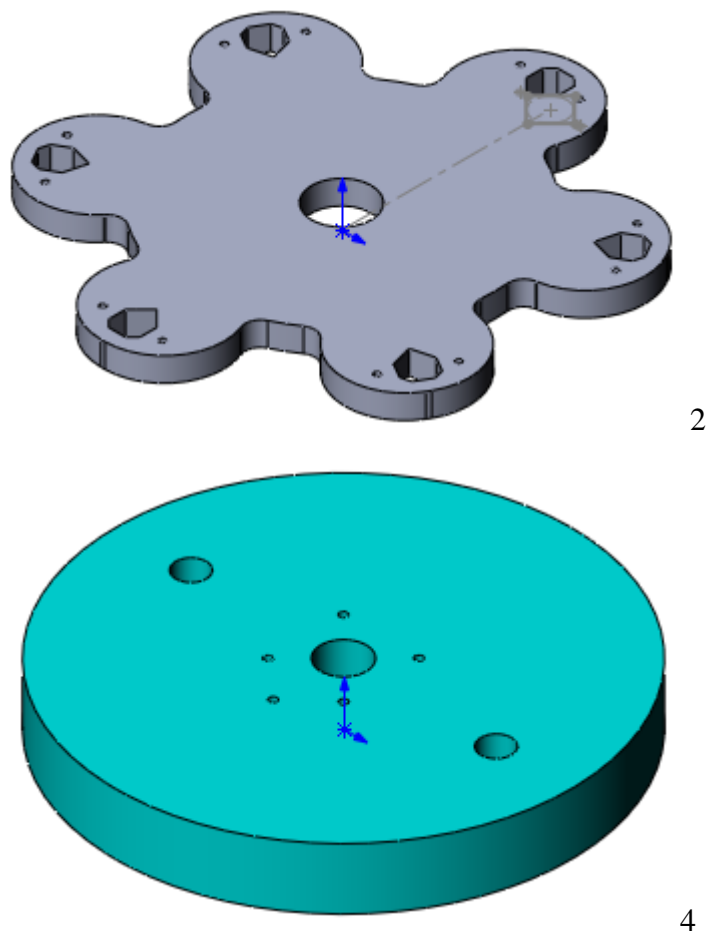
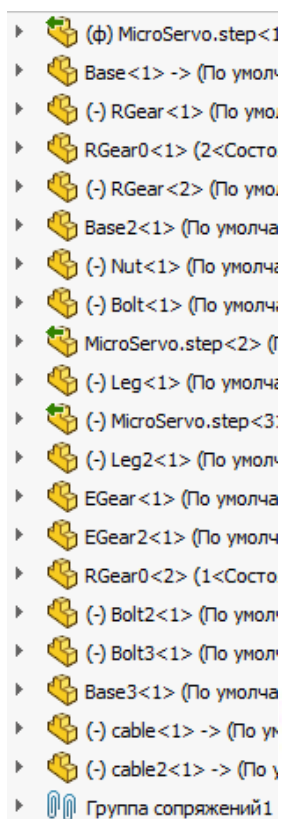


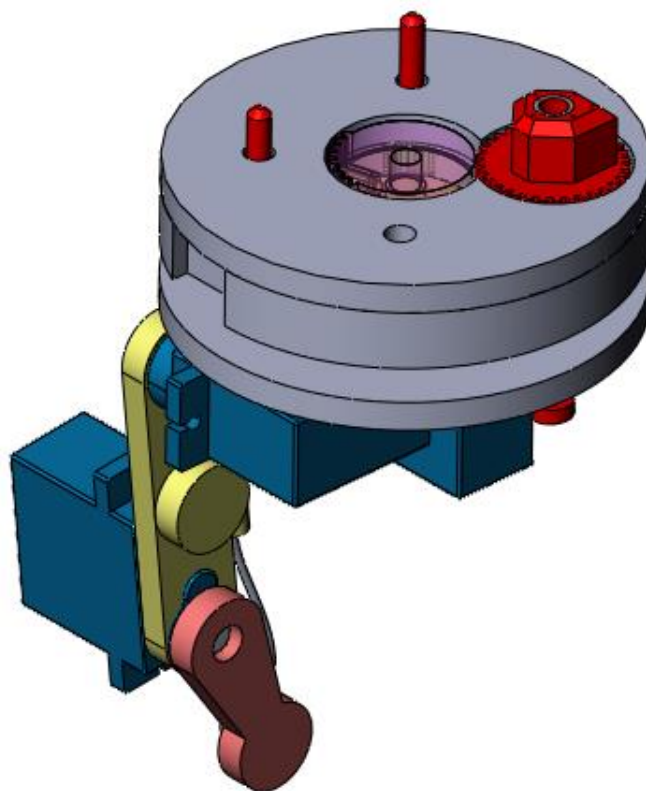
Рисунок 2.2 – Деталі базової складальної одиниці робота

2.3 Параметрична модель робочого органу у SOLIDWORKS

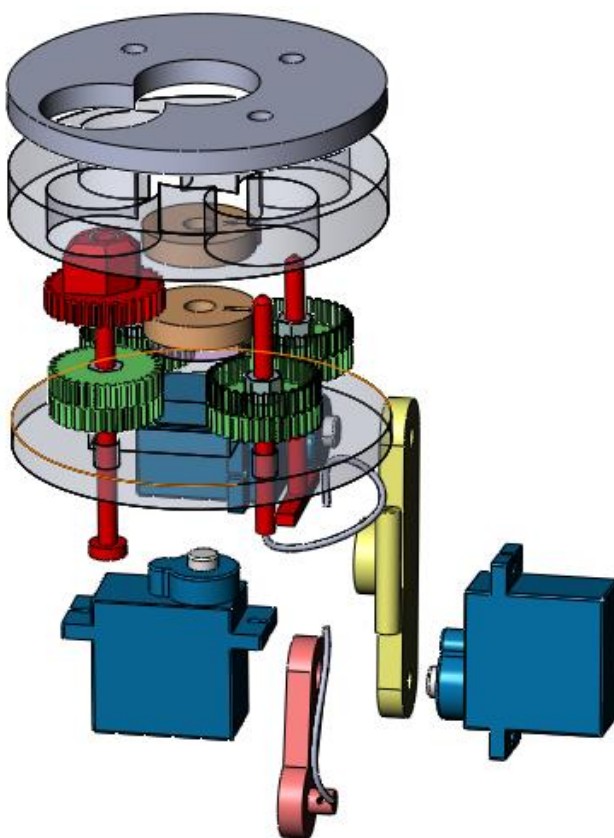
Кожний робочий орган має три сервоприводи sg90. Перший з них забезпечує обертання робочого органу навколо вертикальної осі та передачу руху на храпові механізми. Інші два сервоприводи обертають ланки ніг навколо горизонтальних осей для можливості ходьби. З використанням цих сервоприводів робот може пересуватись по доволі складній місцевості. Це важливо для роботи у важкодоступних місцях. На рисунку 2.3 показано загальний вигляд робочого органу. Червоним кольором показано роз'єм з магнітом, до якого повинен під'єднатись робот. Рисунок 2.4 зображує детально механізм робочого органу.



а



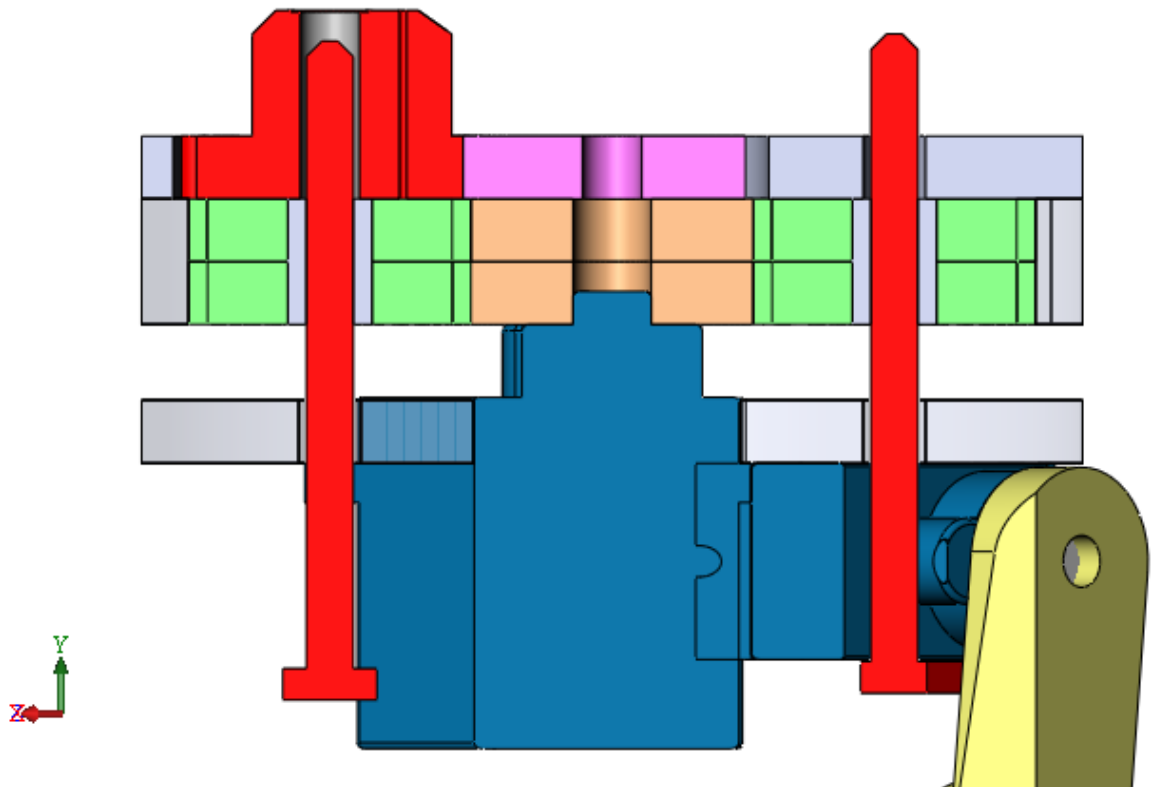
б



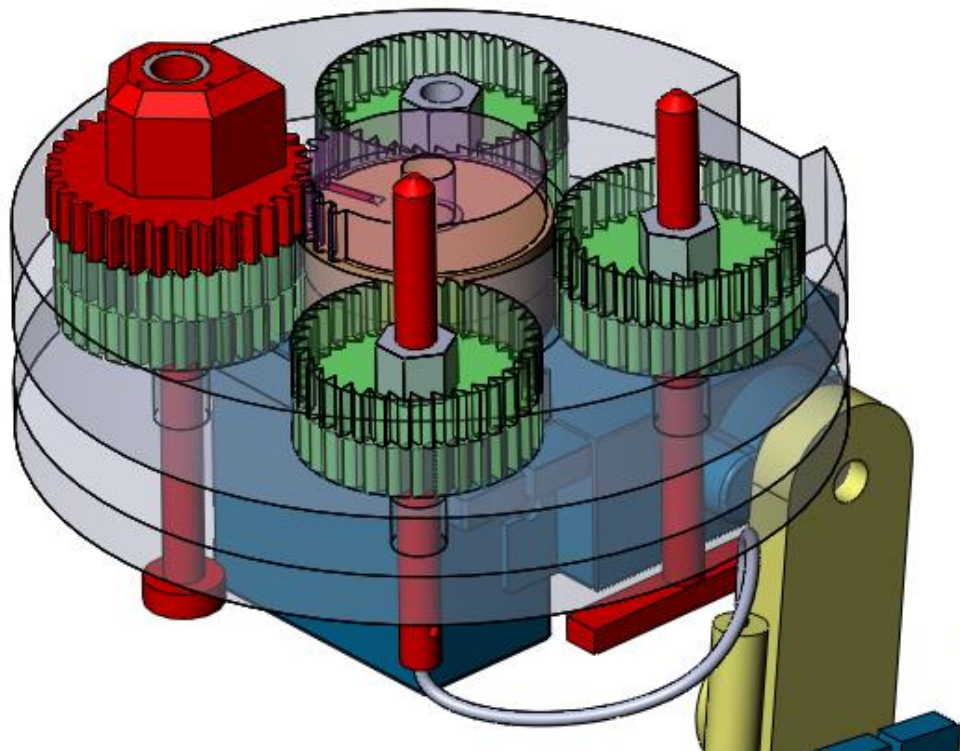
в

а – дерево побудови, б – зібраний, в – розібраний

Рисунок 2.3 – Модель робочого органу робота



а



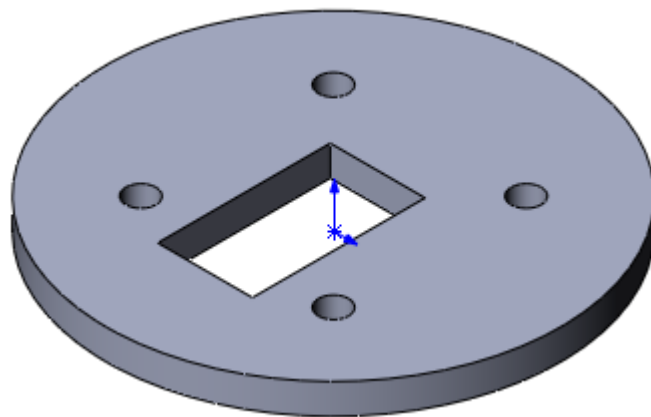
б

а – осьовий перетин, б – прозорий корпус

Рисунок 2.4 – Механізм робочого органу, який забезпечує його обертання навколо вертикальної осі, автономну гарячу заміну та резервування надійності

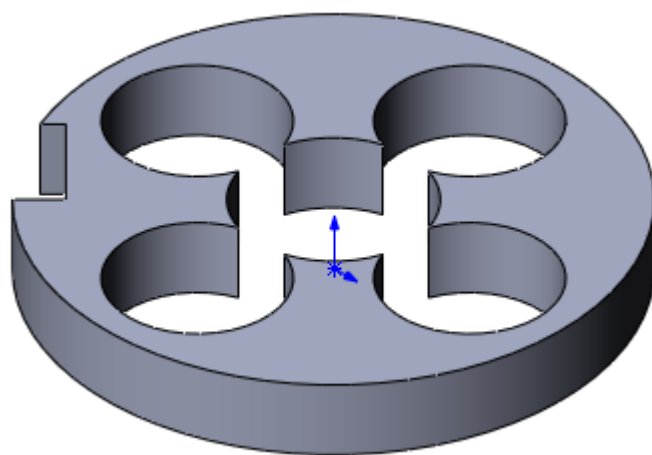
Складальне креслення робочого органу наведено в додатках. Специфікація (рис. 2.5):

- 1 - сервопривід sg90, 3 шт. Завантажено з Grabcad [18].
- 2 - боуден-трос [19].
- 3 - плита нижня (акрил).
- 4 - плита середня (акрил).
- 5 - плита верхня (акрил).
- 6 - ланка верхня (акрил).
- 7 - ланка нижня (акрил).
- 8 - секторне колесо (акрил).
- 9 - колесо-штекер (нейлон).
- 10 – храповик, 2 шт. (акрил, сталь).
- 11 - храпове колесо, 8 шт. (акрил).
- 12 – гайка, 4 шт. (сталь).
- 13 - гвинт-кулачок (сталь).
- 14 - гвинт штекера (сталь).
- 15 - гвинт боуден-тросу (сталь).

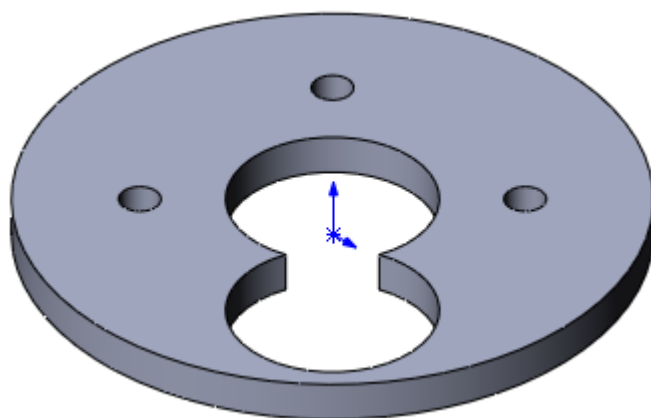


3

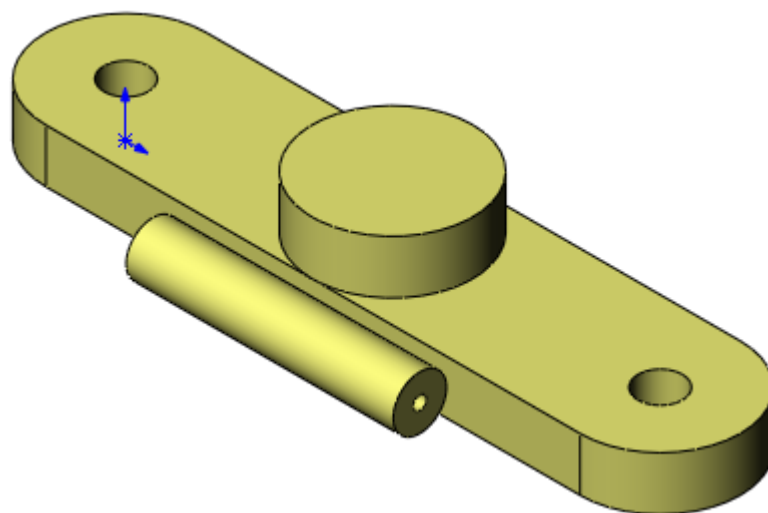
Рисунок 2.5 – Деталі робочого органу



4

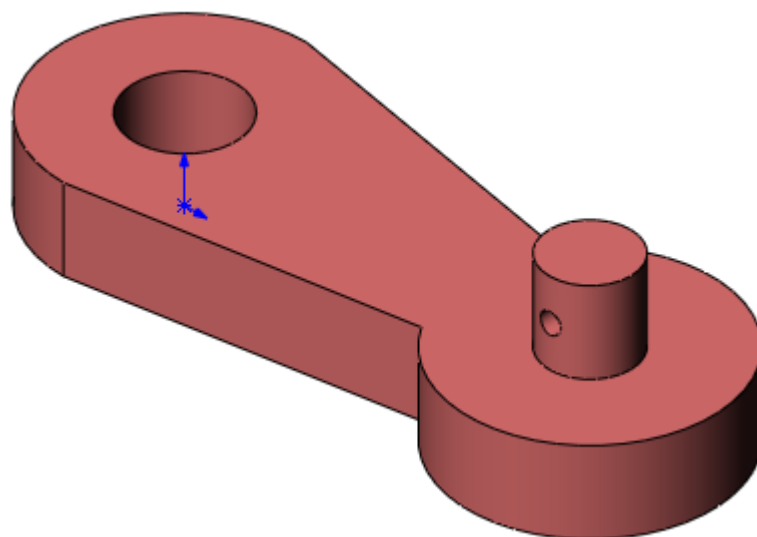


5

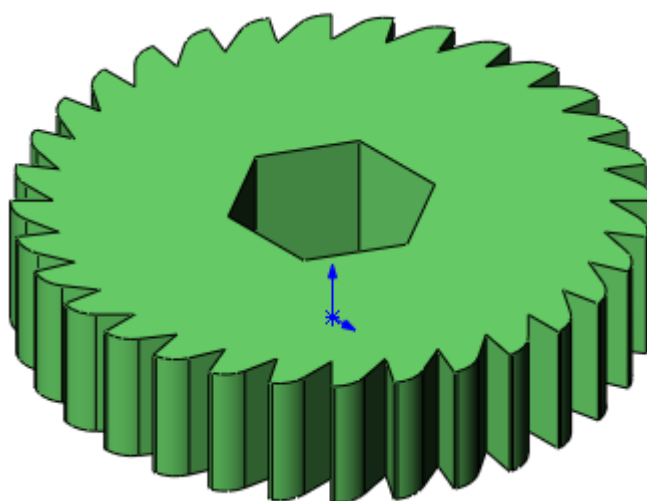


6

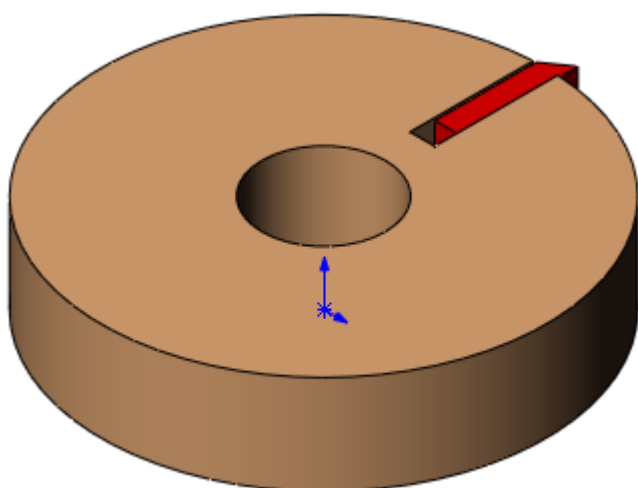
Продовження рисунку



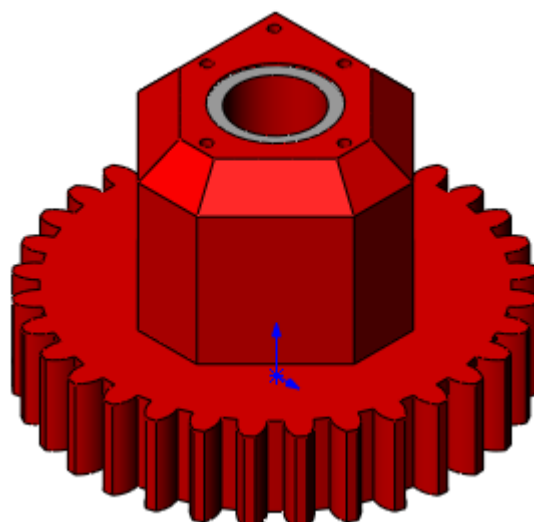
7



11

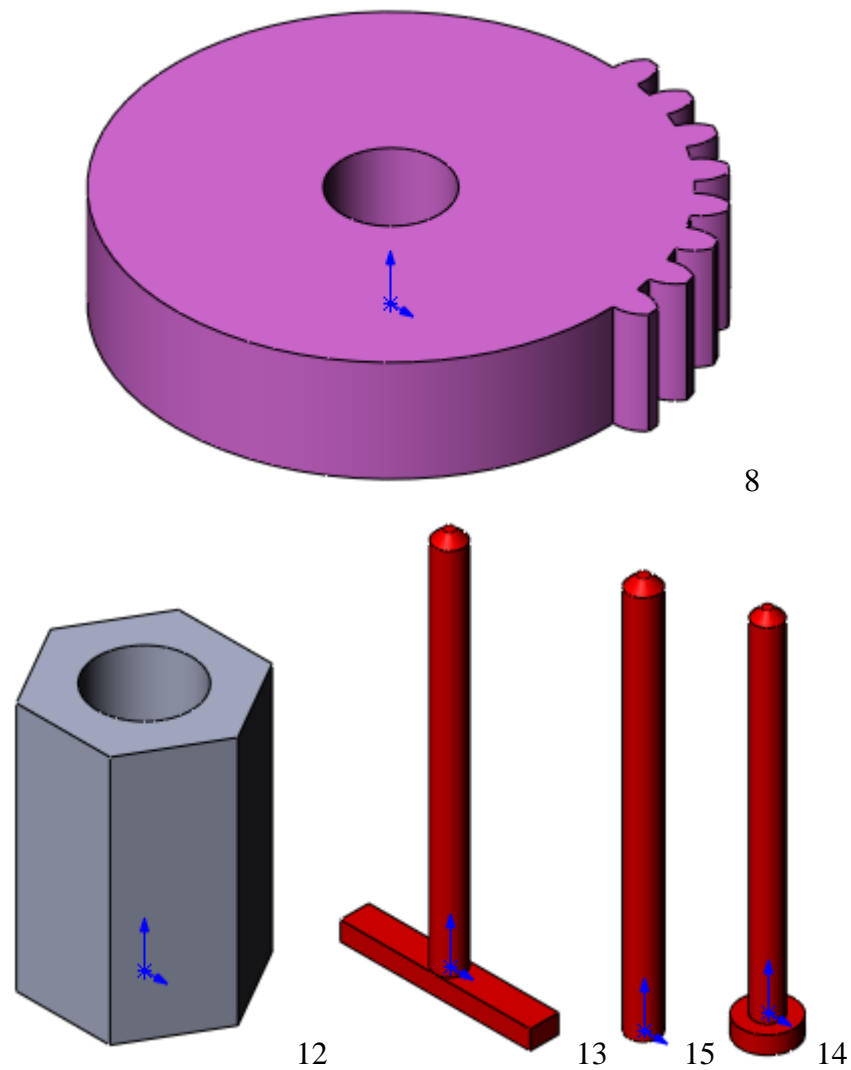


10



9

Продовження рисунку



Продовження рисунку.

2.4 Механізм автономної «гарячої» заміни та резервування надійності

Робот має можливість гарячої заміни робочих органів без участі людини (рис. 2.6). Для цього запасні робочі органи та додаткові інструменти повинні бути розташовані на спеціальній стаціонарній базі, яка може бути обладнана мітками для полегшення їхньої ідентифікації роботом. Робот пересувається до цієї бази, за допомогою храпового механізму і першого сервоприводу відгвинчує гвинт роз'єму і відділяє неробочий орган. Після цього він пересувається до нового робочого органу на базі, під'єднується до роз'єму за допомогою магніту, загвинчує гвинт роз'єму і робот може працювати далі.

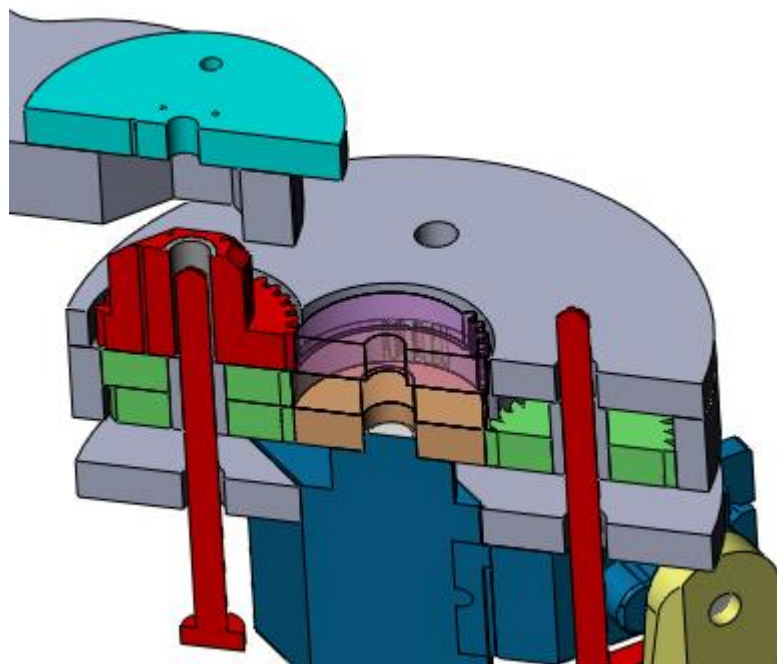
Звичайно, для такої операції робот повинен бути обладнаний сенсорами для визначення положення запасного робочого органу. В якості такого сенсору може

використовуватись відеокамера. Спеціальні алгоритми машинного зору визначають положення робочого органу і робот пересувається до нього для операції заміни. Також для цього можна використовувати такі сенсори як датчики магнітного поля або оптичні датчики, або навіть звичайні датчики контакту. Заміну робочого органу також може виконувати не алгоритм, а людина-оператор, яка віддалено управляє роботом по радіоканалу.

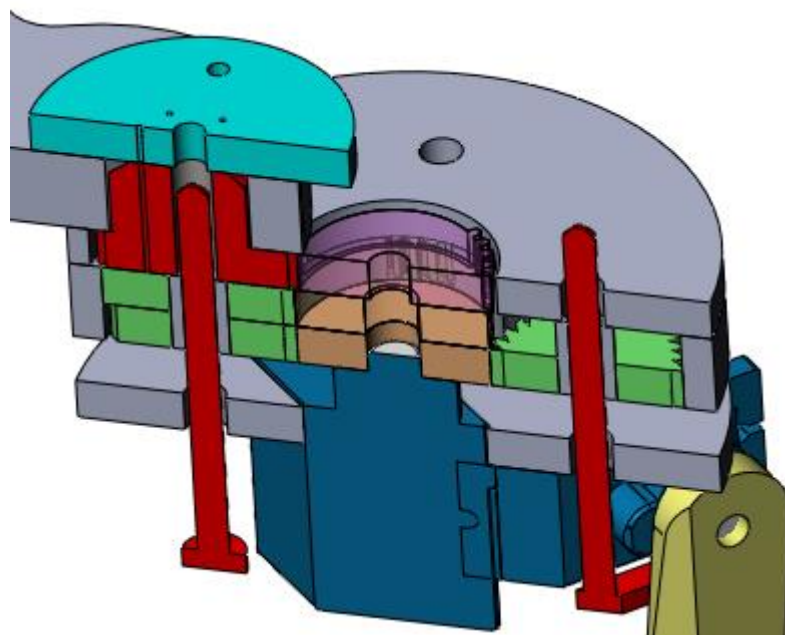
На рисунку 2.6 показано етапи гарячої заміни робочого органу. На першому рисунку робочий орган ще не під'єднаний, але наявність магніту зверху контактного роз'єму дозволяє забезпечити під'єднання електричних контактів, яких є п'ять штук: земля, живлення сервоприводів (5 В) і три контакти для цифрового керування сервоприводами. Як тільки контакт відбувся починає працювати перший сервопривід, який обертає вертикальну вісь з двома храповиками і секторним евольвентним колесом. Цей сервопривід повинен мати можливість обертатись на 360° . Це не проблема знайти таку модель сервоприводу sg90. Сервопривід повертає храповик на заданий кут, після чого здійснює обертання храповика вперед ($+10^\circ$) і назад (-10°), тобто обертаючись в потрібному інтервалі кутів (див. циклограму на рис.). Таким чином храпове колесо обертає гайку, яка загвинчує гвинт в роз'єм. В результаті робочий орган буде надійно закріплений і робот може працювати далі.

Якщо потрібно відгвинтити цей гвинт, то храповик переміщує інше парне храпове колесо завдяки тому, що він може обертатись в іншому інтервалі кутів (див. циклограму на рис. 2.7, 2.8). Робочі інтервали кутів храповиків і секторного колеса не перетинаються. Роз'єм має спеціальну форму, яка забезпечує під'єднання контактів завжди правильно. Помилка неможлива. Для ідентифікації розташування роз'єму може використовуватись магніт і датчик ефекту Холла [20].

На рисунках 2.7 показані робочі зони та циклограма роботи храповиків та секторного колеса. Видно, що в межах кутів від 31° до 51° працює храповик 1R, робочим рухом якого є обертанням праворуч, а холостим – обертання ліворуч. Такий рух забезпечує обертання храпових коліс 1 з гайкою в одну сторону. Це призводить до руху гвинта в одну сторону.

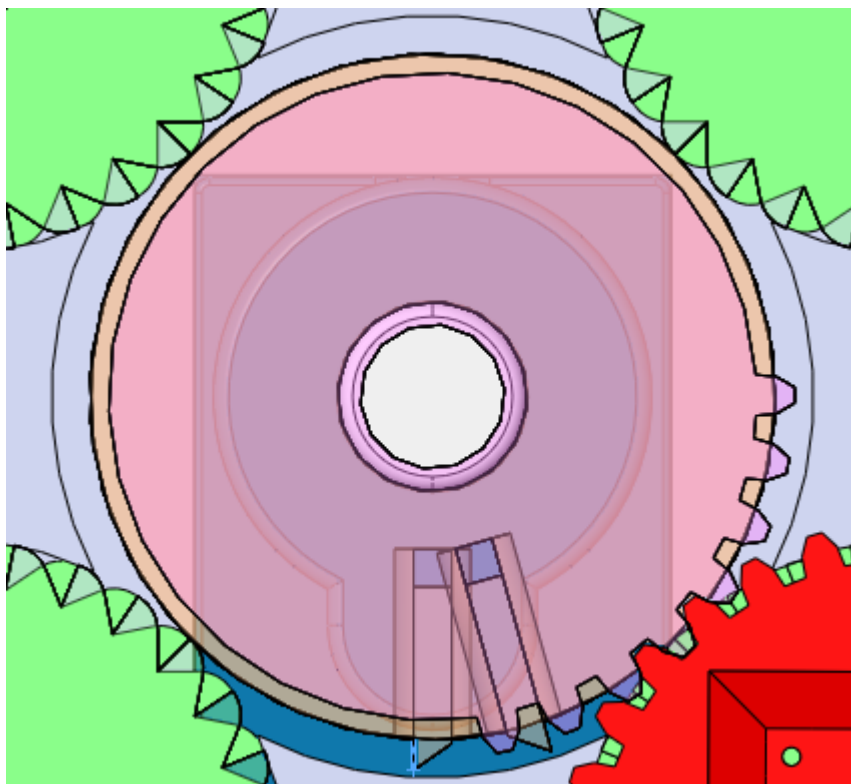


а

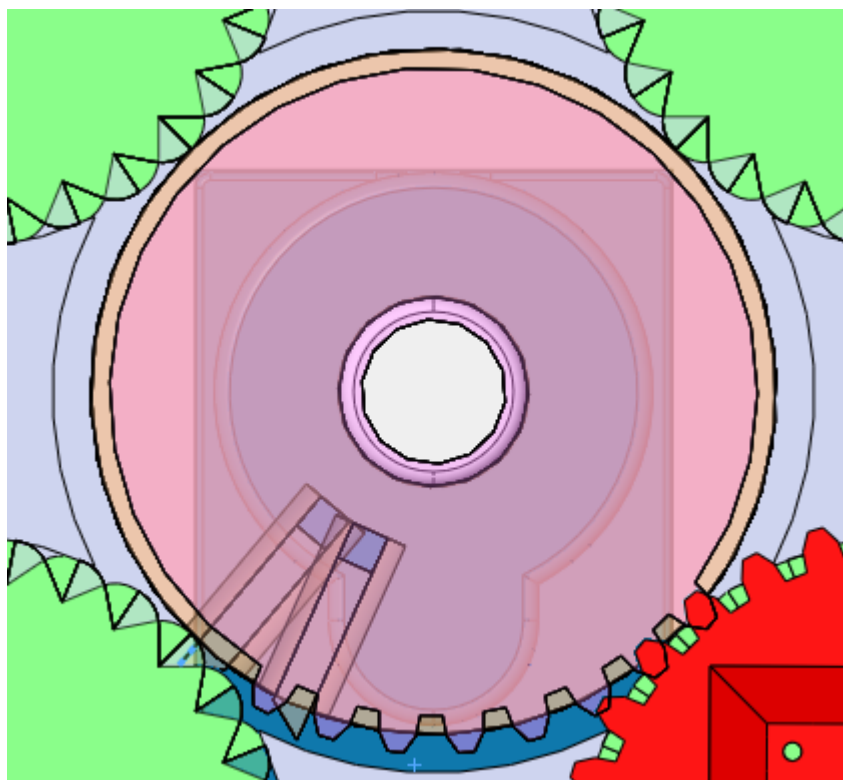


б

Рисунок 2.6 – Етапи гарячої заміни робочого органу: а – від'єднано, б – приєднано
(але ще не загвинчено)

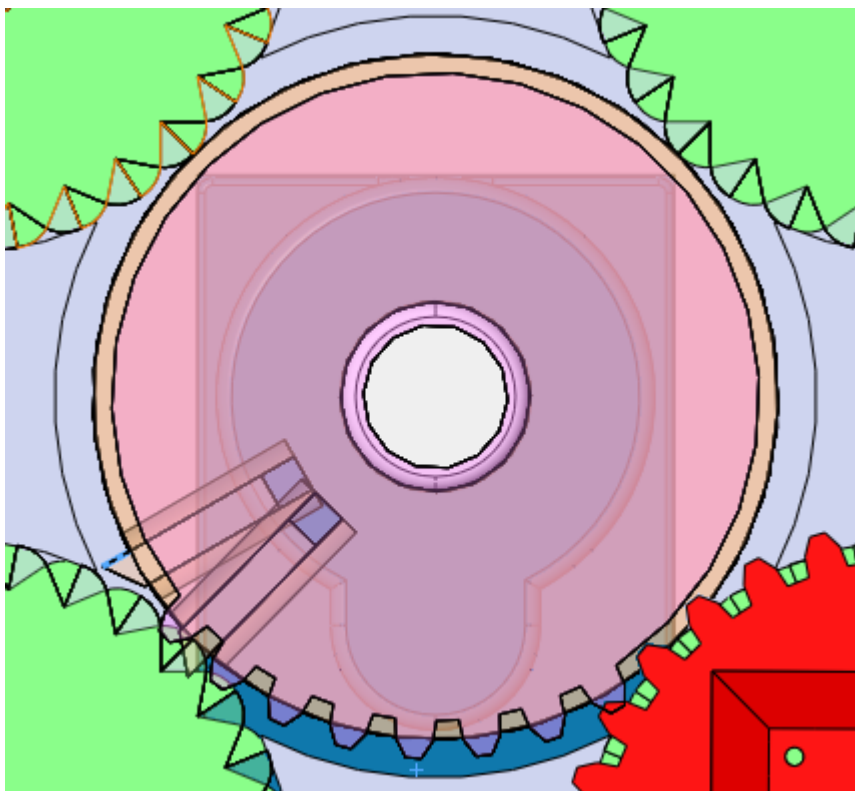


а - 0°

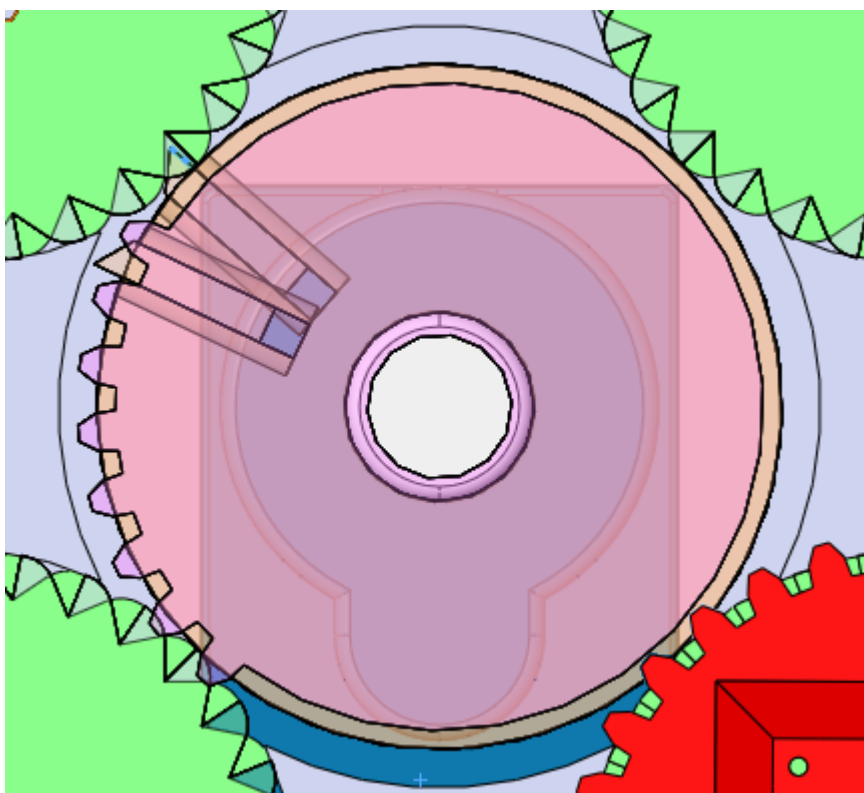


б - 41° (31°-51°)

Рисунок 2.7 – Кути робочих зон першого сервопривода: центр (мінімум-максимум)

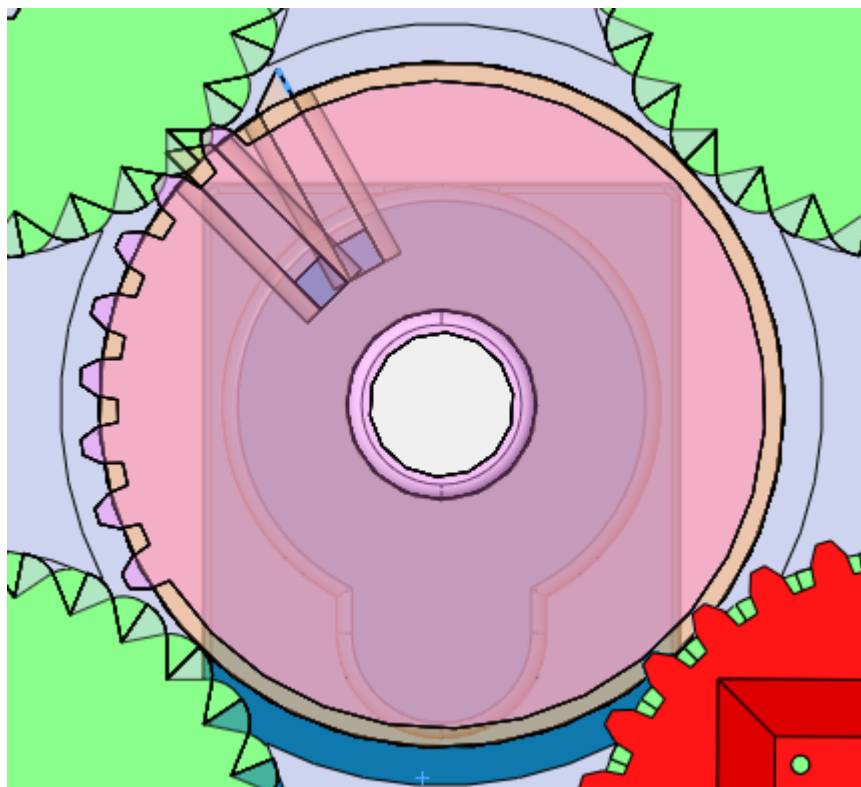


В - 61° (51° - 71°)

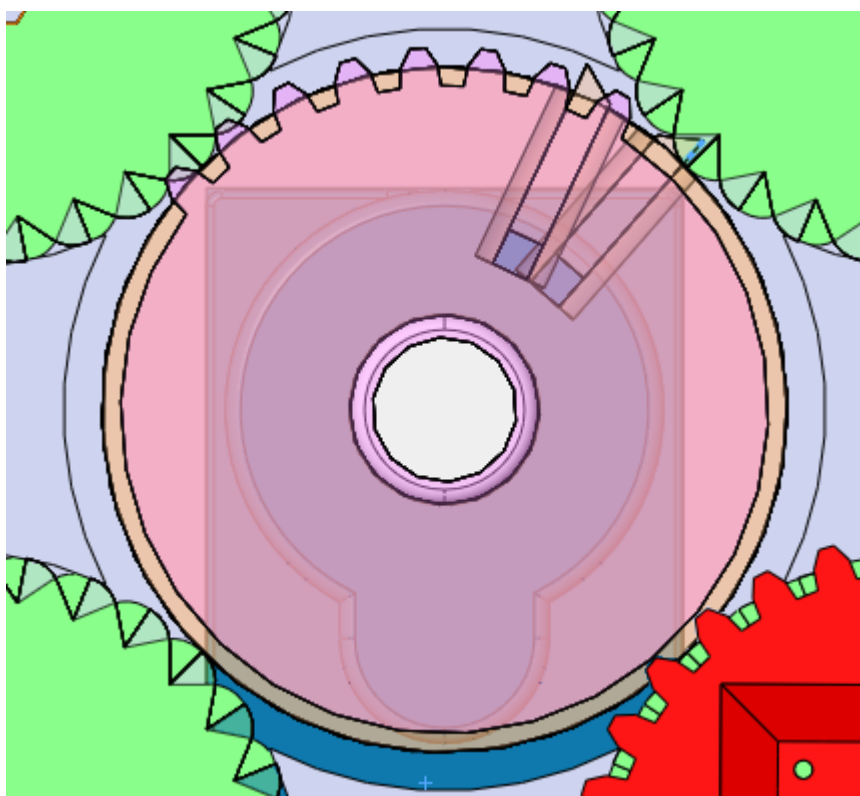


Г - 131° (121° - 141°)

Продовження рисунку.

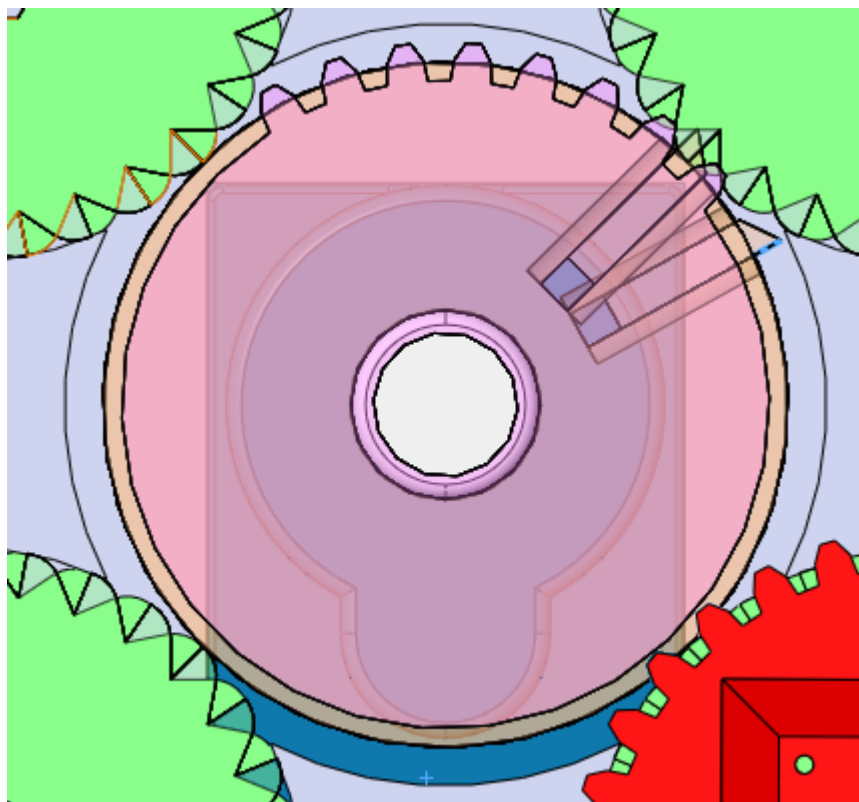


д - 151° (141° - 161°)

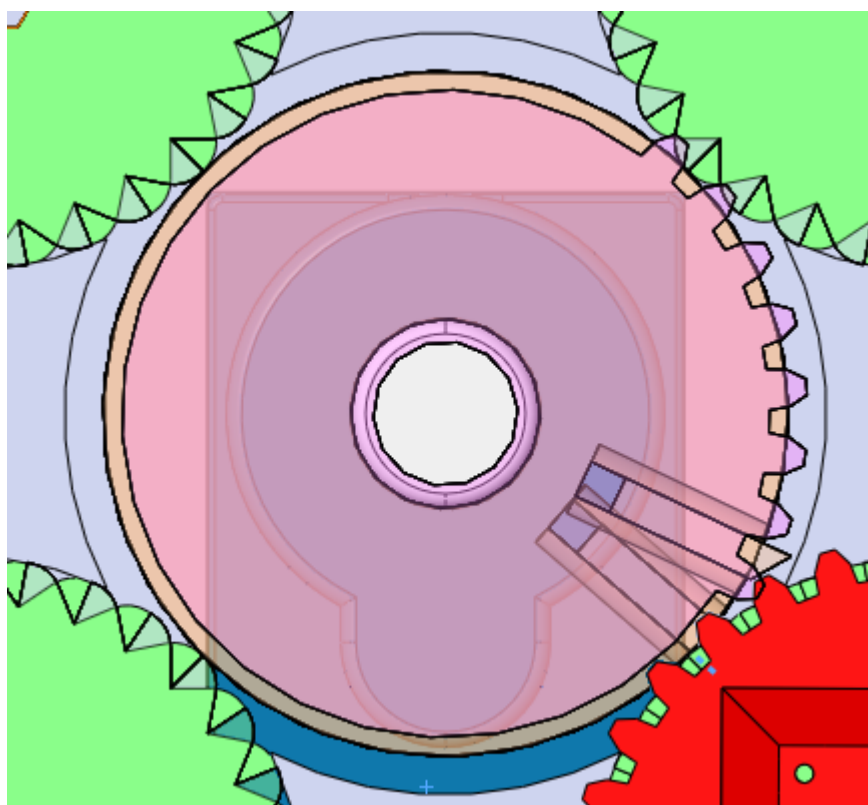


е - 221° (211° - 231°)

Продовження рисунку.

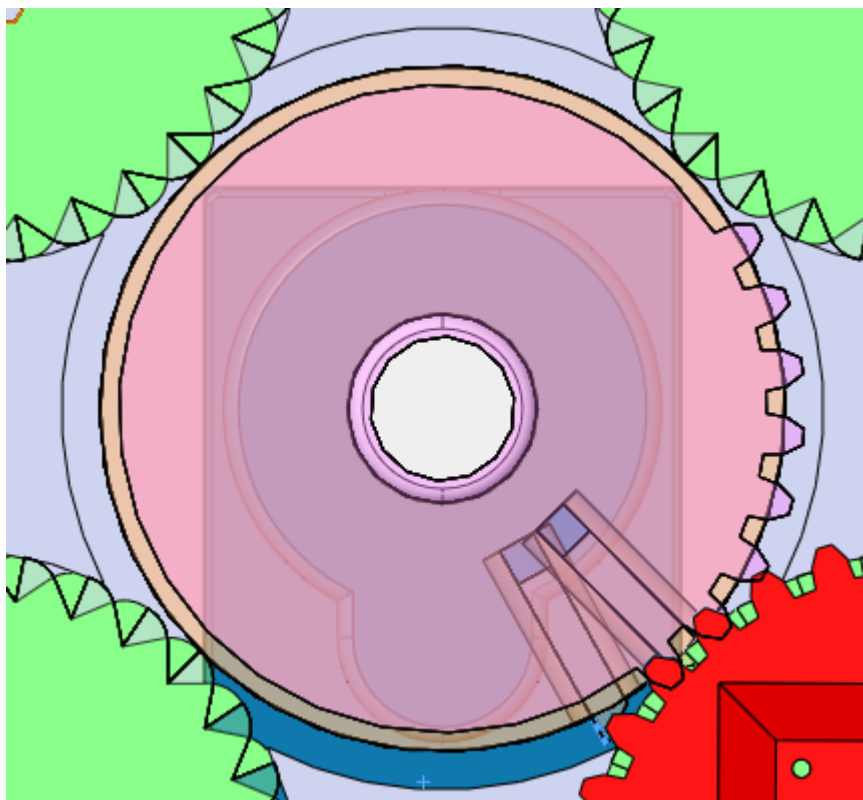


ж - 241° (231° - 251°)



з - 311° (301° - 321°)

Продовження рисунку.



$i - 331^\circ (321^\circ - 341^\circ)$

Продовження рисунку.

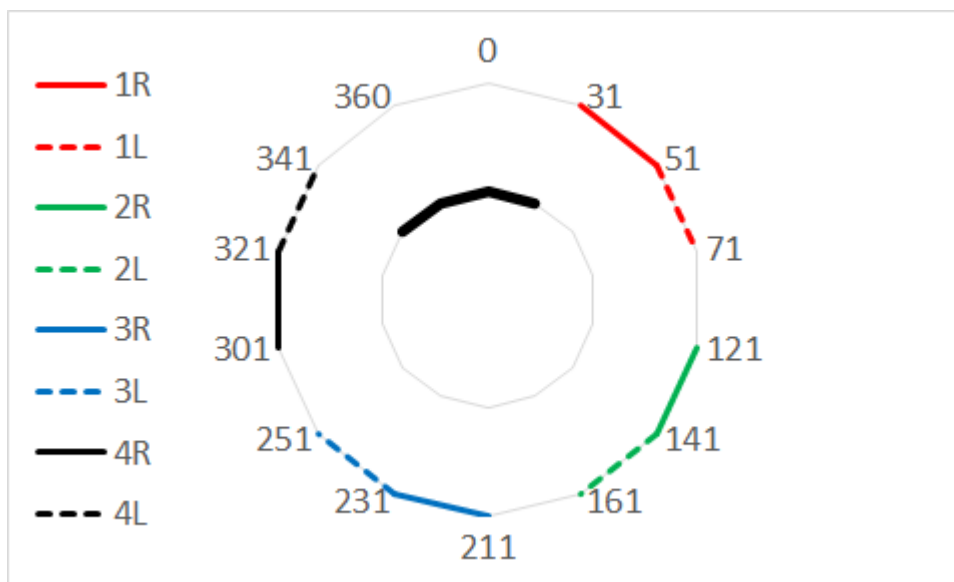


Рисунок 2.8 – Циклограма роботи першого сервопривода для храпових коліс 1-4 та секторного колеса (жирна лінія)

Храпові механізми дозволяють також забезпечити резервування надійності другої та третьої осі у випадку, якщо другий або третій сервопривід виходить з ладу.

Наприклад, якщо другий сервопривід виходить з ладу то храповий механізм переміщує гвинт з Т-подібним кулачком, який рухає ногу робота. Для забезпечення руху ноги в протилежному напрямку використовується пружина. Пружина на рисунку не показана.

Якщо відмовляє третій сервопривід, то храповий механізм переміщує гвинт, який передає рух на нижню ланку ноги за допомогою боуден-тросу. Такий трос часто використовується на велосипедах. Також для забезпечення руху в протилежному напрямку використовується пружина. Ця пружина також не показана на рисунку.

Проблемою може бути подолання сумарного опору пружини і неробочого сервоприводу. Тому бажано щоб неробочі сервоприводи 2 і 3 володіли меншим опором крутному моменту.

Таким чином, навіть якщо сервопривід 2 або 3 відмовляє, то робот має можливість продовжити рух то бази з запасними робочими органами. Якщо роботу вдається переміститись до бази, то він може замінити неробочий орган на новий. Очевидно, що сервопривід 1 повинен володіти вищою надійністю. Наприклад його шестерні можуть бути металевими, а не пластиковими. Відмова сервоприводу 1 не дозволить використовувати механізм резервування надійності та гарячої заміни. В майбутніх дослідженнях планується вирішити цю проблему.

Всього в робочому органі є чотири пари храпових коліс. Перша використовується для загвинчування-розгвинчування роз'єму, друга - для переміщення Т-подібного кулачка і переміщення першої ланки ноги, третій - для переміщення нижньої ланки ноги за допомогою бовден-тросу. Четверта пара не використовується. Проте, у разі потреби, до неї можна приєднати додатковий механізм. Наприклад такий механізм може допомогти перевернути робота в нормальне положення, якщо він перекинеться.

Перевагою розробленої конструкції є її технологічність, так як більшість деталей можуть бути виготовлені шляхом різання акрилового листа лазером на верстаті з числовим програмним керуванням.

Загальна вартість робота не є високою через використання недорогих сервоприводів (близько 1 \$), деталей, які вирізаються з акрилового листа, а також недорогого мікроконтролера esp32 (близько 4 \$), який має вбудовані засоби радіозв'язку та високий потенціал для автономної роботи.

У першу чергу робот призначений для роботи у віддалених місцях, де не може перебувати людина. Наприклад це небезпечні зони з високою забрудненістю або радіоактивністю, або інші планети. Також робот може бути використаний для розмінування місцевості або в інших оборонних цілях.

У найпростішому випадку робот може керуватись шляхом віддаленого керування оператором людиною. Для цього база повинна мати більш потужний радіопередавач, так як вбудований Wi-Fi модуль або Bluetooth не можуть забезпечити відстань передачі більше 100 м. Робот зв'язується з базою через Wi-Fi або Bluetooth, а база передає радіосигнал людині-оператору за допомогою більш потужного радіопередавача.

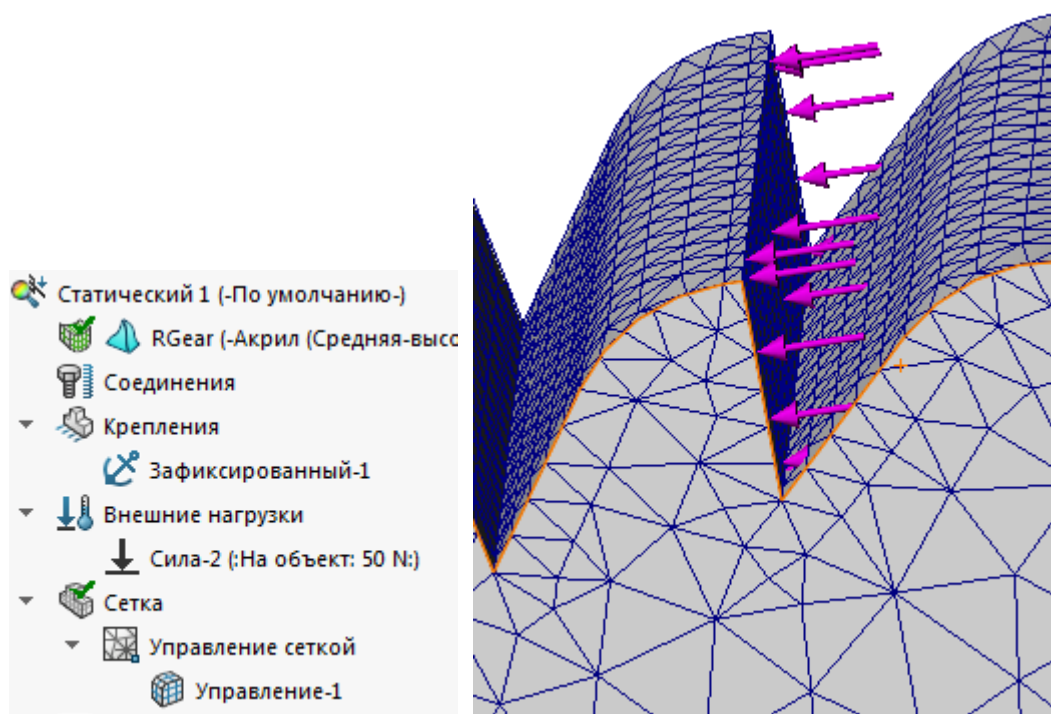
На основі цієї моделі планується виробництво прототипу робота в центрі інноваційного розвитку ІФНТУНГ із матеріалів, які були придбані за фінансової підтримки студентських проєктів ТзОВ «ЛЕОНІ Ваєрінг Системс УА ГмбХ».

2.5 Розрахунок храпового колеса на міцність

За допомогою методу скінченних елементів [13] та модуля SOLIDWORKS Simulation розрахуємо напруження, які виникають в храповому колесі. Умови задачі наступні:

- Матеріал колеса акрил.
- Модуль пружності 3000 МПа.
- Коефіцієнт Пуассона 0,35.
- Границя плинності 45 МПа.
- Сила, що діє на поверхню зуба 50 Н.

Дерево побудови моделі та сітка елементів показані на рисунку 2.9. Колесо нерухомо закріплено по внутрішньому отвору. Сітка в зоні зуба, на який діє сила, є більш дрібною.



а

б

Рисунок 2.9 – Дерево побудови моделі (а) та сітка елементів (б)

Результати (рис. 2.10) показують, що максимальні еквівалентні напруження виникають в западині зуба – 42 МПа. Це менше, ніж границя плинності (45 МПа). Коефіцієнт запасу статичної міцності складає $45/42 = 1,07$, що є задовільним.

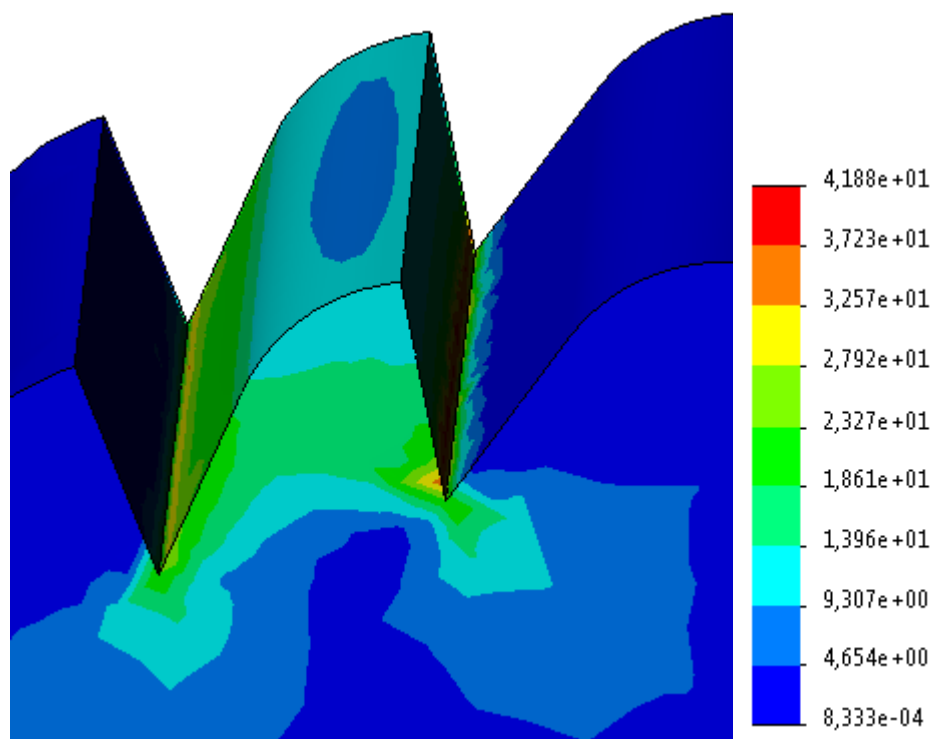


Рисунок 2.10 - Результати – еквівалентні напруження за Мізесом-Губером (МПа)

3 ЕЛЕКТРИЧНА СХЕМА ТА ПРОГРАМА КЕРУВАННЯ РОБОТОМ

3.1 Електрична схема

Електричну схему (рис. 3.1) та програму відлагоджено за допомогою сервісу wokwi.com [21]. На рисунку показано електричну схему з esp32, до якого під'єднано три сервоприводи sg90 та два сенсори - один цифровий, а другий аналоговий. Сервоприводи інших робочих органів під'єднується і працюють за тим самим принципом. Вони на схемі не показані. Важливо відзначити, що у реальній схемі живлення двигунів повинно бути окремим. Для цього можна використовувати мікросхему-стабілізатор напруги, яка на виході видає стабільні 5 В.

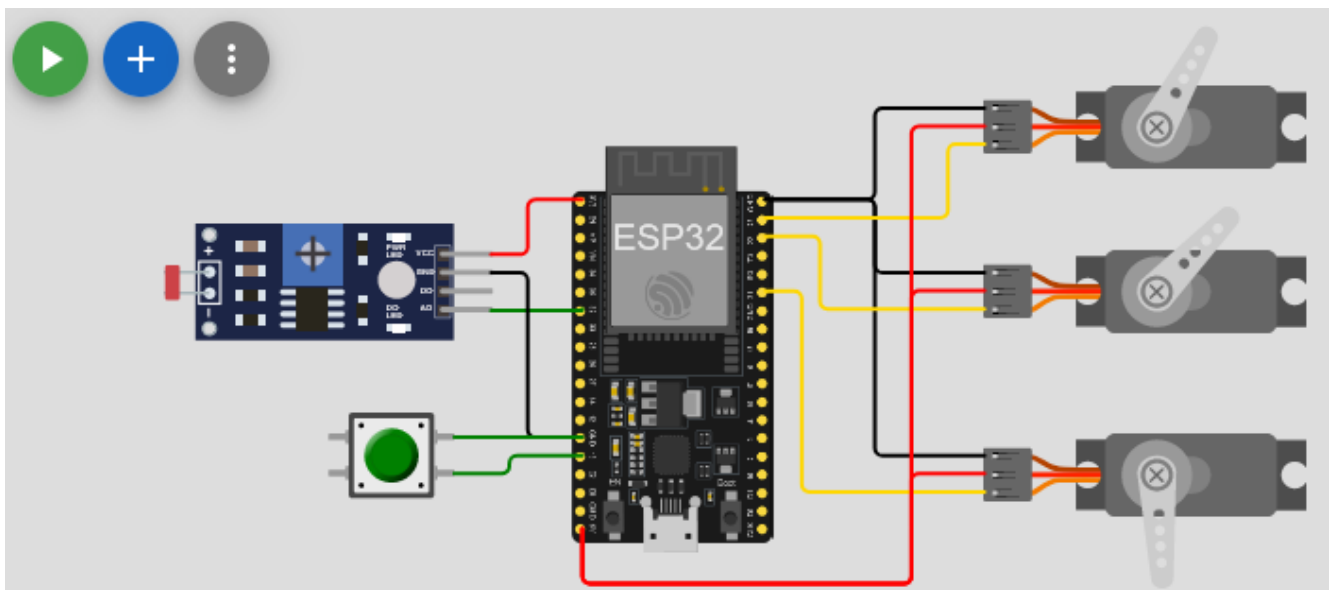


Рисунок 3.1 - Електрична схема під'єднання сенсорів і сервоприводів одного робочого органу

3.2 Функції для керування механізмами

Розроблені Python-функції для керування механізмами одного робочого органу за допомогою трьох сервоприводів показані в наступному коді програми для Micropython:

```
import machine
import time

# цифровий пін для керування сервоприводом
servo_pin1 = machine.Pin(21)
servo_pin2 = machine.Pin(22)
servo_pin3 = machine.Pin(23)

# ініціалізація PWM для керування сервоприводом
servo1 = machine.PWM(servo_pin1, freq=50)
servo2 = machine.PWM(servo_pin2, freq=50)
servo3 = machine.PWM(servo_pin3, freq=50)

# найпростіший цифровий сенсор - кнопка
button = machine.Pin(13, machine.Pin.IN, machine.Pin.PULL_UP)

# аналоговий сенсор освітлення
adc = machine.ADC(machine.Pin(32))
adc atten(machine.ADC.ATTN_11DB)

def rot(servo, a):
    "Повертає серво на кут а"
    # Увага! Перший серво обертається на 360
    servo.duty(int(100*a/180)+24)
    time.sleep(0.01)

def ratchet(n,d,k=1):
    """"Повертає храпове колесо
    n - номер храпового колеса (0-3),
    d - напрямок: 0 (праворуч), 1(ліворуч)
```

k - кількість повторів

"""

кути повороту храповика для різних коліс [праворуч], [ліворуч]

D=[(31,51),(121,141),(211,231),(301,321)], [(71,51),(161,141),(251,231),(341,321)]

for i in range(k):

 rot(servo1, D[d][n][0])

 rot(servo1, D[d][n][1])

def step():

 "Один крок ногою"

 # ногу вперед

 rot(servo1, 360/2)

 rot(servo2, 60)

 rot(servo3, 45)

 time.sleep(0.1)

 # ногу назад

 rot(servo1, 350/2)

 rot(servo2, 50)

 rot(servo3, 35)

 time.sleep(0.1)

def connect():

 "Під'єднує роз'єм - гаряча заміна"

 x=adc.read() # значення сенсора 1

 print(x)

 y=button.value() # значення сенсора 2

 print(y)

 if x<100 and y==0:

 ratchet(0,0,10) # під'єднати

`step()` # крок

... інші кроки

`connect()` # виконати гарячу заміну

Спочатку у нашій програмі створюється об'єкти, які являють собою піни для під'єднання сервоприводів. Кожен сервопривід керується від окремого цифрового піна. Після цього створюються три об'єкти для керування сервоприводами. Керування реалізовано за допомогою широтно-імпульсної модуляції (PWM).

Найпростіший цифровий сенсор (кнопка) під'єднано до піна 13, який працює в режимі входу. Аналоговий сенсор освітлення під'єднано до піна 32, який є входом аналого-цифрового перетворювача (АЦП) в esp32.

Функція `rot()` повертає конкретний сервопривід на кут a . Слід відзначити, що перший сервопривід обертається на 360° , тому значення цього кута треба ділити на два.

Функція `ratchet()` повертає храпове колесо праворуч або ліворуч на один або більшу кількість кроків. Вона використовується для керування храповими механізмами.

Функція `step()` призначена для реалізації одного крокуючого руху ногою робота. В ній виконується один рух ноги вперед і один рух ноги назад для реалізації кроку. Інші ноги робочих органів працюють так само, але у них будуть інші назви сервоприводів.

Функція `connect()` під'єднує роз'єм для реалізації гарячої заміни. Для цього вона зчитує значення двох сенсорів - цифрового і аналогового, і якщо їхнє значення відповідають потрібним, викликається функція `ratchet()`, яка під'єднує роз'єм.

В кінці коду показано приклад виконання програми - виконуються кроки для пересування робота і викликається функція `connect()` для гарячої заміни робочого органу. Для віддаленого керування цей код потрібно доповнити мережевими засобами, що показано нижче.

3.3 Віддалене виконання Python-команд через Wi-Fi

Віддалене керування реалізовано через Wi-Fi. Для цього створено дві програми мовою Python – програма-сервер і програма-клієнт. Програма-сервер запускається з командної оболонки Python на персональному комп'ютері та працює в інтерактивному режимі. Таким чином, оператор може віддалено керувати роботом. Програма-сервер використовує модуль socket для створення точки з'єднання. Коли з'єднання відбулось можна спілкуватись з роботом відсилаючи на нього потрібні команди мовою Python. Також робот присилає необхідно інформацію про його стан. Функція com() отримує цю інформацію і дозволяє передати назад керуючі команди. Наприклад:

```
>>> x=com(b"step()") # рухаємось
1001 1
>>> com(b"connect()") # виконуємо гарячу заміну
```

Код програми-сервера:

```
import socket
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.bind(('192.168.1.116', 8888))
s.listen(1)
soc, addr = s.accept()
print('Server is connected to client', addr)
#soc.close()

def com(y):
    x=soc.recv(255)
    print('Client:', x)
    soc.sendall(y)
    return x
```

```
com(b"print(7)") # приклад виконання команди на esp32
```

Програма-клієнт створена за допомогою Micropython на основі прикладу 04.1_TCP_as_Client [22]. Спочатку в ній відбувається під'єднання до Wi-fi за допомогою функції connectWifi(). Якщо з'єднання то wi-fi успішне, то створюється точка з'єднання за протоколом TCP. Далі, в циклі, робот надсилає на сервер інформацію про свій стан, зокрема інформацію про значення сенсорів і отримує керуючі команди. Ці керуючі команди він виконує за допомогою функції exec(). Таким чином можна виконати довільні команди Micropython.

Код програми-клієнта:

```
import network
import socket
import time

ssidRouter = "*****" # ім'я роутера
passwordRouter = "*****" # пароль роутера
host = "192.168.1.116" # адреса сервера
port = 8888 # порт

wlan=None
s=None

def connectWifi(ssid, passwd): # з'єднатись з роутером
    global wlan
    wlan=network.WLAN(network.STA_IF) # тип вайфай-з'єднання
    wlan.active(True) # включити
    wlan.disconnect()
```

```
wlan.connect(ssid, passwd) # з'єднатись
while (wlan.ifconfig()[0] == '0.0.0.0'): # чекати, поки не приєднається
    time.sleep(1)
return True
```

try:

```
connectWifi(ssidRouter,passwordRouter)
print("connected!")
s = socket.socket()
s.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
s.connect((host,port)) # з'єднатись через TCP-сокет
print("TCP Connected to:", host, ":", port)
while True: # вічний цикл
    s.send('Hello') # передати на сервер дані, наприклад, дані сенсорів
    data = s.recv(255) # отримати від сервера команду
    exec(data) # виконати її
    time.sleep(1) # затримка на 1 секунду
```

except: # якщо в попередньому блоці виникає помилка

```
print("TCP close, please reset!")
if (s):
    s.close() # закрити
wlan.disconnect() # роз'єднатись
wlan.active(False) # вимкнути
```

ВИСНОВКИ

1. Аналіз різних платформ мобільних роботів показав, що крокуюча платформа володіє високою прохідністю, маневреністю та гнучкістю, проте потребує додаткових засобів забезпечення надійності. Огляд відомих конструкцій роботів-гексаподів з квазіколесами і механізмами резервування надійності дозволив сформулювати нову концепцію такого робота з додатковою можливістю автономної заміни вузлів, що відмовили.

2. Розроблено конструкцію робота-гексапода з шістьма робочими органами, кожен з яких має можливість автономної гарячої заміни та має механізми резервування надійності на основі храпових коліс, які можна використати у випадку відмови двох сервоприводів органу. Робот має невисоку вартість компонентів, а його конструкція володіє високою технологічністю. Більшість деталей робота виготовляються шляхом різання лазером акрилового листа на верстаті з ЧПК.

3. Параметричну модель робота та складальні креслення розроблено за допомогою САПР SOLIDWORKS. Модель дозволяє просту зміну параметрів і удосконалення конструкції. Міцність акрилових храпових коліс обґрунтовано в SOLIDWORKS Simulation. Коефіцієнт запасу статичної міцності складає 1,07.

4. Розроблено принципову електричну схему на основі недорогого мікроконтролера esp32. Створено функції керуючої програми для середовища Micropython та виконано їх відлагодження за допомогою сервісу wokwi.com. Реалізовано можливість віддаленого виклику цих функцій та інших довільних команд Micropython через Wi-Fi.

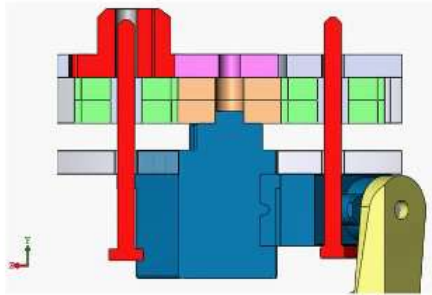
5. На основі розробленої конструкції планується виробництво прототипу робота в центрі інноваційного розвитку ІФНТУНГ із матеріалів, які були придбані за фінансової підтримки студентських проєктів ТзОВ «ЛЕОНІ Ваєрінг Системс УА ГмбХ».

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

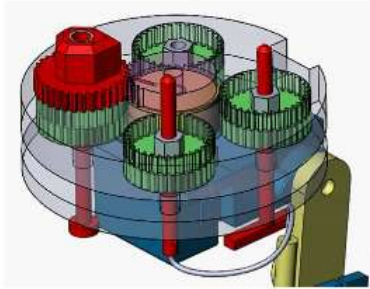
1. Пронюк І. В., Копей В. Б. Мобільний робот-маніпулятор з функцією ідентифікації об'єктів на основі алгоритмів машинного навчання // Нові та нетрадиційні технології в ресурсо- та енергозбереженні: Матеріали міжнародної науково-технічної конференції, 6-7 грудня 2023 р., м. Одеса. Одеса : НУ «Одеська політехніка», 2023. С.281-283.
2. Копей В. Б., Корбеляк Р. В., Пронюк І. В. Оптимізація конструкції мобільного робота за допомогою еволюційних алгоритмів // Нові та нетрадиційні технології в ресурсо- та енергозбереженні: Матеріали міжнародної науково-технічної конференції, 6-7 грудня 2023 р., м. Одеса. Одеса : НУ «Одеська політехніка», 2023. С.143-145.
3. Колесо Ілона. URL: https://uk.wikipedia.org/wiki/Колесо_Ілона (accessed 10.06.24).
4. Гусенична стрічка. URL: https://uk.wikipedia.org/wiki/Гусенична_стрічка (accessed 10.06.24).
5. Boston Dynamics. URL: https://uk.wikipedia.org/wiki/Boston_Dynamics (accessed 10.06.24)
6. Kopei V. B., Proniuk I. V. Mobile robot with quasi-wheels and a mechanism for reserving reliability // Conference Proceedings of XII International Scientific and Technical Conference “ADVANCED TECHNOLOGIES IN MECHANICAL ENGINEERING”, 5-9 February 2024. Ivano-Frankivsk – Yaremche, 2024.. Ivano-Frankivsk : IFNTUOG, 2024. P.24-25.
7. Пронюк І. В., Копей В. Б. Робот «павук» з розширеною функціональністю // Збірник наукових праць XII Міжнародної науково-технічної конференції "Прогресивні технології у машинобудуванні АТМЕ-2024", 5-9 лютого 2024 р. Івано-Франківськ-Яремче, 2024. Івано-Франківськ : ІФНТУНГ, 2024. С.128-129.
8. Clark Haynes, Sarjoun Skaff, Al Rizzi, Uluc Saranlı, Hanns Tappeiner. RiSE Robot. URL: <http://www.msl.ri.cmu.edu/projects/rise/> (accessed 10.06.24).

9. The ATHLETE Rover. URL: <https://web.archive.org/web/20060929014741/http://www-robotics.jpl.nasa.gov/systems/system.cfm?System=11> (accessed 10.06.24).
10. ESP32. URL: <https://uk.wikipedia.org/wiki/ESP32> (accessed 10.06.24)
11. MicroPython - Python for microcontrollers. URL: <https://micropython.org> (accessed 10.06.24).
12. Копей В. Б. Мова програмування Python для інженерів і науковців : навчальний посібник. Івано-Франківськ : ІФНТУНГ, 2019. 272 с.
13. Копей Б.В., Копей В.Б. Використання методу скінченних елементів та тривимірного комп'ютерного моделювання для конструювання та оптимізації параметрів нафтогазового обладнання: Навчальний посібник. Івано-Франківськ : Факел, 2008. 117 с.
14. Копей В.Б., Онисько О.Р., Борушак Л.О., Роп'як Л.Я. Автоматизоване проектування різальних інструментів: Навчальний посібник. Івано-Франківськ : ІФНТУНГ, 2012. 208 с.
15. Grabcad Library. URL: <https://grabcad.com/library> (accessed 10.06.24).
16. Jacob Garcia-Pavy One-cell Battery Box Battery Holder. URL: <https://grabcad.com/library/one-cell-battery-box-battery-holder-1> (accessed 10.06.24)
17. Yasir Mahmood. ESP32-WROVER-DEVKIT-NODEMCU-V1.3 CASE. URL: <https://grabcad.com/library/esp32-wrover-devkit-nodemcu-v1-3-case-1> (accessed 10.06.24).
18. danska. Small Blue Micro Servo. URL: <https://grabcad.com/library/small-blue-micro-servo-1> (accessed 10.06.24).
19. Боуден-трос. URL: <https://uk.wikipedia.org/wiki/Боуден-трос> (accessed 10.06.24).
20. Датчик ефекту Холла. URL: https://uk.wikipedia.org/wiki/Датчик_ефекту_Холла (accessed 10.06.24).
21. Simulate IoT Projects in Your Browser. URL: wokwi.com (accessed 10.06.24).
22. Freenove_ESP32_WROVER_Board. URL: https://github.com/Freenove/Freenove_ESP32_WROVER_Board (accessed 10.06.24).

ДОДАТКИ



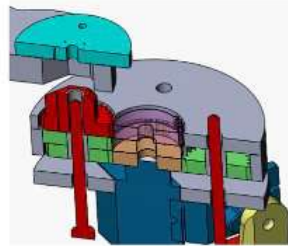
а



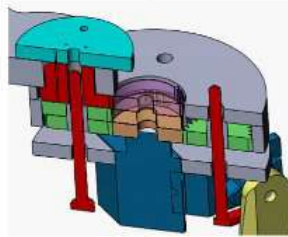
б

а - основний чертєж, б - прозорий корпус

Механізм робочого органу, який забезпечує його обертання навколо вертикальної осі, автономну гарантійну та резервування надійності

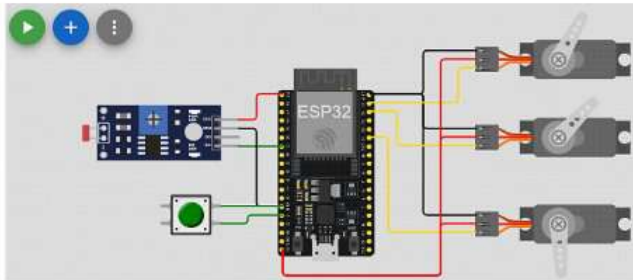


а



б

Етапи гарантійної заміни робочого органу: а - від'єднано, б - приєднано (але ще не закінчено)



Електрична схема під'єднання сенсорів і сервоприводів одного робочого органу

Модуль для керування механізмом

```

import machine
import time

# ширини піч для керування сервоприводом
servo_pin1 = machine.Pin(2)
servo_pin2 = machine.Pin(22)
servo_pin3 = machine.Pin(23)

# ітераційний PWM для керування сервоприводом
servo1 = pwm.pwm(1, servo_pin1, freq=50)
servo2 = machine.PWM(servo_pin2, freq=50)
servo3 = machine.PWM(servo_pin3, freq=50)

# найпростіший ширинний сенсор - кнопка
button = machine.Pin(5), machine.Pin(6), machine.Pin(Pull_UP)

# ітераційний ітератор по функціях
adc = machine.ADC(machine.Pin(32))
adc.start(machine.ADC.ATTN_1DB)

def raise_servo(i):
    """Підняти серво на кут i"""
    # Ширині серво і кут обертання на 90
    servo_duty = int(100 * i / 90 + 25)
    time.sleep(0.05)

def raise_servo(i):
    """Підняти серво на кут i"""
    i - номер крайнього колеса (0-3)
    i - напруження і частота ітерацій
    k - кількість ітерацій
    ...

# який підключити сервопривод для певної осі (servo_pin1, servo_pin2)
(1, 0, 15), (2, 1, 4, 0, 2, 1, 2, 3), (3, 0, 1, 2, 0), (7, 1, 5, 8, 16, 14, 0, 2, 5, 2, 0, 0, 4, 1, 3, 2)
for i in range(4):
    raise_servo(1)
    raise_servo(2)

def stop():
    """Підняти серво на кут i"""
    # часу бездіяльності
    raise_servo(360/2)
    raise_servo(60)
    raise_servo(45)
    time.sleep(0.5)
    # часу очікування
    raise_servo(15/2)
    raise_servo(50)
    raise_servo(35)
    time.sleep(0.5)

def connect():
    """Підняти серво на кут i"""
    i - номер сервоприводу
    print(i)
    i - button value # значення сенсора 2
    print(i)
    i - x=100 and y=0
    i = int(i/10, 2) # під'єднати

stop() # кут
i - номер серво
connect() # виконати гарантійну заміну
    
```

```

Код програми сервера

import socket
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.bind(("192.168.1.100", 8080))
s.listen()
soc, addr = s.accept()
print("Server is connected to client", addr)
s.close()

def client():
    s = socket.socket()
    print("Client", s)
    s.connect(("192.168.1.100", 8080))
    return s

s.close() # кроки виконання команди на esp32
    
```

```

Код програми клієнта

import network
import socket
import time

ssid_router = "*****" # ім'я роутера
password_router = "*****" # пароль роутера
host = "192.168.1.100" # адреса сервера
port = 8080 # порт

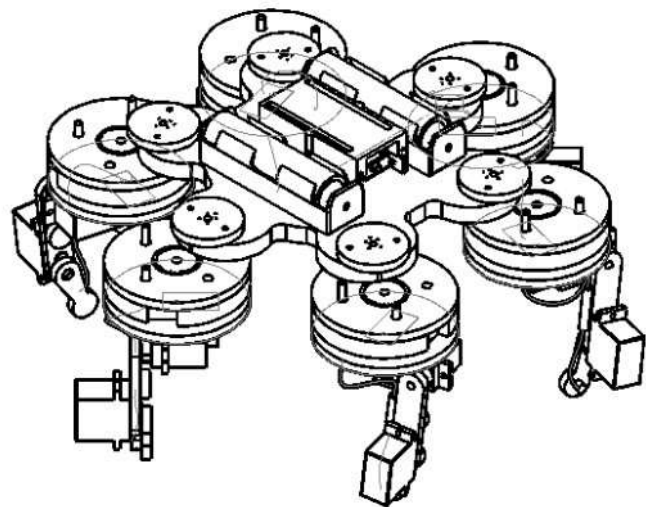
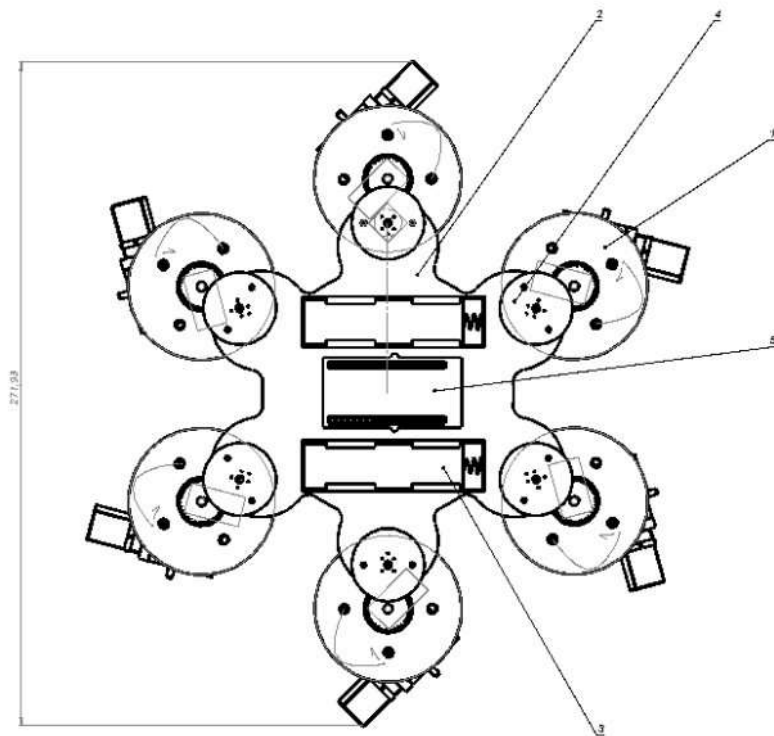
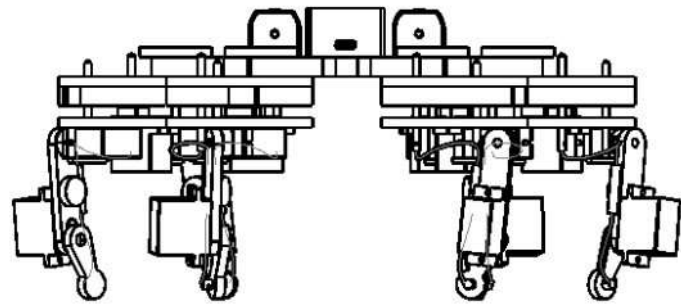
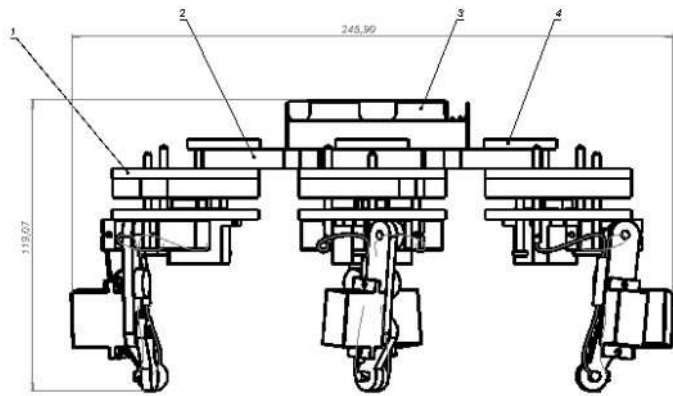
wlan_name = None
s = None

def connect_wifi(ssid_router, password_router):
    global wlan
    wlan = network.WLAN(network.STA_IF) # стан бездіяльності
    wlan.active(True) # виконати
    wlan.disconnect()
    wlan.connect(ssid_router, password_router) # під'єднати
    while wlan.config[0][0] == 0: # чекаємо, поки не під'єднано
        time.sleep(0.1)
    return True

try:
    connect_wifi(ssid_router, password_router)
    print("connected")
    s = socket.socket()
    s.connect(("192.168.1.100", 8080)) # під'єднати через TCP-socket
    print("TCP connect")
    while True: # цикл очікування
        s.send("Hello") # передаємо на сервер дані, наприклад, дані сенсора
        data = s.recv(255) # отримати від сервера команду
        exec(data) # виконати i)
        time.sleep(0.5) # затримка на 1 секунду
    except: # якщо в процесі очікування з'явилася помилка
        print("TCP connect error")
        s.close() # закрити
    wlan.disconnect() # роз'єднати
    wlan.active(False) # бездіяльність
    
```

				BP - 000.05					
№ п/п	№ докум.	Дата	Версія	Програма керування механізмом			Лист	Масштаб	Всього
1	Кодовий Б.Б.	10.2.2023	1.0	Програма керування механізмом			11		1
2	Кодовий Б.Б.	10.2.2023	1.0	Програма керування механізмом					
				ІФТТУНГ					
				Корпус			Формат А1		

Лист №...
 Лист №...
 Лист №...
 Лист №...



				БР - 000 01 СК			
				Робот-паук			
№	Изм.	Исполн.	Дата	№	Изм.	Исполн.	Дата
1		САЛН.Е.Е.		1	1.1	САЛН.Е.Е.	2.1
2		САЛН.Е.Е.		2		САЛН.Е.Е.	
3		САЛН.Е.Е.		3		САЛН.Е.Е.	
4		САЛН.Е.Е.		4		САЛН.Е.Е.	
5		САЛН.Е.Е.		5		САЛН.Е.Е.	
6		САЛН.Е.Е.		6		САЛН.Е.Е.	
7		САЛН.Е.Е.		7		САЛН.Е.Е.	
8		САЛН.Е.Е.		8		САЛН.Е.Е.	

№	Изм.	Исполн.	Дата
1		САЛН.Е.Е.	
2		САЛН.Е.Е.	
3		САЛН.Е.Е.	
4		САЛН.Е.Е.	
5		САЛН.Е.Е.	
6		САЛН.Е.Е.	
7		САЛН.Е.Е.	
8		САЛН.Е.Е.	