

**Міністерство освіти і науки України**

**Луцький національний технічний університет**

(повне найменування вищого навчального закладу)

**Факультет транспорту та механічної інженерії**

(повне найменування факультету)

**Кафедра прикладної механіки та мехатроніки**

(повна найменування кафедри)

**КВАЛІФІКАЦІЙНА РОБОТА  
ЗА СТУПЕНЕМ ВИЩОЇ ОСВІТИ «МАГІСТР»**

**Розробка інтелектуальної системи управління  
промислового робота із дистанційним зворотнім  
зв'язком**

спеціальність 131 Прикладна механіка

(шифр і назва спеціальності)

освітня програма «Прикладна механіка»

(назва освітньої програми)

Виконав: здобувач вищої освіти  
групи ІМмз-11  
Кревський Тарас Олегович

\_\_\_\_\_  
(підпис)

Керівник:  
к.т.н., доцент  
Сичук Віктор Анатолійович

\_\_\_\_\_  
(підпис)

Кваліфікаційну роботу  
допущено до захисту  
«\_\_» \_\_\_\_\_ 20\_\_ р.  
Гарант освітньої програми:  
к.т.н., доцент  
Четвержук Тарас Іванович

\_\_\_\_\_  
(підпис)

Луцьк – 2024 року

Луцький національний технічний університет

Факультет Транспорту та механічної інженерії  
Кафедра Прикладної механіки та мехатроніки  
Другий (магістерський) рівень  
освітньо-професійної програми «Прикладна механіка»  
Спеціальність 131 Прикладна механіка

**ЗАТВЕРДЖУЮ**

Завідувач кафедри  
\_\_\_\_\_ Р.Г. Редько  
“ \_\_\_\_\_ ” \_\_\_\_\_ 2024 р.

**ЗАВДАННЯ  
НА КВАЛІФІКАЦІЙНУ РОБОТУ МАГІСТРА**

Кревський Тарас Олегович  
(прізвище, ім'я, по батькові)

1. Тема роботи Розробка інтелектуальної системи управління промислового робота із дистанційним зворотнім зв'язком, керівник кваліфікаційної роботи магістра Сичук В.А. затверджені наказом вищого навчального закладу від «30» грудня 2023 р., № 452/01-02.
2. Строк подання студентом роботи 10.12.2024 р.
3. Вихідні дані до роботи: документації: плати Arduino Uno, ультразвукового дальноміра HC-SR04, тензодатчика HX711, драйверів TB6600 та DM556.
4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)  
Вступ. 1 Загальна частина. 2 Конструкторська частина. 3 Технологічна частина. 4 Дослідна частина. Висновок. Список використаних джерел.
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)  
Електрична схема роботизованої системи – 1 лист (ф. А1), 3D модель системи управління із зворотнім зв'язком – 1 лист (ф. А1), Програмне забезпечення плати Arduino UNO – 1 лист (ф. А1), Насадка – 1 лист (ф. А2), Лоток – 1 лист (ф. А3), Гайка висувна – 1 лист (ф. А3).

6. Консультанти розділів роботи

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
Загальний	Сичук В.А.		
Конструкторський	Сичук В.А.		
Технологічний	Сичук В.А.		
Дослідний	Сичук В.А.		

7. Дата видачі завдання \_\_\_\_\_

Керівник \_\_\_\_\_

(підпис)

Завдання прийняв до виконання \_\_\_\_\_

(підпис)

**КАЛЕНДАРНИЙ ПЛАН**

№	Назва етапів виконання кваліфікаційної роботи магістра	Термін виконання етапів виконання кваліфікаційної роботи магістра	Примітка
1.	Загальна частина.		
2.	Конструкторська частина.		
3.	Технологічна частина.		
4.	Дослідна частина.		
5.	Представлення роботи до захисту		
6.	Електронний варіант кваліфікаційної роботи магістра		

Студент \_\_\_\_\_

(підпис)

Кревський Т.О.  
(прізвище та ініціали)

Керівник роботи \_\_\_\_\_

(підпис)

Сичук В.А.  
(прізвище та ініціали)

## АНОТАЦІЯ

Кревський Т.О. Розробка інтелектуальної системи управління промислового робота із дистанційним зворотнім зв'язком. – Рукопис.

Кваліфікаційна робота магістра на здобуття кваліфікації другого (магістерського) рівня вищої освіти із спеціальності 131 Прикладна механіка спеціалізації технологія машинобудування. – Луцький національний технічний університет. – Луцьк, 2024.

Сутність роботи полягає у проектуванні, виготовленні та програмуванні роботи роботизованої системи, у якій реалізовано зворотній зв'язок, тобто виконання операцій роботом напряму залежить від самого стану виконання необхідних переходів. В результаті розроблено дослідний стенд у кому реалізовані вище перелічені функції. В якості датчиків зворотного зв'язку застосовані наступні елементи: ультразвуковий дальномір HC-SR04, тензодатчик HX711, два кольорових світлодіоди жовтого та блакитного кольорів.

Ключові слова: зворотній зв'язок, роботизована система, тензодатчик, ультразвуковий дальномір.

## ANNOTATION

Krevskiy T.O. Development of an intelligent control system for an industrial robot with remote feedback. - Manuscript.

Master's Qualification for Qualification of the Second (Master's) Level of Higher Education in Specialty 131 Applied Mechanics of Specialization mechanical engineering technology. - Lutsk National Technical University. - Lutsk, 2024.

The essence of the work is to design, manufacture and program the operation of a robotic system in which feedback is implemented, i.e. the performance of operations by the robot directly depends on the state of the necessary transitions. As a result, a research stand was developed in which the above functions were implemented. The following elements were used as feedback sensors: an ultrasonic rangefinder HC-SR04, a strain gauge HX711, two color LEDs of yellow and blue colors.

Keywords: feedback, robotic system, strain gauge, ultrasonic rangefinder.

## Зміст

Вступ.....	
1 ЗАГАЛЬНИЙ РОЗДІЛ.....	
1.1 Принципи та теорія автоматизації.....	
1.2 Типи зворотного зв'язку роботизованих систем.....	
1.3 Висновки щодо розділу.....	
2 КОНСТРУКТОРСЬКИЙ РОЗДІЛ.....	
2.1 Моделювання певних деталей та вузлів роботизованого станда.....	
2.2 Моделювання збірки роботизованого станда.....	
2.3 Висновок по розділу .....	
3 ТЕХНОЛОГІЧНИЙ РОЗДІЛ.....	
3.1 Налагодження обробки та виготовлення деталі насадка за допомогою 3D друку.....	
3.2 Налагодження обробки та виготовлення деталі лоток на фрезерному верстаті з ЧПУ.....	
3.3 Налагодження техпроцесу та виготовлення деталі висувна гайка класичним обробленням.....	
3.4 Вибір електронних компонентів для роботи роботизованого комплексу з зворотнім зв'язком.....	
3.5 Висновки щодо розділу.....	
4 ДОСЛІДНИЦЬКИЙ РОЗДІЛ.....	
4.1 Алгоритм роботи на випробування роботизованої системи з зворотнім зв'язком.....	
4.2 Розробка електричної схеми роботи роботизованого станду.....	
4.3 Розробка прошивки контролера який керує роботою роботизованої системи.....	
4.4 Складання усіх елементів роботизованого станду.....	
4.5 Висновки по розділу.....	
Висновок.....	
Список використаних джерел.....	

## ВСТУП

Роботизовані системи з зворотнім зв'язком є дуже важливими при виконанні виробничих робіт, оскільки зворотній зв'язок по суті контролює роботу усієї системи і при можливих збоях в автоматичному режимі відновить роботу.

**Актуальність теми** та доцільність магістерського дослідження важко переоцінити, бо проектуючи та розробляючи роботизований комплекс в якому реалізовано зворотній зв'язок можна на власному досвіді переконатися про важливість такого типу зв'язку.

**Мета** роботи полягає у розробці дослідного роботизованого станку по перенесенню/сортванню деталей враховуючи їх масу. В даній системі в якості зворотного зв'язку є три електронних компоненти, а саме ультразвуковий датчик, датчик маси, сигнальні світлодіоди

**Об'єкт дослідження** – дослідний роботизований станок який призначений для переміщення деталей.

**Предмет дослідження** – реалізація зворотного зв'язку в роботизованій системі та дослідження його впливу на роботу системи загалом, виявлення позитивних сторін щодо наявності такого виду зв'язку.

**Методи дослідження**, що застосовувалися при виконанні роботи пов'язані з 3D моделюванням, розробкою G-коду для обробки тестової деталі в програмі FeatureCAM та Creality Slicer.

**Наукова новизна одержаних результатів** полягає в реалізації зворотного зв'язку роботизованого станку та отримання позитивних результатів при проведенні експериментальних випробувань.

**Практичне значення одержаних результатів** полягає в виготовленні навчально/наукового/дослідного роботизованого станку на якому можна проводити експериментальні дослідження в сфері роботизації та механотроніки та управлінню виробничими процесами.

**Особистий внесок здобувача.** Основні результати, які становлять суть магістерської роботи, висновки і рекомендації не викликають сумніву. Було поставлено мету та задачі досліджень, також проведено усі роботи щодо проектування, виготовлення та налагодження роботизованого станку який

виконує роботу і слідкує за її виконанням, а в разі виникненні неполадок у виконанні роботи виправляє їх.

**Апробація отриманих результатів роботи.** Основні ідеї та результати роботи розглядалися на науково-практичних семінарах кафедри.

**Структура та обсяг роботи.** Випускна робота магістра має вступ, 4 розділи, висновки, список використаних джерел. Результат досліджень наведений на \_\_\_ сторінках машинописного тексту, ілюструється \_\_\_ рисунками та \_\_\_ таблицями. Список використаних джерел складає \_\_\_ найменувань.

# 1 ЗАГАЛЬНИЙ РОЗДІЛ

## 1.1 Принципи та теорія автоматизації

Скористаємося джерелом [1] для ознайомлення з поняттями та прикладами принципів та теорії автоматизації у виробництвах.

Описані вище розробки забезпечили три основні будівельні блоки автоматизації: (1) джерело живлення для виконання певних дій, (2) засоби керування зворотним зв'язком і (3) машинне програмування. Майже без винятку автоматизована система демонструватиме всі ці елементи.

### Джерело живлення

Автоматизована система призначена для виконання певної корисної дії, і ця дія вимагає потужності. Існує багато доступних джерел живлення, але найчастіше в сучасних автоматизованих системах використовується електроенергія. Електрична енергія є найбільш універсальною, оскільки її можна легко генерувати з інших джерел (наприклад, викопного палива, гідроелектростанцій, сонячної та ядерної енергії) і її можна легко перетворювати в інші види енергії (наприклад, механічну, гідравлічну та пневматичну) виконувати корисну роботу. Крім того, електрична енергія може зберігатися у високопродуктивних акумуляторах з тривалим терміном служби.

Дії, які виконуються автоматизованими системами, зазвичай бувають двох типів: (1) обробка та (2) передача та позиціонування. У першому випадку енергія використовується для виконання певної операції обробки над деякою сутністю. Процес може включати формування металу, формування пластику, перемикання електричних сигналів у системі зв'язку або обробку даних у комп'ютеризованій інформаційній системі. Усі ці дії передбачають використання енергії для перетворення об'єкта (наприклад, металу, пластику, електричних сигналів або даних) з одного стану чи стану в

інший, більш цінний стан чи стан. Другий тип дій — перенесення та позиціонування — найбільш легко помітити в автоматизованих виробничих системах, призначених для роботи над продуктом. У цих випадках продукт, як правило, має бути переміщений (перенесений) з одного місця в інше під час ряду етапів обробки. У кожному місці обробки зазвичай потрібне точне позиціонування продукту. В автоматизованих комунікаційних та інформаційних системах терміни передача та позиціонування стосуються переміщення даних (або електричних сигналів) між різними процесорними пристроями та доставки інформації до вихідних терміналів (принтерів, відеодисплеїв тощо) для інтерпретації та використання людей.

Управління зі зворотним зв'язком широко використовується в сучасних автоматизованих системах. Система керування зі зворотним зв'язком складається з п'яти основних компонентів: (1) входу, (2) процесу, що контролюється, (3) виходу, (4) чутливих елементів і (5) контролера та виконавчих пристроїв. Ці п'ять компонентів показано на малюнку 1. Для опису такого типу системи часто використовується термін керування зі зворотним зв'язком.

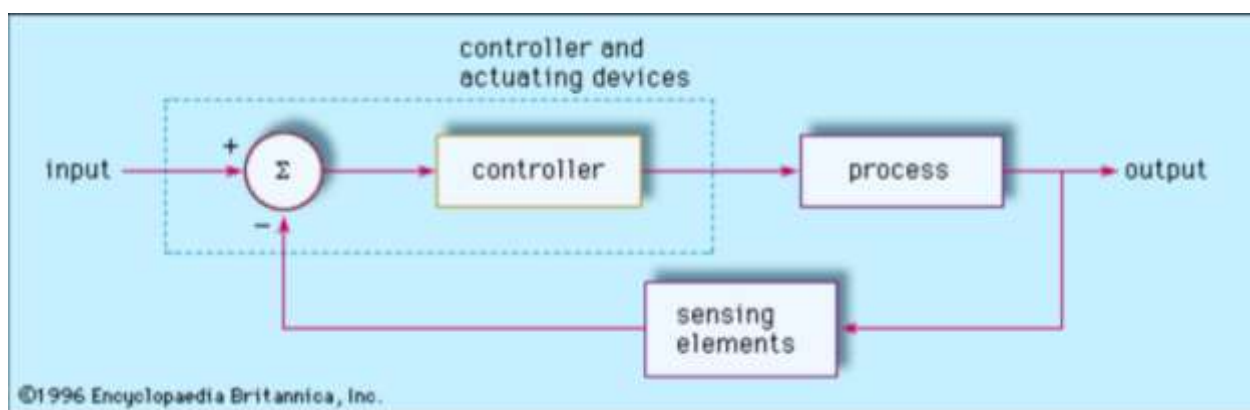


Рисунок 1.1 – Компоненти системи керування зі зворотним зв'язком та їхні взаємозв'язки.

Вхідними даними системи є еталонне значення або задане значення для виходу системи. Це являє собою бажане робоче значення виходу.

Використовуючи попередній приклад системи опалення як ілюстрацію, вхідні дані є бажаним налаштуванням температури для кімнати. Процесом, яким керують, є нагрівач (наприклад, піч). В інших системах зворотного зв'язку процес може бути виробничою операцією, ракетними двигунами космічного човника, автомобільним двигуном круїз-контролю або будь-яким іншим процесом, до якого подається енергія. Вихід - це змінна процесу, яка вимірюється та порівнюється з входом; у наведеному вище прикладі це кімнатна температура.

Чутливі елементи — це вимірювальні пристрої, які використовуються в контурі зворотного зв'язку для контролю значення вихідної змінної. У прикладі системи опалення ця функція зазвичай виконується за допомогою біметалічної стрічки. Цей пристрій складається з двох металевих смуг, з'єднаних по довжині. Два метали мають різні коефіцієнти теплового розширення; таким чином, коли температура стрічки підвищується, вона згинається прямо пропорційно зміні температури. Таким чином, біметалева стрічка здатна вимірювати температуру. Існує багато різних типів датчиків, які використовуються в системах керування зі зворотним зв'язком для автоматизації.

Призначення контролера та виконавчих пристроїв у системі зворотного зв'язку полягає в тому, щоб порівняти виміряне вихідне значення з опорним вхідним значенням і зменшити різницю між ними. Загалом, контролер і виконавчий механізм системи є механізмами, за допомогою яких здійснюються зміни в процесі для впливу на вихідну змінну. Ці механізми зазвичай розроблені спеціально для системи та складаються з таких пристроїв, як двигуни, клапани, електромагнітні перемикачі, поршневі циліндри, шестерні, силові гвинти, системи шківів, ланцюгові передачі та інші механічні й електричні компоненти. Перемикач, підключений до біметалічної планки терморегулятора, є регулятором і виконавчим пристроєм системи опалення. Коли потужність (температура в приміщенні) нижче

встановленого значення, перемикач вмикає нагрівач. Коли температура перевищує встановлену температуру, обігрів вимикається.

### Машинне програмування

Запрограмовані інструкції визначають набір дій, які система повинна автоматично виконати. Програма вказує, що повинна робити автоматизована система і як повинні функціонувати її різні компоненти, щоб досягти бажаного результату. Зміст програми значно відрізняється від однієї системи до іншої. У відносно простих системах програма складається з обмеженої кількості чітко визначених дій, які виконуються безперервно і багаторазово в належній послідовності без відхилень від одного циклу до іншого. У більш складних системах кількість команд може бути досить великою, а рівень деталізації кожної команди може бути значно вищим. У відносно складних системах програма передбачає зміну послідовності дій у відповідь на зміну сировини або інших робочих умов.

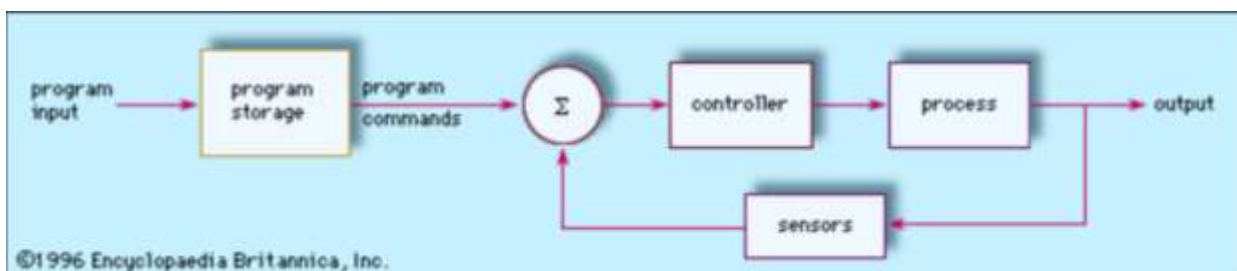


Рисунок 1.2 – Взаємозв’язок програмного керування та керування зворотним зв’язком в автоматизованій системі.

Команди програмування пов’язані з керуванням зі зворотним зв’язком в автоматизованій системі, оскільки програма встановлює послідовність значень для входів (заданих точок) різних контурів керування зі зворотним зв’язком, які складають автоматизовану систему. Дана команда програмування може вказати задану точку для контуру зворотного зв’язку, який, у свою чергу, керує деякою дією, яку має виконати система. По суті, метою циклу зворотного зв’язку є перевірка того, що запрограмований крок виконано. Наприклад, у контролері робота програма може вказати, що рука

має переміститися у визначене положення, а система керування зворотним зв'язком використовується для перевірки правильності виконання руху. Взаємозв'язок програмного керування та керування зі зворотним зв'язком в автоматизованій системі проілюстровано на рисунку 2.

Деякі із запрограмованих команд можуть виконуватись у простому режимі відкритого циклу, тобто без потреби в циклі зворотного зв'язку для перевірки правильності виконання команди. Наприклад, команда натиснути електричний вимикач може не вимагати зворотного зв'язку. Потреба в управлінні зворотним зв'язком в автоматизованій системі може виникнути, коли є зміни в сировині, що подається у виробничий процес, і система повинна враховувати ці зміни, вносячи коригування в свої контрольовані дії. Без зворотного зв'язку система не зможе здійснювати достатній контроль над якістю результату процесу.

Запрограмовані команди можуть міститися на механічних пристроях (наприклад, механічних кулачках і зв'язках), перфорованій паперовій стрічці, магнітній стрічці, магнітних дисках, пам'яті комп'ютера або на будь-якому іншому носії, який протягом багатьох років розроблявся для певних застосувань. Сьогодні для автоматизованого обладнання зазвичай використовується комп'ютерна технологія зберігання як засіб для зберігання запрограмованих команд і перетворення їх у керовані дії. Однією з переваг комп'ютерного сховища є те, що програму можна легко змінити або вдосконалити. Зміна програми, яка міститься на механічних кулачках, вимагає значної роботи.

Програмовані машини часто здатні приймати рішення під час роботи. Можливість прийняття рішень міститься в керуючій програмі у формі логічних інструкцій, які керують роботою такої системи за різних обставин. За певних обставин система реагує в один бік; за різних обставин воно реагує іншим способом. Є кілька причин для забезпечення автоматизованої системи можливістю прийняття рішень, включаючи (1) виявлення та відновлення

помилки, (2) моніторинг безпеки, (3) взаємодію з людьми та (4) оптимізацію процесу.

Виявлення та відновлення помилок стосується рішень, які повинна прийняти система у відповідь на небажані умови роботи. У роботі будь-якої автоматизованої системи іноді виникають збої та помилки під час нормального циклу операцій, для яких необхідно вжити певних коригувальних дій для відновлення системи. Звичайною реакцією на несправність системи було звернення за допомогою людини. Існує зростаюча тенденція в автоматизації та робототехніці, щоб дозволити системі самій відчувати ці несправності та виправляти їх певним чином без втручання людини. Це визначення та виправлення називаються виявленням помилок і відновленням, і це вимагає, щоб у системі була запрограмована можливість прийняття рішень.

Моніторинг безпеки — це особливий випадок виявлення та відновлення помилок, коли несправність становить загрозу безпеці. Рішення потрібні, коли датчики автоматизованої системи виявляють, що виникли умови безпеки, які можуть бути небезпечними для обладнання або людей поблизу обладнання. Метою системи моніторингу безпеки є виявлення небезпеки та вжиття найбільш відповідних заходів для її усунення або зменшення. Це може включати зупинку роботи та сповіщення обслуговуючого персоналу про стан, або це може включати більш складний набір дій для усунення проблеми безпеки.

Автоматизовані системи зазвичай повинні певним чином взаємодіяти з людьми. Банківський автомат, наприклад, повинен отримувати інструкції від клієнтів і діяти відповідно. У деяких автоматизованих системах можлива низка різних інструкцій від людей, і здатність системи приймати рішення має бути досить складною, щоб мати справу з масивом можливостей.

Четвертою причиною прийняття рішень в автоматизованій системі є оптимізація процесу. Потреба в оптимізації найчастіше виникає в процесах, у яких є економічний критерій ефективності, оптимізація якого є бажаною.

Наприклад, мінімізація витрат зазвичай є важливою метою у виробництві. Автоматизована система може використовувати адаптивне керування для отримання відповідних сигналів датчиків та інших вхідних даних і прийняття рішень для приведення процесу до оптимального стану.

## **1.2 Типи зворотного зв'язку роботизованих систем**

Скористаємося джерелом [2] для аналізу типів зворотного зв'язку роботизованих систем

### 1 Пропорційне керування

Пропорційне керування — це найпростіший тип системи керування зі зворотним зв'язком, де корекція, застосована до входу, пропорційна похибці. Наприклад, якщо рука робота повинна досягти певного положення, пропорційний контролер збільшить або зменшить напругу на двигуні залежно від того, наскільки далеко рука знаходиться від цільового положення. Перевага пропорційного керування полягає в тому, що його легко реалізувати та зрозуміти, і воно може забезпечити швидку реакцію. Однак пропорційне керування має деякі недоліки, такі як похибка стаціонарного режиму, перерегулювання та коливання.

### 2 Пропорційно-інтегрально-похідна управління

Пропорційно-інтегрально-похідна (PID) керування — це більш досконалий тип системи керування зі зворотним зв'язком, де корекція, застосована до вхідного сигналу, є комбінацією трьох термінів: пропорційного, інтегрального та похідної. Пропорційний член такий самий, як і в пропорційному регулюванні, інтегральний член накопичує похибку з часом і усуває похибку в стаціонарному стані, а похідний член передбачає майбутню похибку та зменшує перерегулювання та коливання. ПІД-регулювання широко використовується в робототехніці, оскільки воно може досягти високої точності та стабільності для різних систем. Однак ПІД-

регулювання вимагає ретельного налаштування параметрів і може не працювати належним чином для нелінійних або змінних у часі систем.

### 3 Адаптивне керування

Адаптивне керування — це тип системи керування зі зворотним зв'язком, яка може регулювати свої параметри та поведінку відповідно до змін у системі чи середовищі. Наприклад, якщо роботіві потрібно пересуватися на місцевості з різними нахилами та тертям, адаптивний контролер може змінити його швидкість і крутний момент на основі даних датчика та бажаного шляху. Перевага адаптивного керування полягає в тому, що воно може впоратися з невизначеністю та перешкодами, а також покращити продуктивність і надійність системи. Однак адаптивне керування є більш складним і потребує обчислень, ніж інші типи систем керування зі зворотним зв'язком.

### 4 Нечітке керування

Нечітке керування — це тип системи керування зі зворотним зв'язком, яка використовує нечітку логіку для обробки неточної та розпливчастої інформації. Нечітка логіка — це форма міркування, яка допускає кілька значень між істиною та хибністю, наприклад дуже низьке, низьке, середнє, високе та дуже високе. Наприклад, якщо робот повинен слідувати за людиною, нечіткий контролер може використовувати лінгвістичні правила для визначення відповідної швидкості та напрямку на основі відстані та кута до людини. Перевага нечіткого керування полягає в тому, що воно може керувати нелінійними та складними системами та включати людські знання та досвід. Однак нечітке керування може не забезпечувати точних і оптимальних рішень і може вимагати великої кількості правил і параметрів.

## 5 Керування нейронною мережею

Нейронне мережеве керування — це тип системи керування зі зворотним зв'язком, яка використовує штучні нейронні мережі для вивчення та наближення динаміки та поведінки системи. Штучні нейронні мережі — це обчислювальні моделі, які імітують структуру та функції біологічних нейронів і можуть навчатися на основі даних і адаптуватися до нових ситуацій. Наприклад, якщо роботу потрібно врівноважити полюс на його кінці, контролер нейронної мережі може вивчити зв'язок між входом і виходом системи та генерувати відповідні керуючі сигнали. Перевага керування нейронними мережами полягає в тому, що вони можуть керувати нелінійними та невизначеними системами, а також вчитися на прикладах і відгуках. Однак для керування нейронною мережею може знадобитися багато даних і навчання, і воно може не гарантувати стабільності та продуктивності.

### **1.3 Висновки щодо розділу**

В даному розділі проведено огляд принципу роботи та теорія автоматизації. Також проаналізовано матеріал щодо типів зворотного зв'язку роботизованих систем. Опанувавши дану інформацію виникла ідея розробити роботизовану систему в якій реалізовано зворотній зв'язок, тобто виконання операцій системи буде залежати від стану виконання кожного етапу функціонування.

## 2 КОНСТРУКТОРСЬКИЙ РОЗДІЛ

### 2.1 Моделювання певних деталей та вузлів роботизованого станда

Для реалізації проекту необхідно змоделювати, а потім і виготовити необхідні деталі для того щоб з них скласти дослідний стенд на якому можна буде здійснювати операції під час яких буде демонструватися зворотний зв'язок.

В якості програмного забезпечення для 3D моделювання деталей скористаємося SolidWorks.

Наведемо приклад моделювання деталі типу Лоток. Отже в режимі ескіз за допомогою прямих ліній будемо фігуру (рис. 2.1), що нагадує перевернуту літеру П. Обов'язково проставляємо розміри (оскільки дана модель буде виготовлятися в реальності).

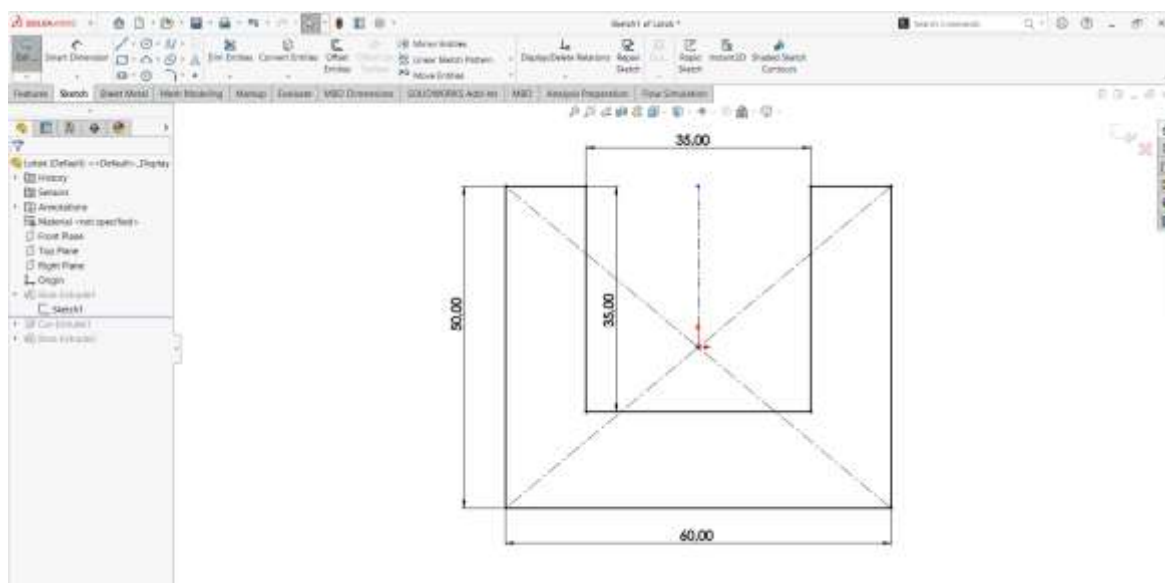


Рисунок 2.1 – Побудова ескізу лотка

Далі скористаємося функцією Boss-Extrude і «витягнемо» щойно змодельований ескіз на довжину 170 мм, даний процес представлено на рисунку 2.2. В результаті проведеної роботи отримуємо тривимірну модель лотка першого етапу.

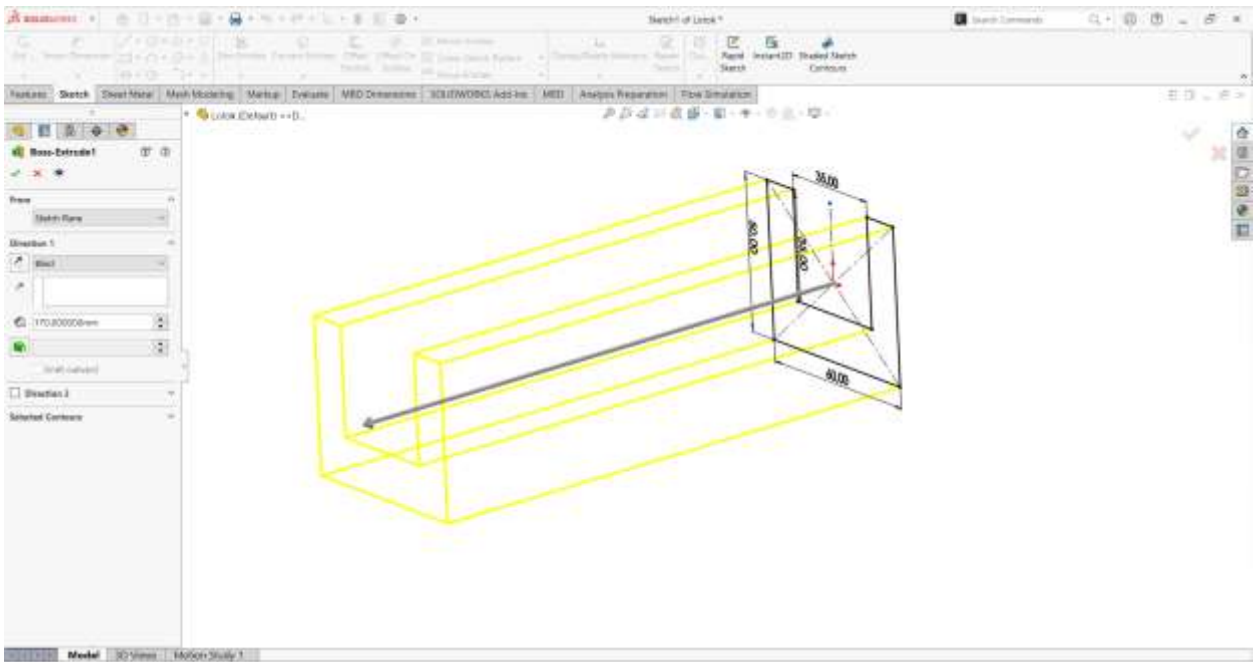


Рисунок 2.2 – Витягування ескізу лотка

Наступним етапом є моделювання отворів (призначені для кріплення до рухомої каретки роботизованого комплексу). Отже в режимі ескізу у визначених місцях (а саме посередині лотка) за допомогою примітиву коло будуємо два кола необхідного діаметру та у необхідних місцях (розмірами це диктується). Дане налагодження представлено на рис. 2.3.

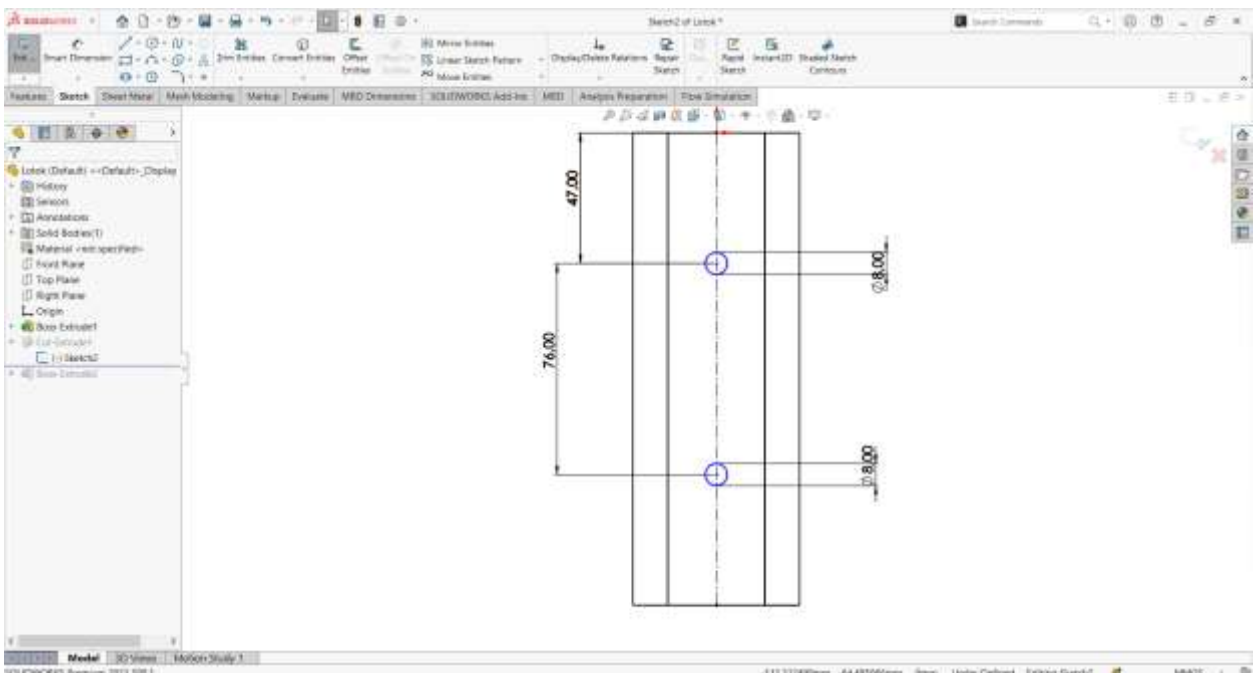


Рисунок 2.3 – Креслення двох кіл в режимі ескіз

Далі щоб утворити отвори необхідно скористатися функцією Cut-Extrude і вибравши ескіз з колами витягнути виріз через всю товщину стінки (рис. 2.4.).

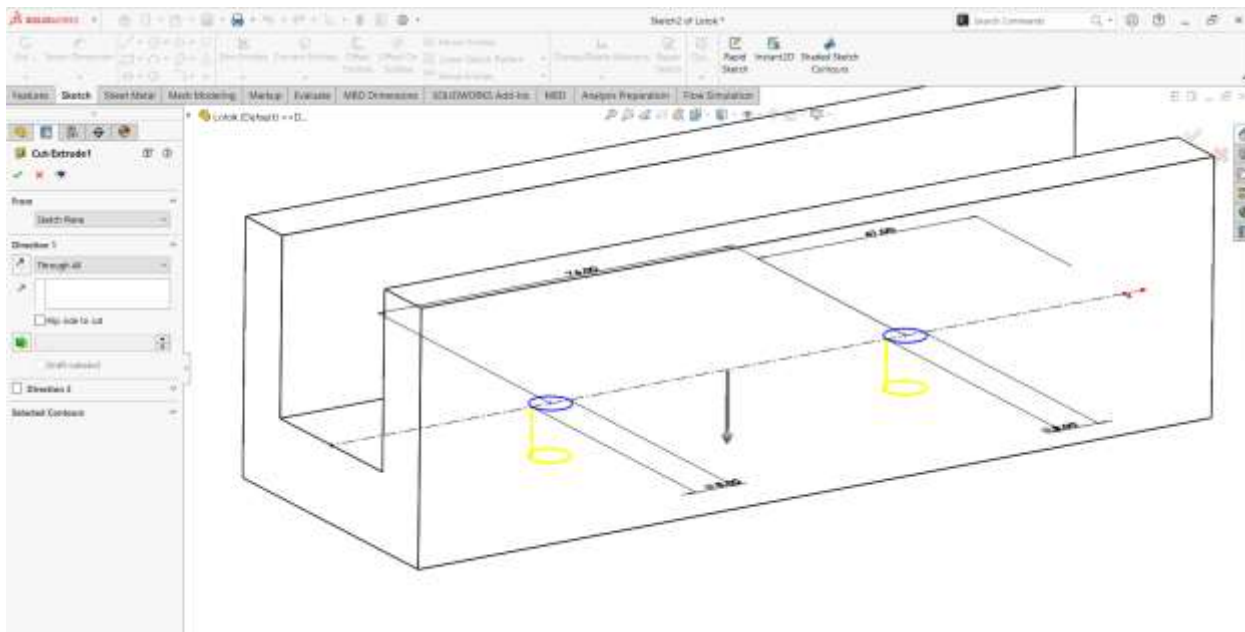


Рисунок 2.4 – Вирізання отворів

Далі оскільки лоток не ненаскрізним, то необхідно домоделювати стінку. Отже в режимі ескізу будуємо прямокутник у визначеному місці (рис. 2.5.).

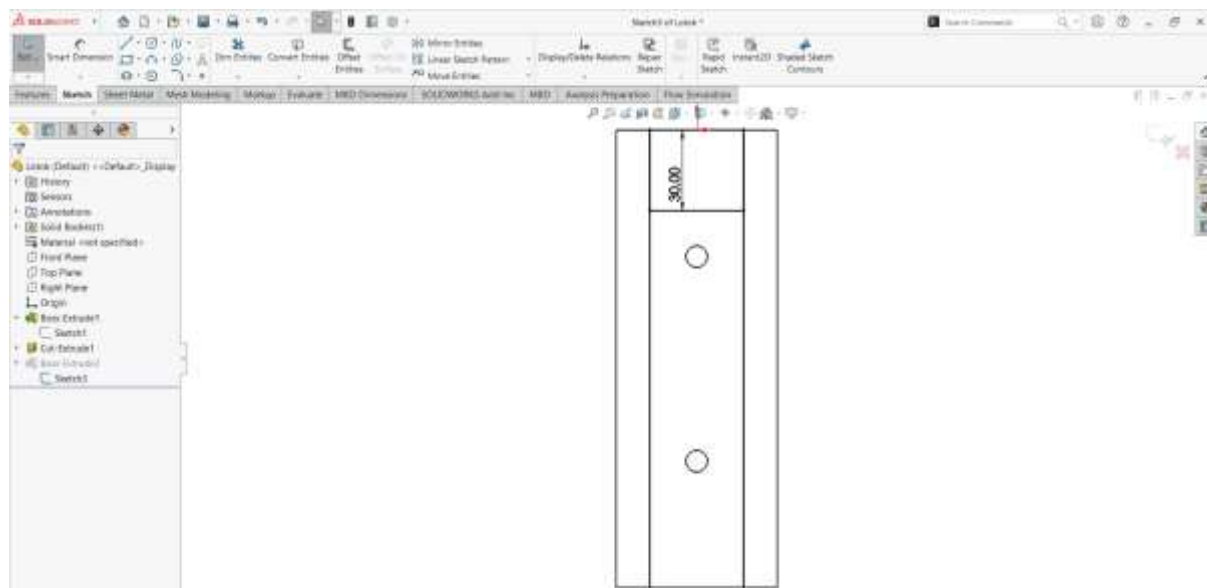


Рисунок 2.5 – Ескіз глухої стінки

Даний ескіз за допомогою функції Boss-Extrude витягуємо так щоб утворилася глуха стінка. Загалом в результаті проведеної роботи отримуємо готову 3D модель (рис. 2.6.), яка в подальшому буде виготовлятися на фрезерному верстаті з ЧПУ.

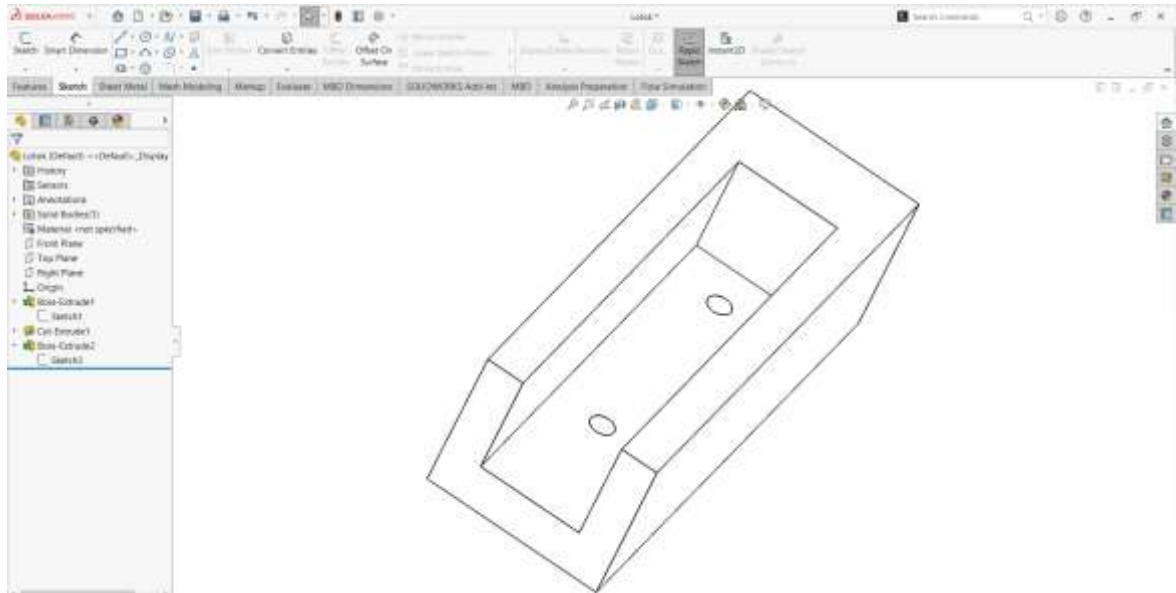


Рисунок 2.6 – Повністю змодельована деталь лоток.

Також розглянемо послідовність моделювання деталі типу гайка висувна. Отже в режимі ескіз будуємо шестигранник з віддаллю між паралельними гранями 14,5 мм (рис. 2.7).

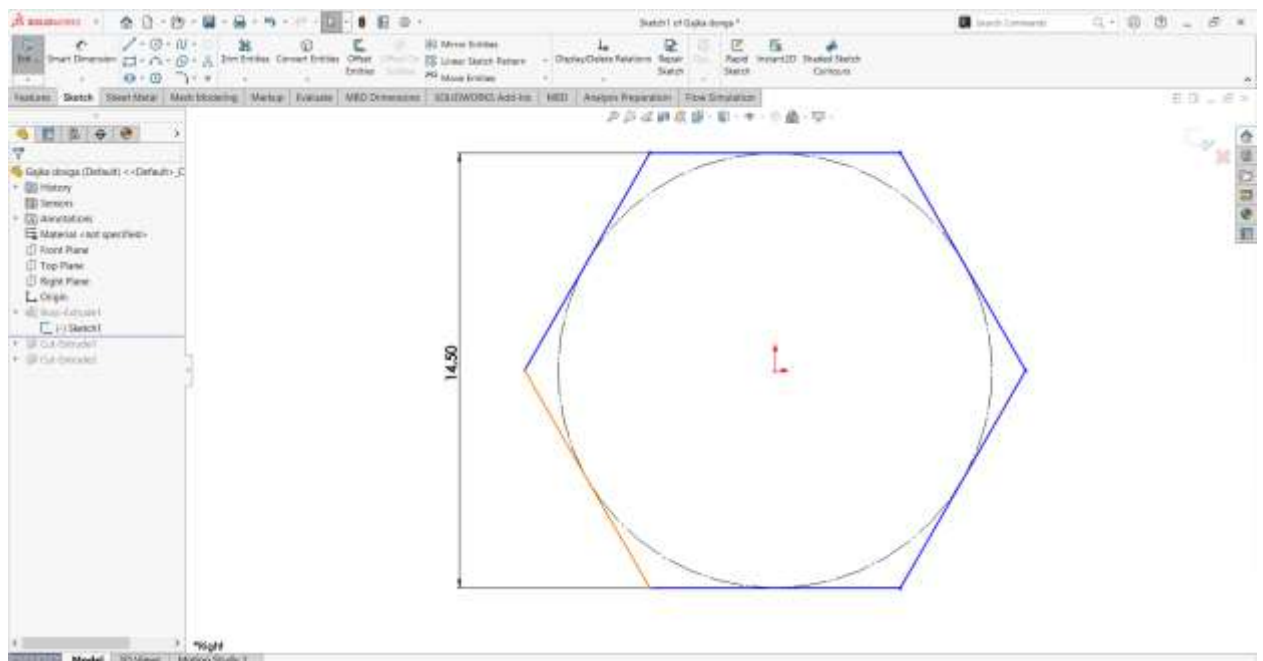


Рисунок 2.7 – Креслення шестигранника

Далі скористаємося функцією Boss-Extrude і витягнемо даний ескіз на довжину 73 мм (рис. 2.8).

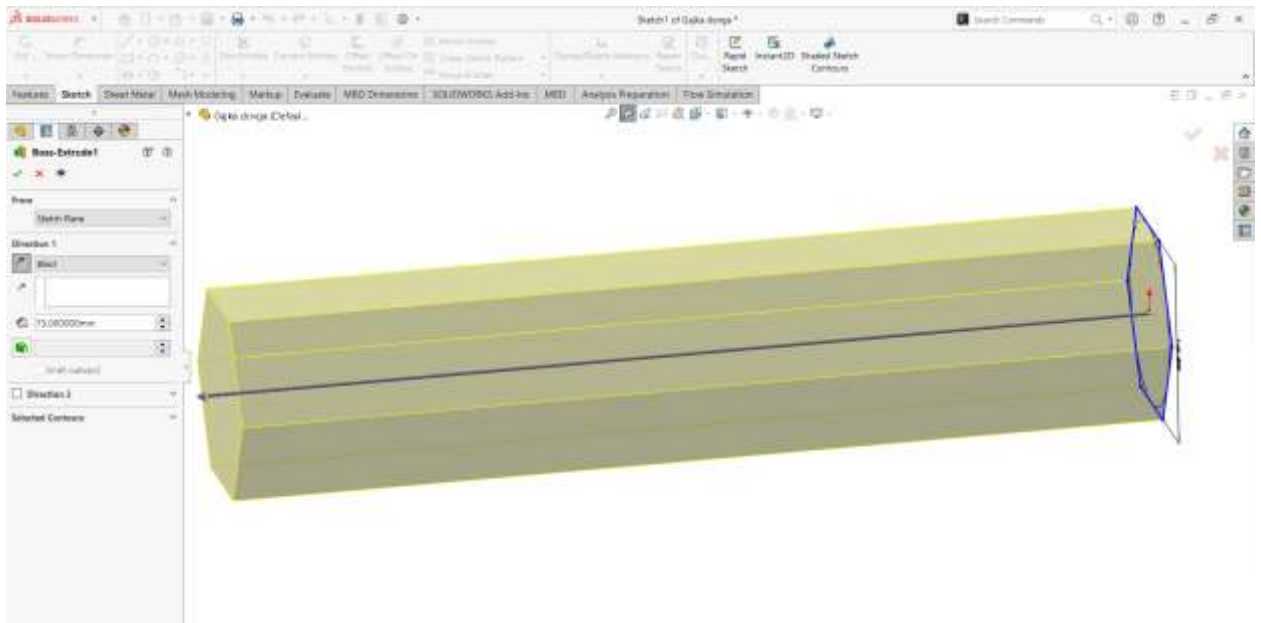


Рисунок 2.8 – Витягування шестигранника

Далі оскільки це гайка, то необхідно в режимі ескізу нарисувати отвір діаметром 5 мм (там буде нарізатися різь в реальній деталі) це представлено на рисунку 2.9.

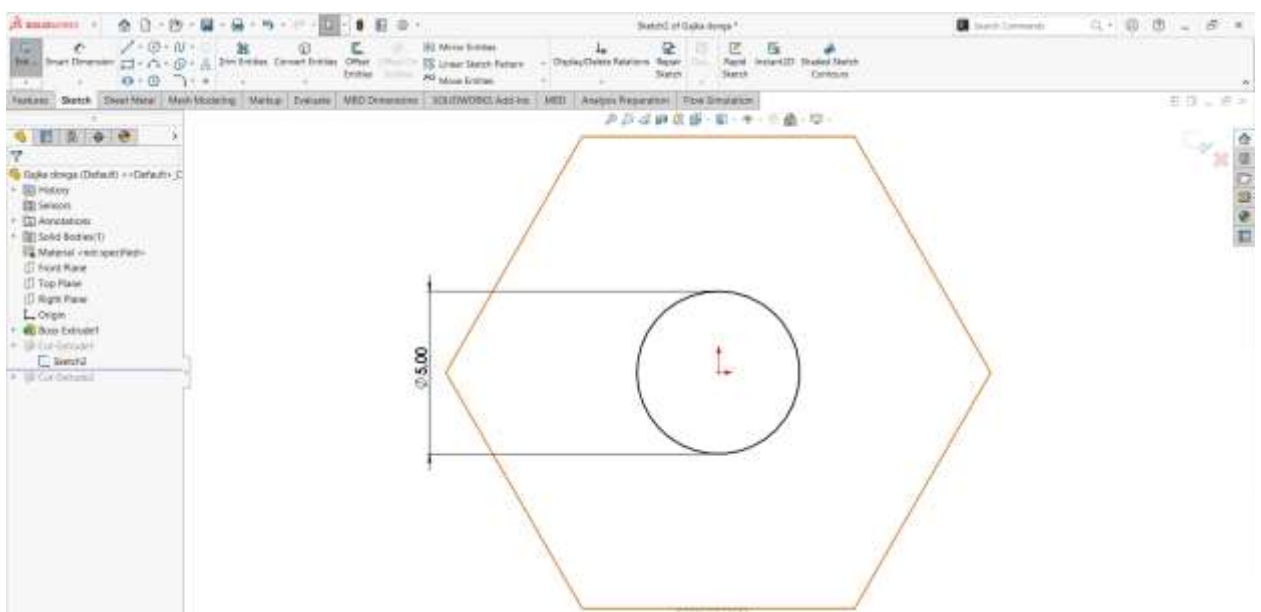


Рисунок 2.9 – Моделювання майбутнього отвору в режимі ескіз

Знову ж таки застосуємо функцію Cut-Extrude і витягнемо даний ескіз на всю довжину шестигранника (рис. 2.10).

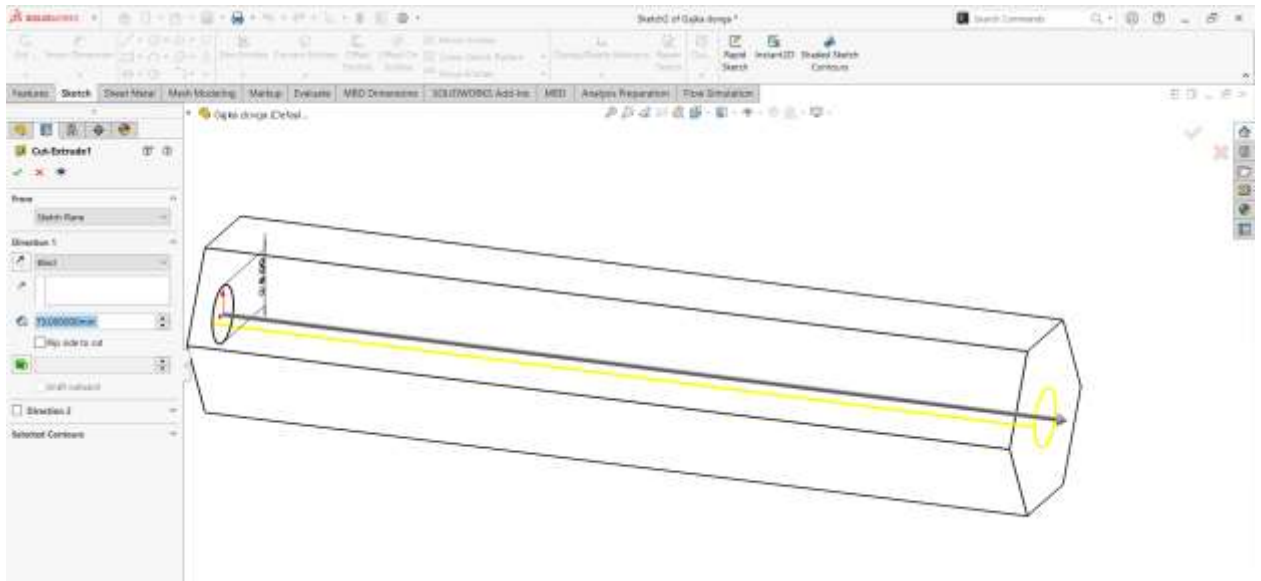


Рисунок 2.10 – Вирізання наскрізного отвору.

Далі уже вищезгаданими маніпуляціями необхідно зробити два отвори на верхній частині гайки. Здійснивши це отримуємо готову 3D модель гайки висувної. Результат роботи можна побачити на рисунку 2.11.

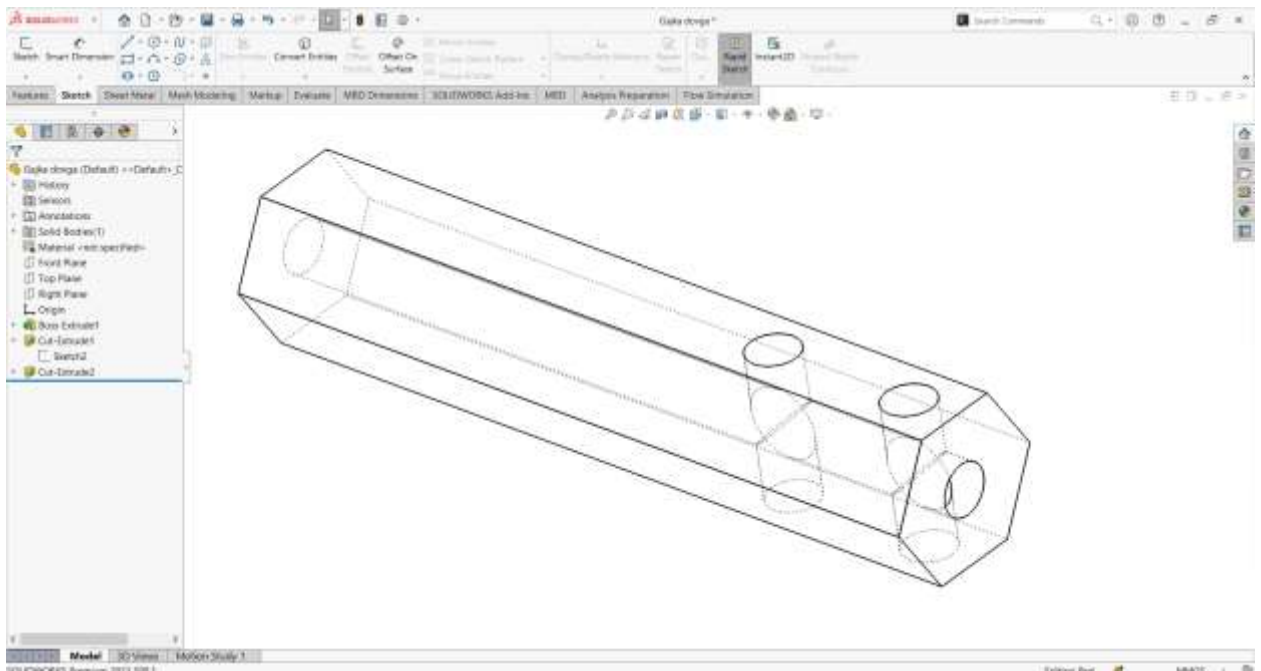


Рисунок 2.11 – Готова 3D модель гайки висувної.

Ще опишемо алгоритм побудови деталі насадка.

Побудова елементів з яких складається дана деталь формуються аналогічним чином, що й описані вище деталі.

Етап 1 – побудова основи деталі, тобто квадрат з шестигранним отвором всередині (рис. 2.12).

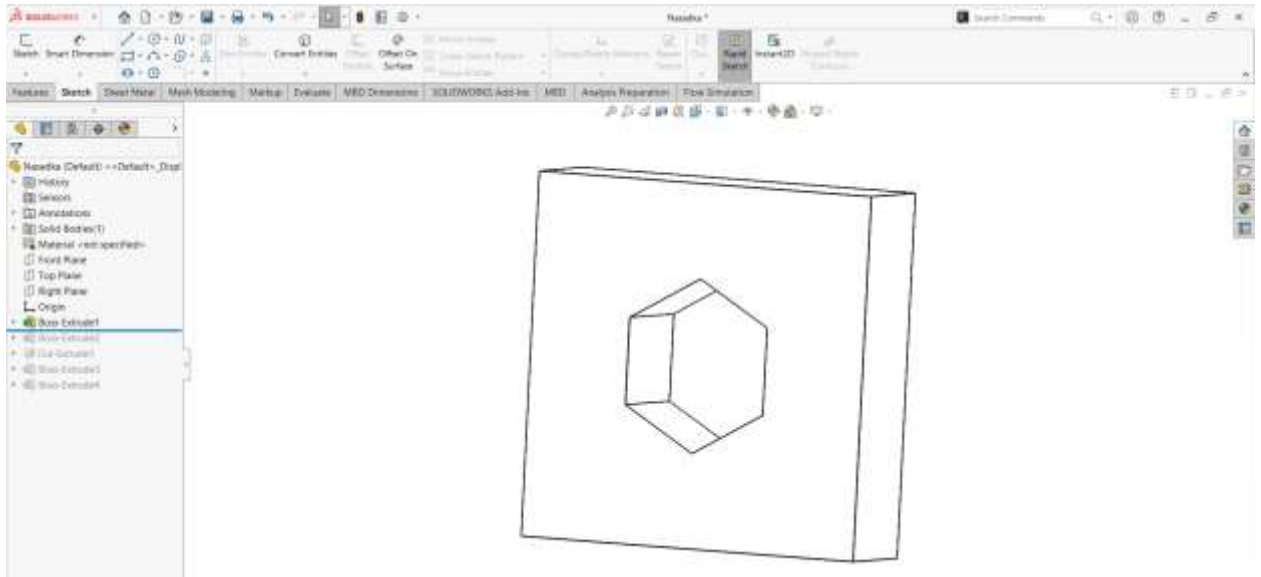


Рисунок 2.12 – етап 1 - квадрат з шестигранним отвором всередині.

Етап 2 – побудова чотирьох ніжок (рис. 2.13).

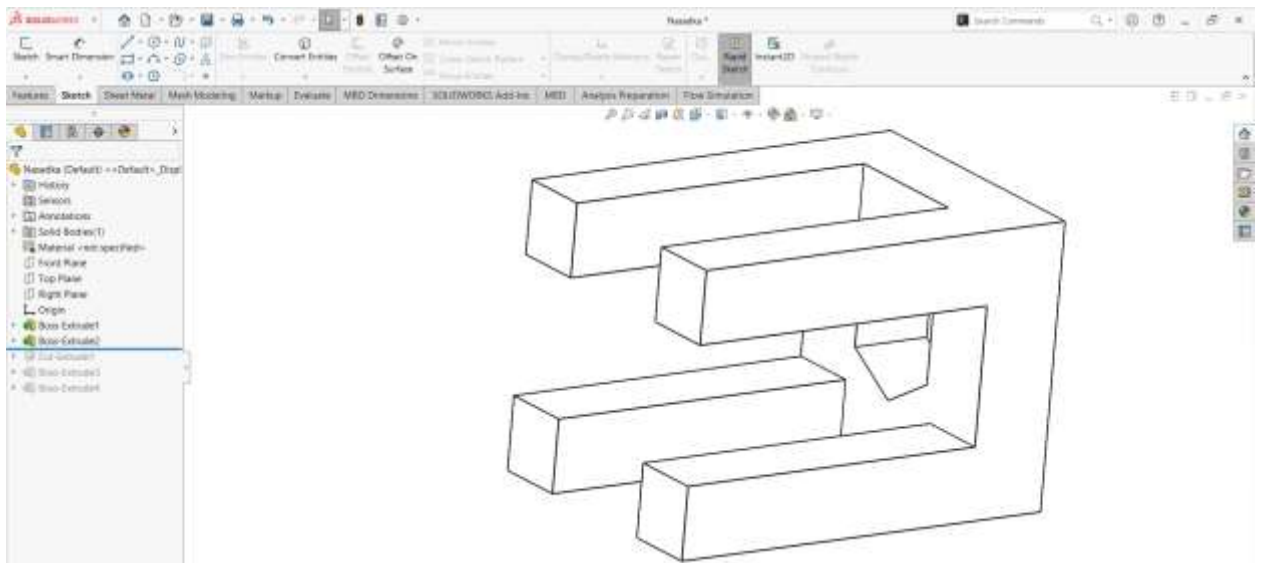


Рисунок 2.13 – етап 2 – побудова чотирьох ніжок.

Етап 3 – побудова чотирьох наскрізних отворів в ніжках (рис. 2.14).

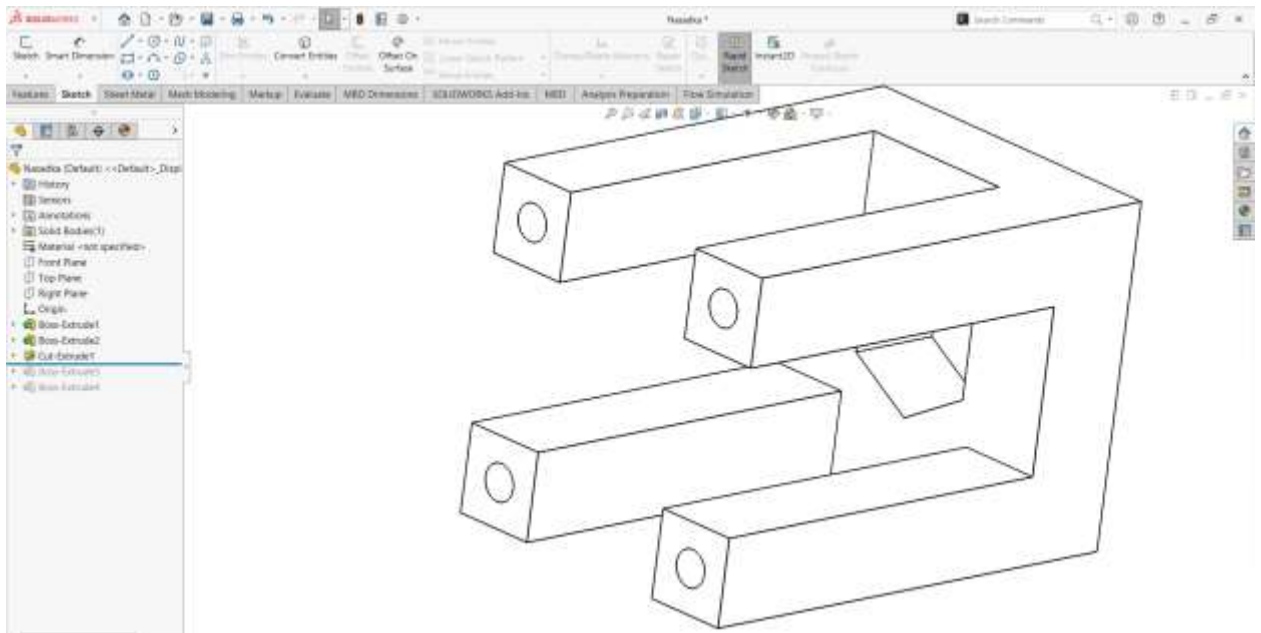


Рисунок 2.14 – етап 3 – побудова чотирьох наскрізних отворів в ніжках.

Етап 4 – побудова чотирикутного підсилювального елемента (рис. 2.15), щоб зміцнити ніжки при експлуатації.

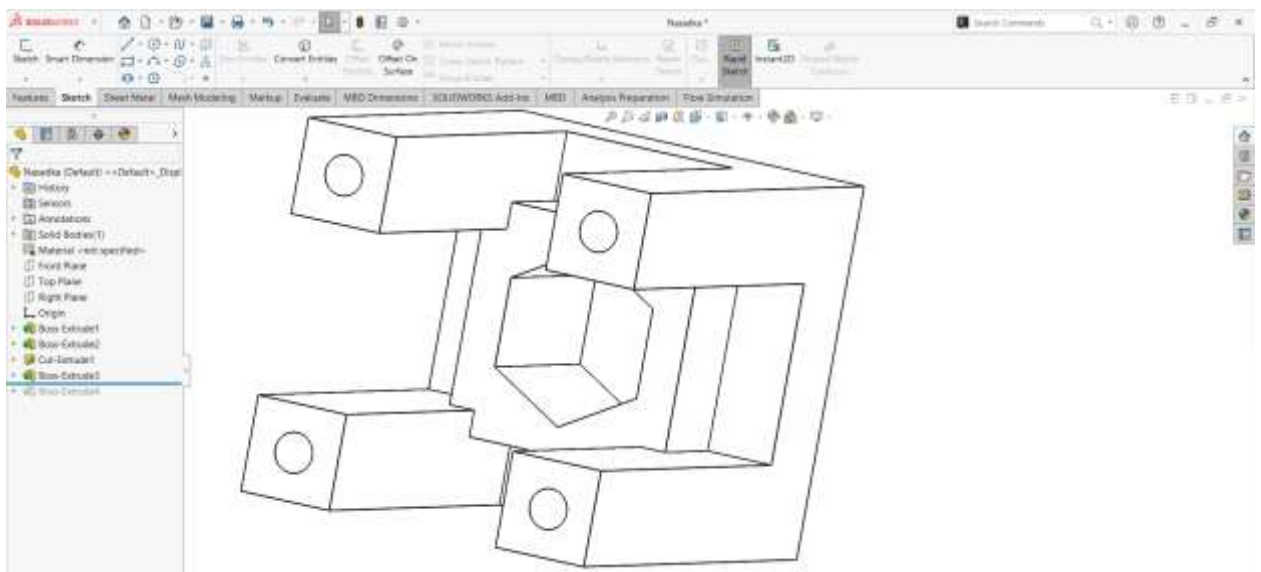


Рисунок 2.15 – етап 4 – побудова чотирикутного підсилювального елемента.

Завершальний етап 5 – побудова шестигранного виступу для подовження деталі (рис. 2.16).

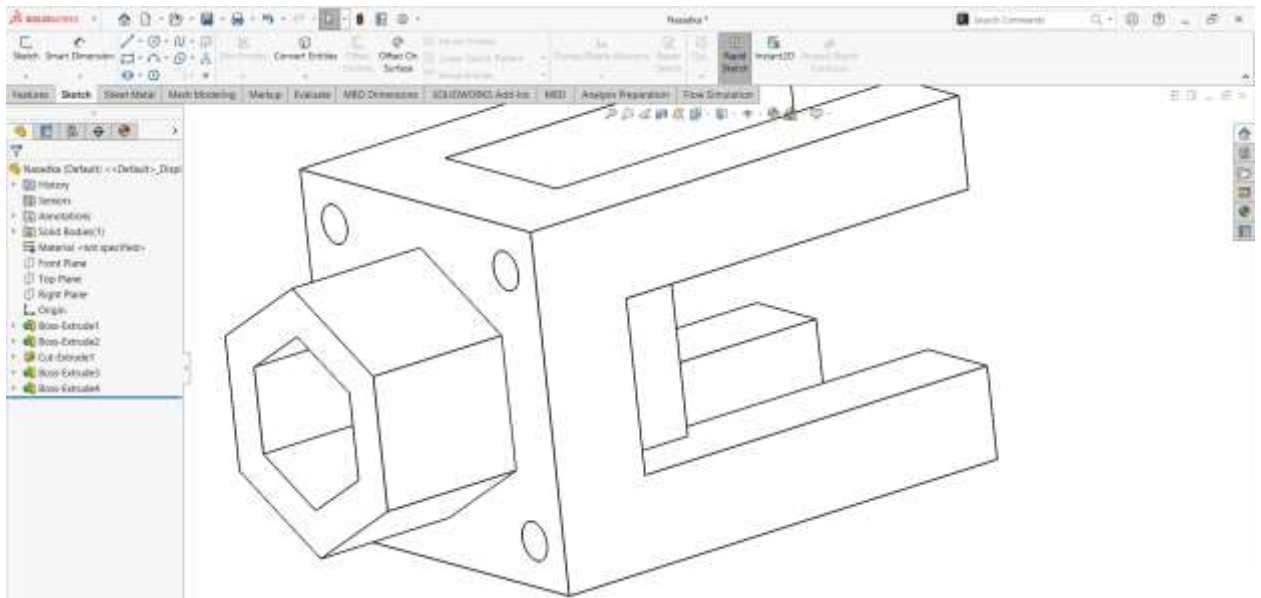


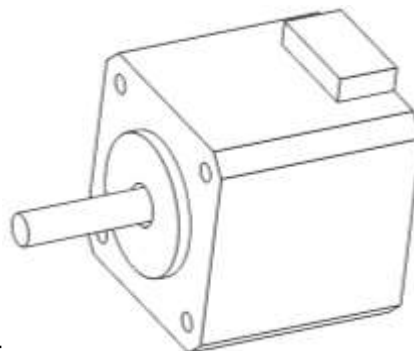
Рисунок 2.16 – етап 5 – побудова шестигранного виступу.

Подібним чином були змодельовані усі інші деталі з яких можна в програмі SolidWorks здійснити моделювання збірки роботизованого комплексу.

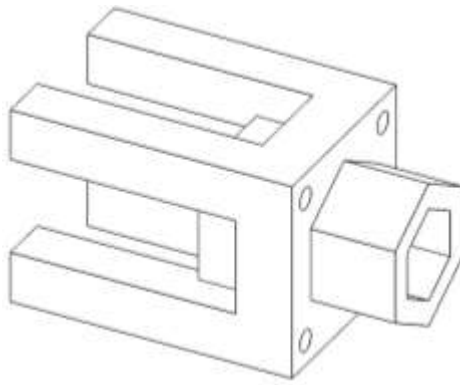
## 2.2 Моделювання збірки роботизованого станда

Приведемо приклад збірки блока роботизованого комплексу роль якого в автоматичному режимі захопити та відпустити деталь.

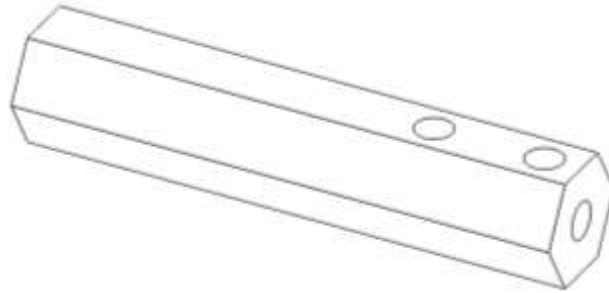
Дана збірка складається з наступних основних елементів:



- кроковий двигун NEMA 14

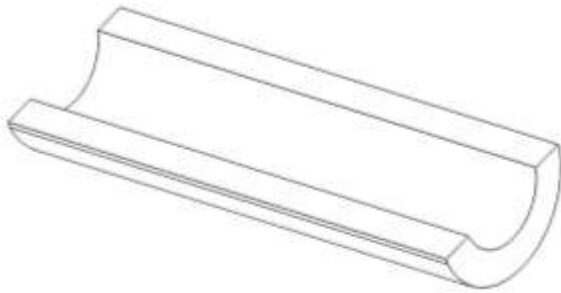


- насадка на двигун

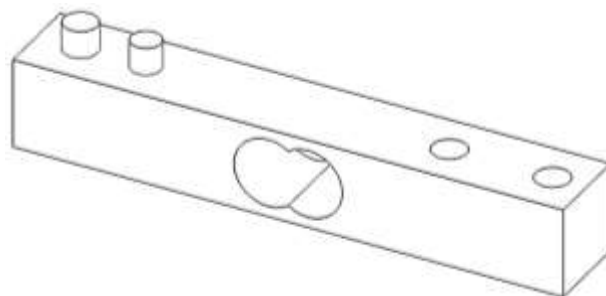
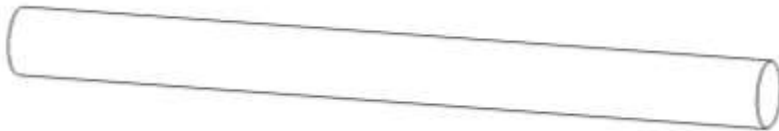


- висувна гайка

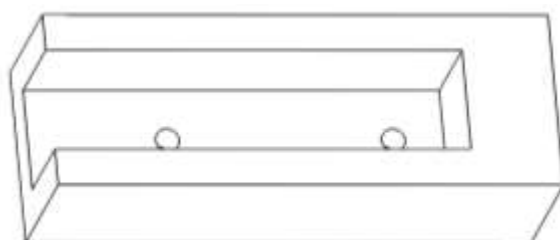
- муфта з'єднання різьбової шпильки з ротором двигуна



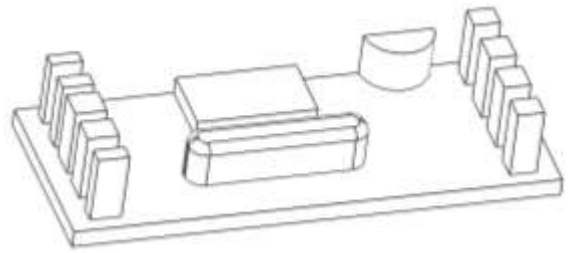
- шпилька M5, яка вкручується в висувну гайку



- тензодатчик



- ЛОТОК



- плата керування тензодатчиком HX711

Сутність здійснення збирання деталей в одну збірну одиницю полягає в тому, що поверхні одних деталей необхідно певним чином сумістити з поверхнями інших деталей. До прикладу суміщеннями можуть слугувати розміщення, як:

- Паралельність поверхонь;
- Перпендикулярність поверхонь;
- Концентричність поверхонь;
- Розташування поверхонь на певній віддалі одна від іншої;
- Розташування поверхонь під визначеним кутом одна від іншої;
- Дотичність поверхонь;
- Співпадіння поверхонь
- І багато інших видів

Отже скориставшись суміщеннями в результаті отримали наступну збірну одиницю, яка зображена на рисунку 2.17.

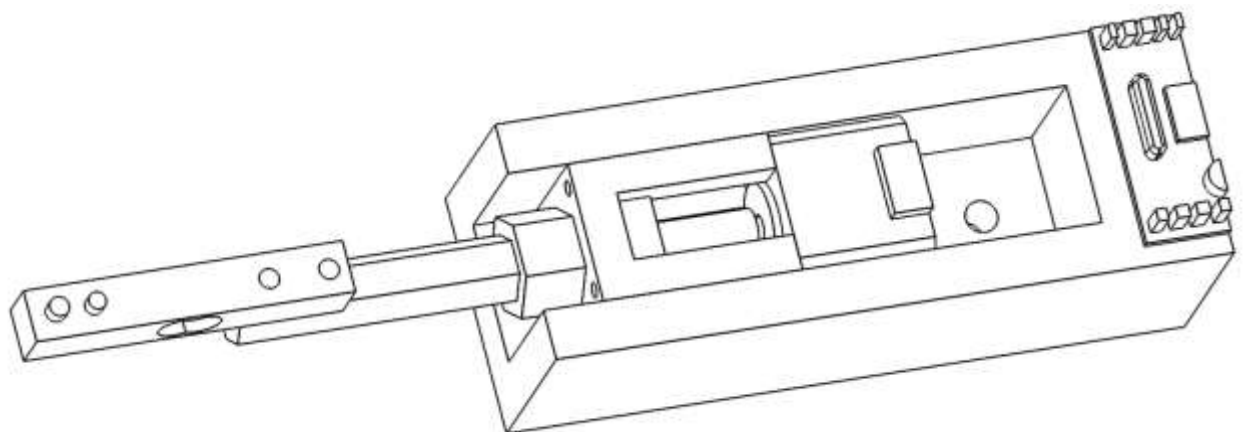


Рисунок 2.17 – Збірка висувного механізму.

В даному випадку були проведені наступні налаштування. Одна з сторін мотора стала співпалою з внутрішньої нижньою стінкою лотка. Аналогічно необхідно сумістити сторону мотора припустимо з правою стінкою лотка. Далі торцеву стінку мотора необхідно розмістити не визначеній віддалі від торцевої поверхні лотка.

Потім торцеву поверхню шпильки необхідно сумістити з торцевою поверхнею ротора двигуна, а циліндричну поверхню шпильки і циліндричну поверхню ротора двигуна розмістити концентрично одна відносно одної.

Далі торцеві поверхні ніжок насадки необхідно сумістити з торцевою поверхнею двигуна (насадка стає на двигун імітація що насадка прикрутилася гвинтами до двигуна).

Плоску поверхню висувної гайки сумістити з визначеною внутрішньою поверхню шестигранного отвору насадки. Таку ж операцію необхідно провести і з іншою поверхнею як гайки так і насадки. Таким чином гайка має змогу рухатися зворотно поступально в шестигранному отворі насадки.

На тензодатчику і висувній гайці є отвори. Відповідно необхідно вибрати отвори на висувній гайці і на даному тензодатчику відповідно розташували їх концентрично. Також необхідно вибрати плоску нижню поверхню на тензодатчику і плоску поверхню на висувній гайці і сумістити їх. Таким чином відбулася імітація прикручування тензодатчика до висувної гайки за допомогою болтів з гайками.

Відповідно подібні дії потрібно зробити і з іншими елементами даної збірки щоб вона склалася.

Загалом весь роботизований комплекс складається з набагато більшої кількості деталей, які аналогічно потрібно розмістити впорядковано одна відносно іншої. В загальній збірці є також наступні деталі: кроковий двигун NEMA 17, ультразвуковий дальномір, корпус, напрямні, ходовий гвинт, маточка гайка, рухома платформа, драйвери, підшипники, муфти, плати керування, блоки живлення і т.п. Результат збірки наведено на рисунку 2.18.

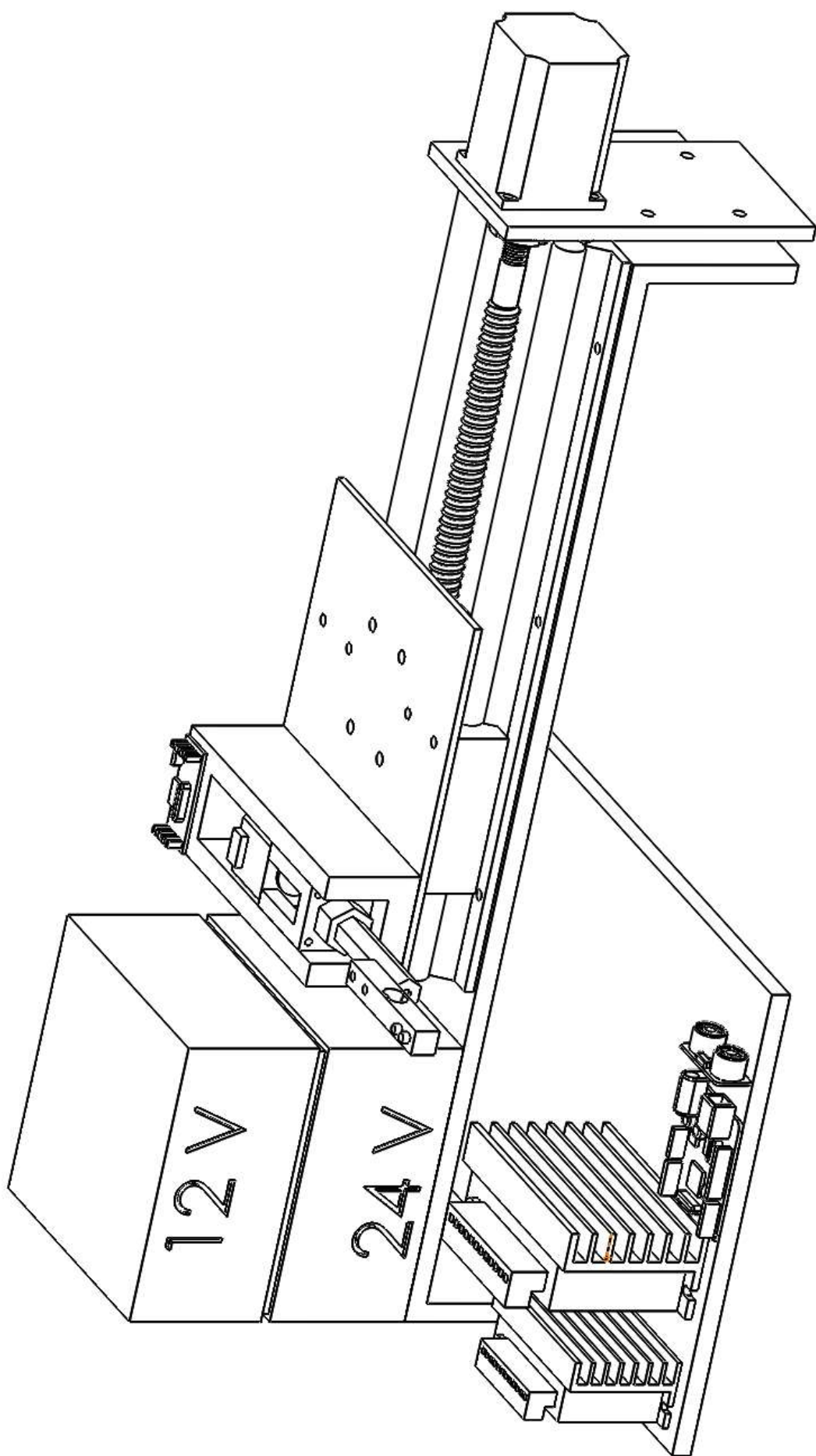


Рисунок 2.18 – 3D модель збірки роботизованого комплексу з зворотним зв'язком

Дане моделювання збірки необхідне для того щоб переконатися в правильності моделювання усіх компонентів і що коли їх виготовити в реальності, то вони складуться.

### **2.3 Висновки щодо розділу**

В даному розділі здійснено моделювання усіх деталей роботизованого комплексу та виконано їх збірку в один функціональний механізм. Переконавшись в тому, що усі елементи добре розташовуються один відносно одного і не перетинаються – то можна переходити до етапу в якому можна проектувати технологію їх виготовлення в реальному житті.

## 3 ТЕХНОЛОГІЧНИЙ РОЗДІЛ

### 3.1. Налаштування обробки та виготовлення деталі насадка за допомогою 3D друку

Щоб виготовити деталь насадка варто скористатися 3D друком оскільки деталь є корпусною і найлегше її буде виготовити саме цим способом.

Перед виготовленням самої деталі необхідно провести налаштування її виготовлення в програмному забезпеченні Creality Slicer v 4.8. Вибір даного ПЗ зумовлений застосуванням 3D принтера Creality Ender 3, тобто використання рекомендованого ПЗ з обладнанням вітається.

Отже, на уявний стіл 3D принтера виставляємо раніше спроектовану деталь насадка (рис. 3.1).

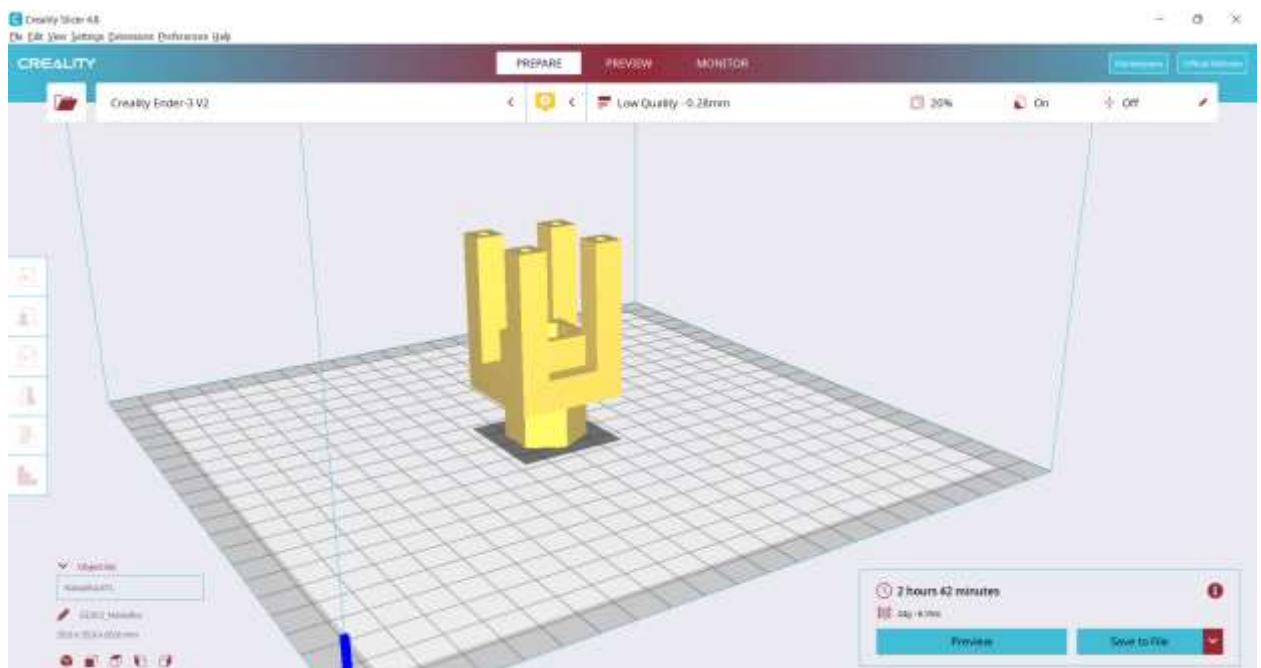


Рисунок 3.1 – Налаштування виготовлення деталі для 3D принтера.

Необхідно провести певні налаштування для максимально необхідної якості деталі. Отже:

- матеріал друку – PLA пластик;
- діаметр дроту пластика – 1,75 мм;

- температура розігріву пластику при друці – 200°C;
- температура столу при друці – 60°C;
- товщина шару – 0,28мм;
- відсоток заповнення матеріалом – 20%;
- застосування «підтримки» – так.

Натиснувши кнопку «Slice» програма автоматично здійснила налагодження друку.

Представимо фрагмент програмного коду на виготовлення лотка:

M140 S60

M105

M190 S60

M104 S200

M105

M109 S200

M82 ;absolute extrusion mode

G28 ;Home

G92 E0 ;Reset Extruder

G1 Z2.0 F3000 ;Move Z Axis up

G1 X10.1 Y20 Z0.28 F5000.0 ;Move to start position

G1 X10.1 Y200.0 Z0.28 F1500.0 E15 ;Draw the first line

G1 X10.4 Y200.0 Z0.28 F5000.0 ;Move to side a little

G1 X10.4 Y20 Z0.28 F1500.0 E30 ;Draw the second line

G92 E0 ;Reset Extruder

G1 Z2.0 F3000 ;Move Z Axis up

G92 E0

G92 E0

G1 F1500 E-4

;LAYER\_COUNT:232

;LAYER:0

M107

G0 F6000 X84.732 Y84.844 Z0.28

;TYPE:SKIRT

G1 F1500 E0

G1 F1200 X84.844 Y84.732 E0.00738

G1 X85.535 Y84.098 E0.05104

G1 X86.277 Y83.524 E0.09472

G1 X87.066 Y83.017 E0.13839

G1 X87.894 Y82.578 E0.18203

G1 X88.758 Y82.211 E0.22574

G1 X89.649 Y81.919 E0.2694

G1 X90.562 Y81.705 E0.31307

G1 X91.49 Y81.568 E0.35675

G1 X92.251 Y81.516 E0.39227

G1 X92.636 Y81.506 E0.4102

G1 X93.179 Y81.5 E0.43549

G1 X126.676 Y81.5 E1.99525

G1 X127.535 Y81.511 E2.03525

G1 X128.471 Y81.564 E2.0789

G1 X129.399 Y81.697 E2.12256

G1 X130.313 Y81.909 E2.16625

G1 X131.206 Y82.197 E2.20994

G1 X132.07 Y82.561 E2.25359

G1 X132.901 Y82.997 E2.29729

G1 X133.691 Y83.502 E2.34095

G1 X134.435 Y84.072 E2.38459

G1 X135.156 Y84.732 E2.43011

G1 X135.268 Y84.844 E2.43748

G1 X135.902 Y85.535 E2.48115  
G1 X136.476 Y86.277 E2.52483  
G1 X136.983 Y87.066 E2.5685  
G1 X137.422 Y87.894 E2.61214  
G1 X137.789 Y88.758 E2.65585  
G1 X138.081 Y89.649 E2.69951  
G1 X138.295 Y90.562 E2.74318  
G1 X138.432 Y91.49 E2.78686  
G1 X138.483 Y92.221 E2.82098  
G1 X138.494 Y92.613 E2.83924  
G1 X138.5 Y93.136 E2.86359  
G1 X138.499 Y126.8 E4.43113  
G1 X138.489 Y127.529 E4.46508  
G1 X138.436 Y128.465 E4.50873  
G1 X138.304 Y129.394 E4.55242  
G1 X138.093 Y130.308 E4.5961  
G1 X137.805 Y131.2 E4.63975  
G1 X137.442 Y132.065 E4.68343  
G1 X137.006 Y132.896 E4.72713  
G1 X136.502 Y133.686 E4.77076  
G1 X135.931 Y134.431 E4.81447  
G1 X135.268 Y135.156 E4.86022  
G1 X135.156 Y135.268 E4.86759  
G1 X134.465 Y135.902 E4.91126  
G1 X133.723 Y136.476 E4.95494  
G1 X83.396 Y87.209 E19.20076  
G1 X83.9 Y86.439 E19.24361  
G1 X84.469 Y85.715 E19.28649

.....

Варто зауважити, що ПЗ Creality Slicer також показує час на виконання роботи. В даному випадку це 2 години та 42 хвилини.

Отже на рисунку 3.2 можна побачити результат роботи.

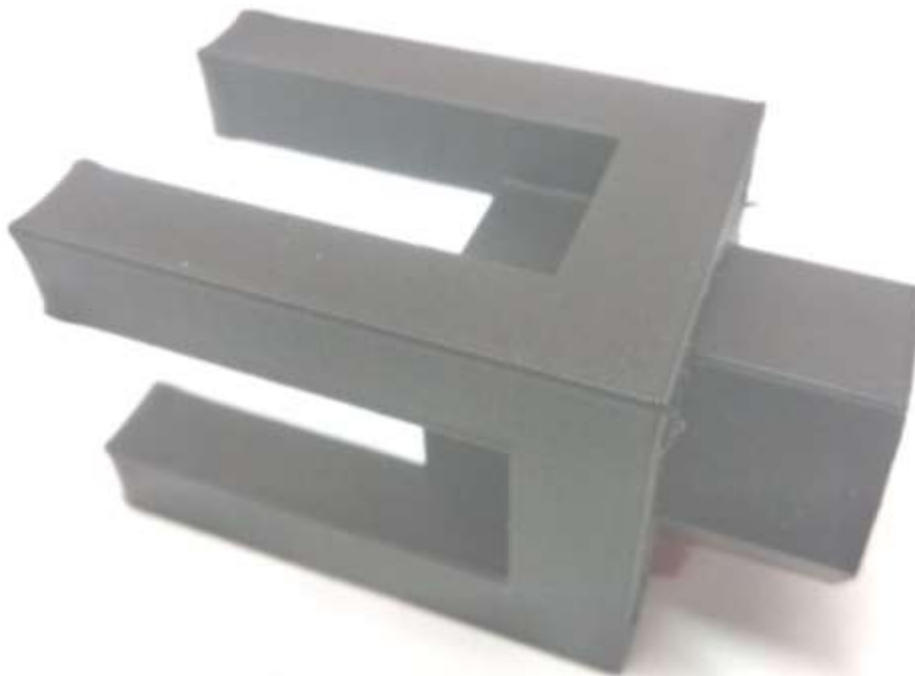


Рисунок 3.2 – Деталь насадка виготовлена на 3D принтері.

Деталь вийшла такою як і при моделюванні, тому її можна використовувати в подальшому.

### **3.2 Налагодження обробки та виготовлення деталі лоток на фрезерному верстаті з ЧПУ**

Деталь типу лоток будемо виготовляти на 3-х осьовому фрезерному верстаті з числовим програмним управлінням.

Для налагодження роботи застосуємо програмне забезпечення FeatureCAM.

Отже, в даному програмному забезпеченні відкриваємо файл що раніше змодельовали. Програма відразу запитує який метод обробки різанням та в яких розмірних одиницях буде виконуватися робота (рис. 3.3).

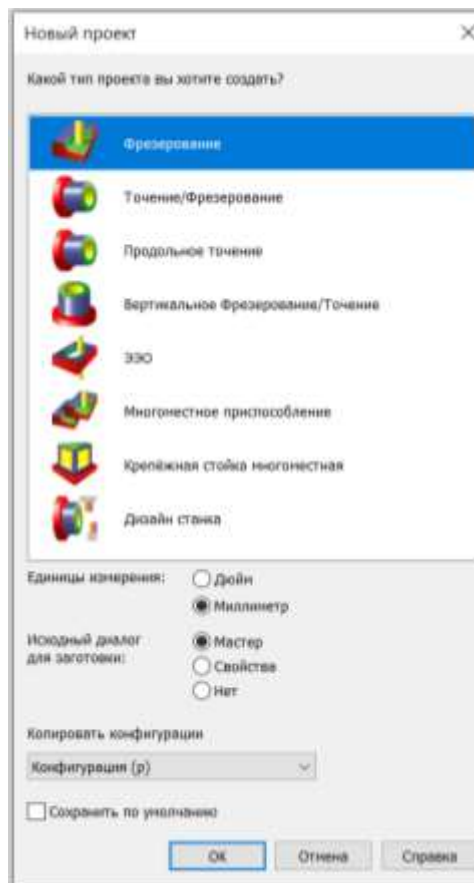


Рисунок 3.3 – Вибір типу обробки різанням та в яких розмірних одиницях.

Далі програма запитує чи вірно буде розташована деталь відносно осей Z (рис. 3.4) X Y.

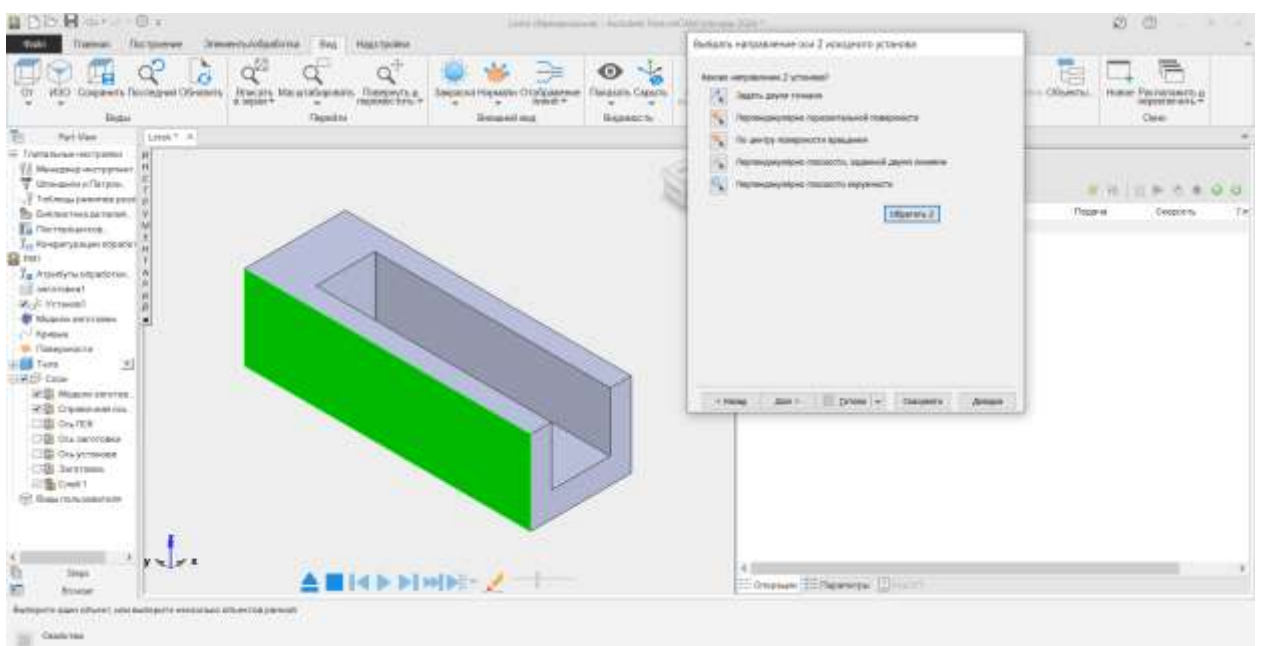


Рисунок 3.4 – Розташування деталі відносно осі Z.

Відповідно коли деталь розташована не зовсім коректно по відношенню до осей верстата, то є можливість це виправити.

Далі програмі необхідно вказати форму та розмір заготовки. В нашому випадку форма – паралелепіпед, а розміри – такі як габаритний розмір самої деталі (рис. 3.5).

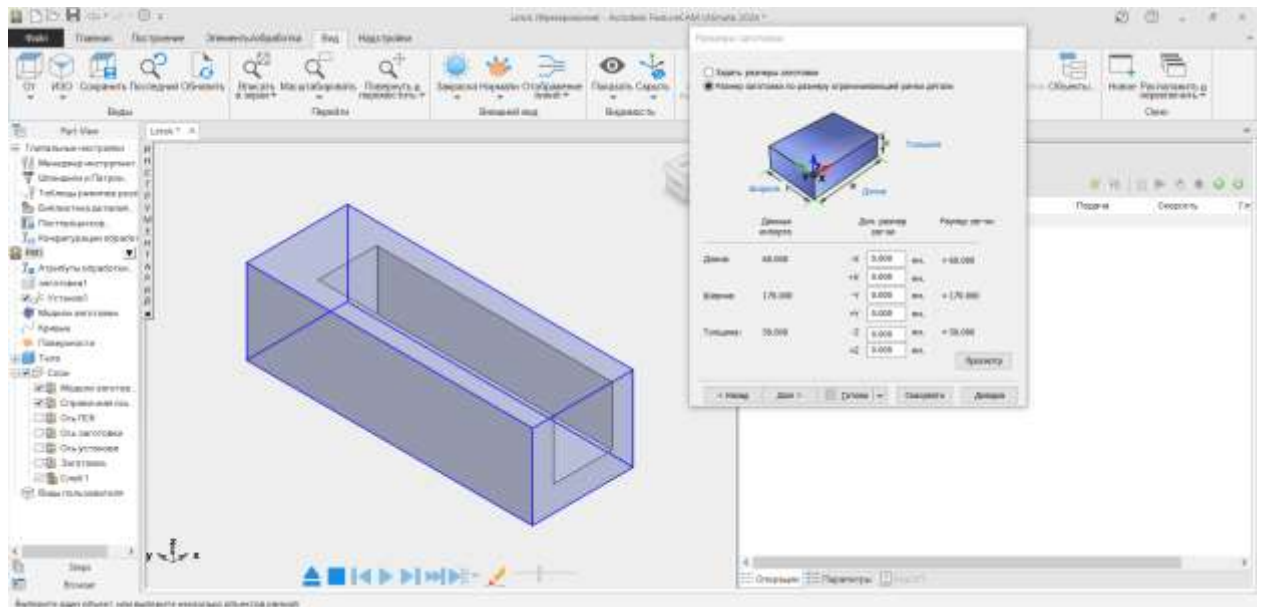


Рисунок 3.5 – Форма та розмір заготовки.

Далі необхідно вказати початкову точку відліку фрези (рис. 3.6). в нашому випадку виберемо в середині на поверхні заготовки

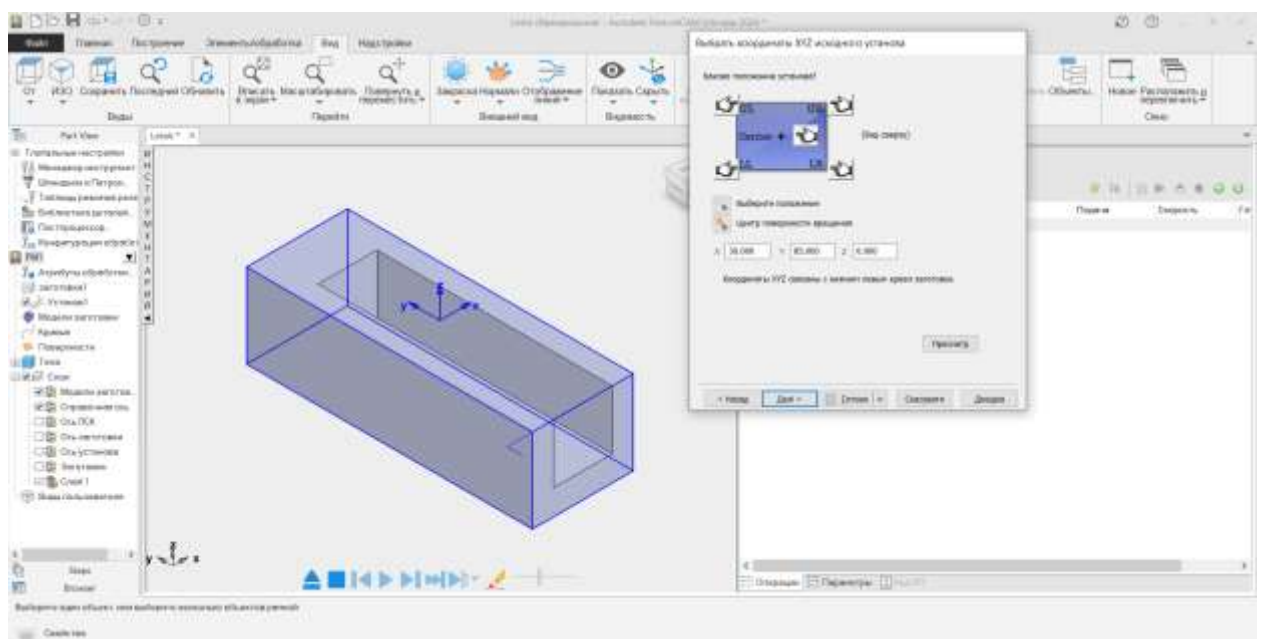


Рисунок 3.6 – Нуль фрези.

Далі в автоматичному режимі програма розпізнає (рис. 3.7) усі елементи які треба обробити (карман, отвори, загальну обрізку від цілого шматка заготовки, і т.д.).

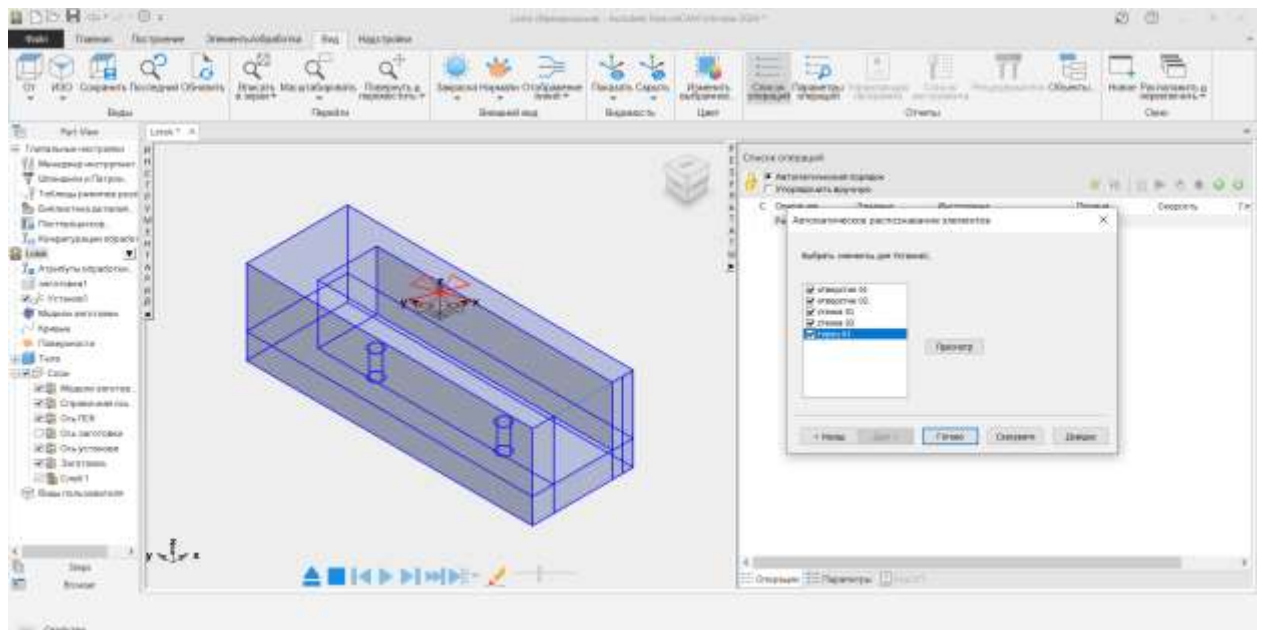


Рисунок 3.7 – Розпізнавання елементів для обробки.

В результаті програма далі підбере необхідні інструменти та режими різання, щоб виготовити деталь (рис. 3.8).

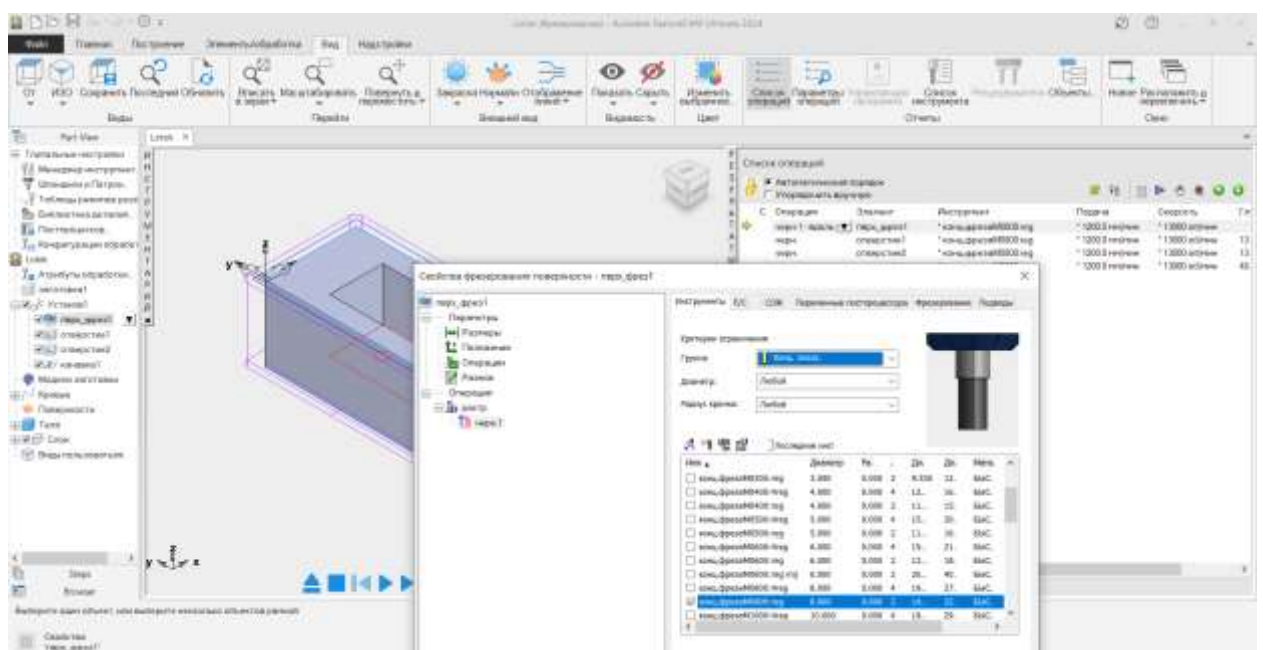


Рисунок 3.8 – Підбір інструментів та режимів обробки.

Здійснивши усі налагодження можна переглянути майбутній результат роботи (рис. 3.9).

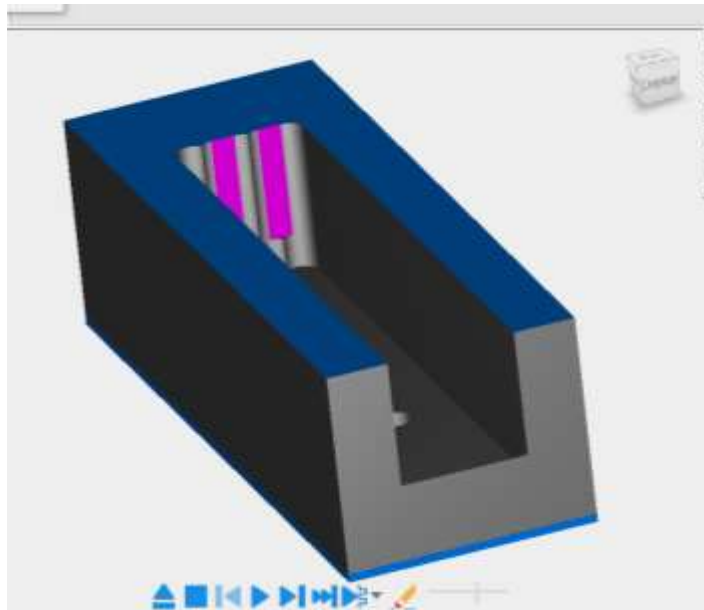


Рисунок 3.9 – Симуляція процесу обробки.

Відповідно при перегляді результату обробки можна вносити правки для отримання бажаного результату. Коли все добре можна згенерувати G-код згідно якого верстат зробить роботу. Фрагмент G-коду представлено нижче:

```
N35 G00 G92 X0 Y0 Z0

N40 ( ROUGH1 Y PARALLEL/-90 ПВРХ_ФРЕЗ1 )
N45 M6 T1
N50 G94
N55 X18.8 Y-34.1 S13000 M03
N60 Z3.0
N65 G01 Z-5.0 F600.
N70 Y-174.0 F1200.
N75 G00 Z3.0
N80 X24.4
N85 G01 Z-5.0 F600.
N90 Y-34.1 F1200.
N95 G00 Z3.0
N100 X30.0
N105 G01 Z-5.0 F600.
N110 Y-174.0 F1200.
N115 G00 Z3.0
N120 X35.6
N125 G01 Z-5.0 F600.
N130 Y-34.1 F1200.
N135 G00 Z3.0
N140 X41.2
N145 G01 Z-5.0 F600.
```

N150 Y-174.0 F1200.  
N155 G00 Z10.0  
N160 X18.8 Y-34.1  
N165 Z-2.0  
N170 G01 Z-10.0 F600.  
N175 Y-174.0 F1200.  
N180 G00 Z-2.0  
N185 X24.4  
N190 G01 Z-10.0 F600.  
N195 Y-34.1 F1200.  
N200 G00 Z-2.0  
N205 X30.0  
N210 G01 Z-10.0 F600.  
N215 Y-174.0 F1200.  
N220 G00 Z-2.0  
N225 X35.6  
N230 G01 Z-10.0 F600.  
N235 Y-34.1 F1200.  
N240 G00 Z-2.0  
N245 X41.2  
N250 G01 Z-10.0 F600.  
N255 Y-174.0 F1200.  
N260 G00 Z10.0  
N265 X18.8 Y-34.1  
N270 Z-7.0  
N275 G01 Z-15.0 F600.  
N280 Y-174.0 F1200.  
N285 G00 Z-7.0  
N290 X24.4  
N295 G01 Z-15.0 F600.  
N300 Y-34.1 F1200.  
N305 G00 Z-7.0  
N310 X30.0  
N315 G01 Z-15.0 F600.  
N320 Y-174.0 F1200.  
N325 G00 Z-7.0  
N330 X35.6  
N335 G01 Z-15.0 F600.  
N340 Y-34.1 F1200.  
N345 G00 Z-7.0  
N350 X41.2  
N355 G01 Z-15.0 F600.  
N360 Y-174.0 F1200.  
N365 G00 Z10.0  
N370 X18.8 Y-34.1  
N375 Z-12.0  
N380 G01 Z-20.0 F600.  
N385 Y-174.0 F1200.  
N390 G00 Z-12.0  
N395 X24.4  
N400 G01 Z-20.0 F600.  
N405 Y-34.1 F1200.  
N410 G00 Z-12.0  
N415 X30.0  
N420 G01 Z-20.0 F600.  
N425 Y-174.0 F1200.  
N430 G00 Z-12.0  
N435 X35.6  
N440 G01 Z-20.0 F600.  
N445 Y-34.1 F1200.  
N450 G00 Z-12.0  
N455 X41.2  
N460 G01 Z-20.0 F600.  
N465 Y-174.0 F1200.

```
N470 G00 Z10.0
N475 X18.8 Y-34.1
N480 Z-17.0
N485 G01 Z-25.0 F600.
N490 Y-174.0 F1200.
N495 G00 Z-17.0
N500 X24.4
N505 G01 Z-25.0 F600.
N510 Y-34.1 F1200.
N515 G00 Z-17.0
N520 X30.0
N525 G01 Z-25.0 F600.
N530 Y-174.0 F1200.
N535 G00 Z-17.0
N540 X35.6
N545 G01 Z-25.0 F600.
N550 Y-34.1 F1200.
N555 G00 Z-17.0
N560 X41.2
N565 G01 Z-25.0 F600.
```

.....

Далі з цим кодом підходимо до верстата та робимо наступні дії:

- кріпимо заготовку на стіл верстата;
- кріпимо в шпindelь верстата інструмент;
- завантажуюємо G-код обробки деталі;
- виставляємо фрезу в нульове положення;
- запускаємо роботу верстата.

В результаті верстат вирізав деталь лоток яка представлена на рис. 3.10.

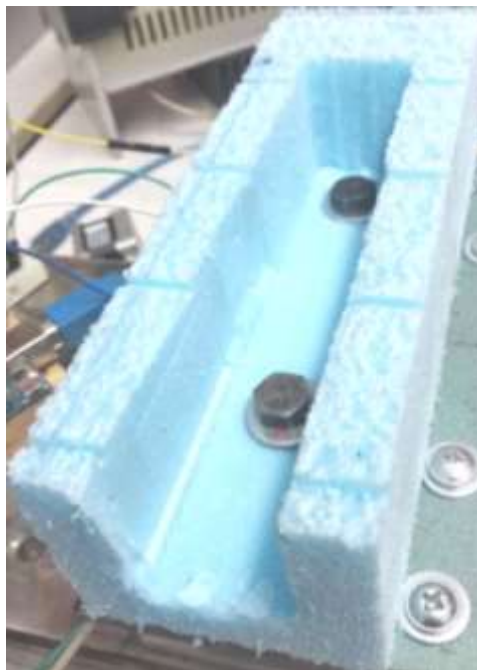


Рисунок 3.10 – Виготовлена деталь лоток.

Деталь вийшла такою як моделювалася, тому її можна використовувати в подальшій роботі щодо проекту.

### **3.3 Налагодження техпроцесу та виготовлення деталі висувна гайка класичним обробленням**

Деталь висувна гайка буде виготовлятися на універсальних верстатах з застосуванням класичних інструментів. Отже представимо техпроцес на виготовлення даної деталі:

- від загального шматка заготовки (прут шестигранник) відрізаємо заготовку висувної гайки;
- в токарному верстаті з обох боків торцюємо заготовку в розмір;
- зацентруємося на торці заготовки;
- просверджуємо наскрізь отвір свердлом діаметром 4,5 мм;
- нарізаємо внутрішню різь М5 на довжину 20 мм;
- закріплюємо заготовку на свердлильному верстаті та просверджуємо в радіальному напрямку отвори діаметром 6 мм розташовані згідно креслення деталі.

В результаті проведених робіт отримуємо деталь яка представлена на рисунку 3.11.



Рисунок 3.11 – Виготовлена деталь висувна гайка.

Виготовлена деталь вийшла згідно розмірів і тому буде далі використовуватися

### **3.3 Вибір електронних компонентів для роботи роботизованого комплексу з зворотнім зв'язком**

Крім виготовлення необхідних деталей варто також підібрати покупні електронні компоненти. Отже приведемо обрані компоненти з їх технічними характеристиками.

Кроковий двигун NEMA 14 (рис. 3.12).

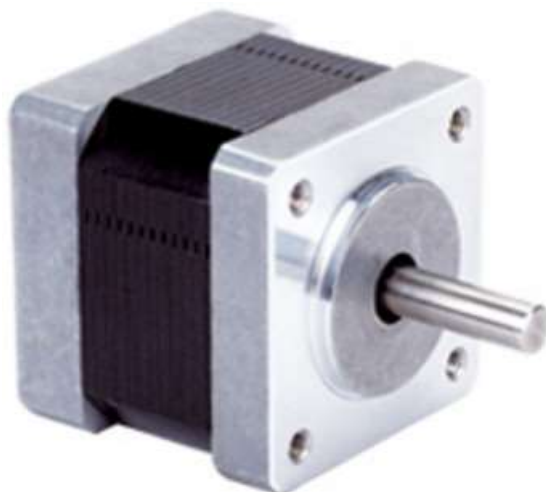


Рисунок 3.12 – Кроковий двигун NEMA 14.

#### **Параметри ротора:**

Шаг: 1,8 град.

Номінальний струм: 0,4 А

Опір фази: 30 Ом ( $\pm 10\%$ )

Момент утримання: мін. 12 Н·см

Інерція ротора: 13 г·см<sup>2</sup>

#### **Загальні характеристики:**

Точність кроку:  $\pm 5\%$

Температурний режим: макс 80°C

Робоча температура довкілля: -20°C... +50°C

Опір ізоляції: 100 МΩ. 500V DC

Радіальний люфт: макс. 0.02mm (450 г.)

Аксiальний люфт: макс. 0.08mm (450 г.)

Макс. радіальне навантаження: 28 Н

Макс. аксiальне навантаження: 10 Н

Вага: 180 г.

Кроковий двигун NEMA 23 (рис. 3.13).



Рисунок 3.13 – Кроковий двигун NEMA 23.

Основні характеристики:

Модель: 57BYG250B-8

Типорозмір фланця: NEMA 23 (56 мм)

Довжина двигуна: 54.5 мм

Діаметр валу: 8 мм (скос 7.5мм)

Кількість фаз: 2

Струм обмоток(макс.): 3.0 А

Момент, що крутить: 1.2 Нм

Максимальний статичний момент, що крутить: 12.6 кгс/см<sup>2</sup>

Крок: 1.8° (200 крок/об)

Точність кроку: 5%

Опір обмоток: 0.9 Ом

Індуктивність: 2.5 мГн

Кількість висновків: 4

Клас ізоляції:

Робоча температура двигуна: 0...80 °С

Температура довкілля: -10...+40 °С

Вага: 0.71 кг

Драйвер ТВ6600 (рис. 3.14).

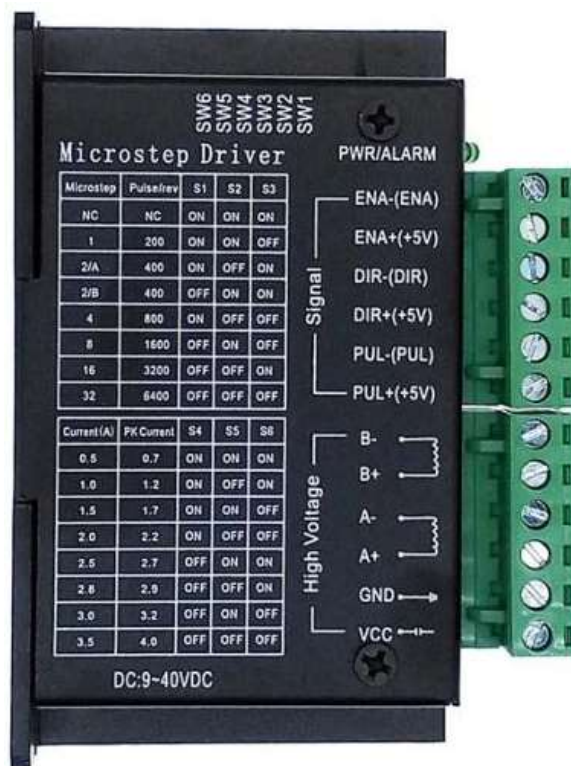


Рисунок 3.14 – Драйвер ТВ6600.

Характеристики:

Напруга живлення схеми керування: 4.75-5.25 В.

Напруга живлення силової частини: 9-42 ст.

Максимальний струм обмоток двигуна: 4 А (протягом 100 мс)

Номінальний тривалий струм: 2 А (рекомендовано примусове обдування радіатора)

Максимальна частота ШІМ: 100кГц.

Дільники мікрокроку: 1, 1/2, 1/4, 1/8, 1/16, 1/32.

Розміри: 96мм x 80мм x 28мм

Драйвер DM556 (рис. 3.15).

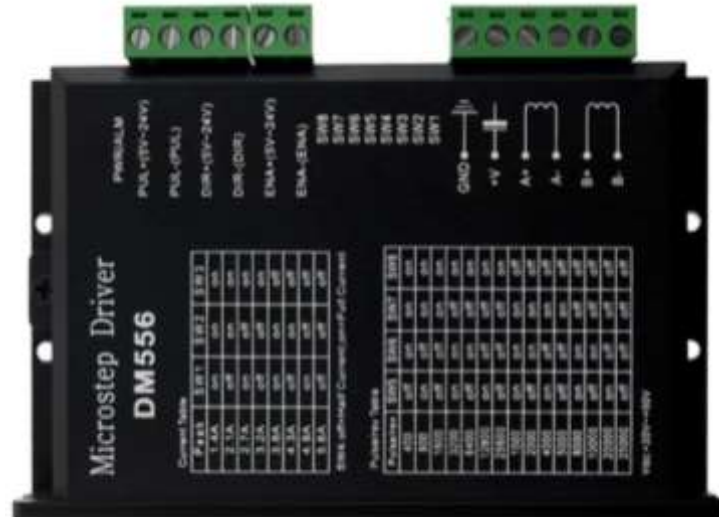


Рисунок 3.15 – Драйвер DM556.

Робочий струм фази – 4А

Максимальний струм фази – 5,6 А

Мінімальний струм фази – 1 А

Напруга живлення 24 – 50в

Підходить для двигунів Nema17-Nema34 до 5А

Плата Arduino uno (рис. 3.16).



Рисунок 3.16 – Плата Arduino uno.

Мікроконтролер ATmega328

Робоча напруга 5В

Напруга живлення (рекомендована) 7-12В

Напруга живлення (гранична) 6-20В

Цифрові входи/виходи 14 (з них 6 можуть використовуватися як ШИМ-виходи)

Аналогові входи 6

Максимальний струм одного виводу 40 мА

Максимальний вихідний струм виведення 3.3V 50 мА

Flash-пам'ять 32 КБ (ATmega328) з яких 0.5 КБ використовуються завантажувачем

SRAM 2 КБ (ATmega328)

EEPROM 1 КБ (ATmega328)

Тактова частота 16 МГц

Ультразвуковий датчик HC-SR04 (рис. 3.17).



Рисунок 3.17 – Ультразвуковий датчик HC-SR04.

- Робоча напруга: 3.8 - 5.5В
- Тип: HC-SR04
- Струм: 8 мА
- Частота: 40 кГц

- Максимальна дистанція: 1500 мм
- Мінімальна дистанція: 0 см
- Роздільна здатність: 3 мм
- Ширина імпульсів: 10 мкс
- Кут: 15 градусів
- Зовнішні габарити: 37x20x15 мм

Тензодатчик з платою керування НХ711 (рис. 3.18).

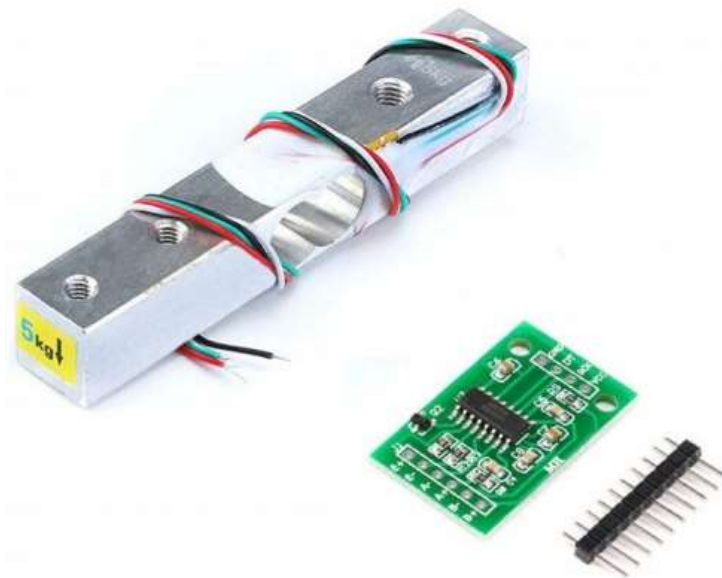


Рисунок 3.18 – Тензодатчик з платою керування НХ711.

розміри: 80x12.7x12.7мм

отвори кріплення: 2\*М4, 2\*М5 (відстань 15 мм)

матеріал: алюміній

точність: висока

комплексна помилка (повторюваність): 0.03% FS

нелінійність: 0.03% FS

Імпеданс:  $1000 \pm 50\Omega$

рекомендована напруга: 5-10В

максимальна вага: до 2 кг

Підключення до НХ711:

червоний: E+

чорний: E-

білий: A-

зелений: A+

Також серед електронних компонентів є два світлодіоди блакитного та жовтого кольорів які працюють від напруги 5В.

### **3.4 Висновки щодо розділу**

В даному розділі описано проведення налагодження та виготовлення деяких деталей, а саме:

- деталь типу лоток (виготовляється на фрезерному верстаті з ЧПУ);
- деталь типу насадка (виготовляється на 3D принтері);
- деталь типу висувна гайка (виготовляється на універсальних верстатах класичним способом).

Також здійснено підбір усіх необхідних електронних компонентів з яких буде складатися роботизований комплекс з зворотнім зв'язком.

## **4 ДОСЛІДНИЦЬКИЙ РОЗДІЛ**

### **4.1. Алгоритм роботи на випробування роботизованої системи з зворотнім зв'язком**

Сутність функціонування роботизованої системи з зворотнім зв'язком полягає в наступному:

- на систему подається живлення;
- висувна гайка з тензодатчиком подається вперед з метою захоплення деталі причому коли не відбулося захоплення і давач маси не отримав необхідний мінімум по масі, то процес захоплення буде відбуватися і надалі;
- коли захоплення з масою більше за вказаний мінімум відбулося, то рухома каретка системи переміщається в зону (згідно маси припустимо 100 гр, 200 гр, 300 грн різні зони припустимо 100мм 200мм 300мм відносно початкової точки). Прибувши в зону деталь відпускається і «падає» припустимо в бункер;
- за падінням деталі слідкує ультразвуковий датчик і в залежності від зони припустимо 100 мм спрацьовує світлодіод певного кольору припустимо блакитного, а коли буде зона 2 200 мм то спрацює датчик і увімкне світлодіод жовтого кольору. Таким чином відбувається завершеність чи то наявність донесення деталі в необхідну зону. Замість світлодіода можна застосувати лічильник і там самим контролювати кількість деталей які попадають в різні зони.
- Після того як світлодіод погасне цикл розпочинається знову тобто рухома каретка їде в зону захоплення нової деталі.

### **4.2 Розробка електричної схеми роботи роботизованого стенду**

На рисунку 4.1 представлена електрична схема згідно якої працює роботизована система з зворотнім зв'язком.

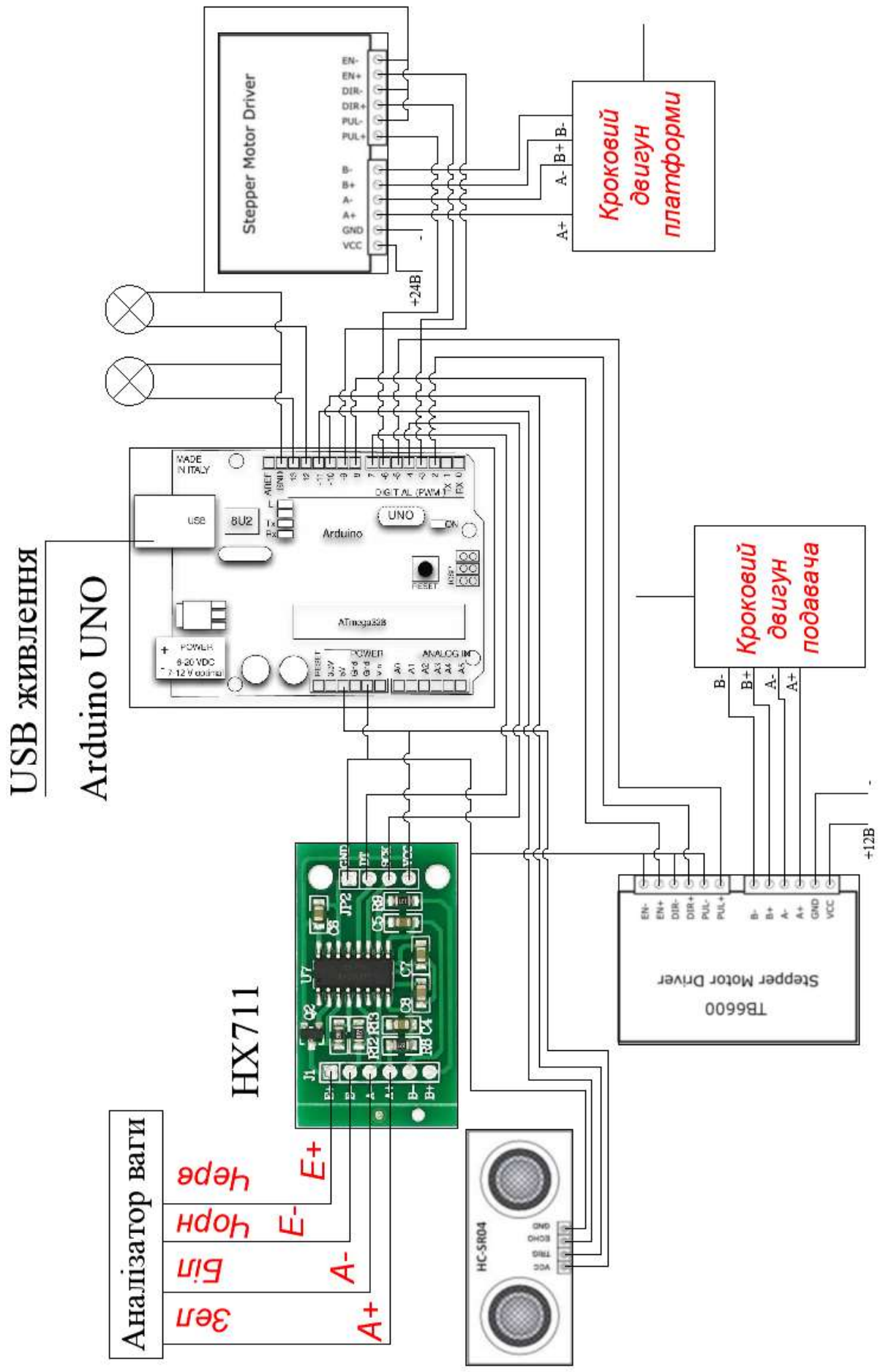


Рис. 4.1 – Електрична схема роботи системи

Сутність схеми полягає в тому що все зав'язано на платі Arduino Uno в якій зашита прошивка яка і керує електричними компонентами системи. Зворотній зв'язок забезпечується тензодатчиком та ультразвуковим датчиком. Керування кроковими двигунами здійснюється за рахунок управління драйверами які в свою чергу і управляють двигунами. Варто відмітити що плата Arduino Uno працює від живлення 5В, кроковий двигун подавача від живлення 12В, кроковий двигун переміщення рухомої каретки від живлення 24В.

### **3.3 Розробка прошивки контролера який керує роботою роботизованої системи**

Програмування плати Arduino UNO на виконання необхідного алгоритму здійснювалося в середовища Arduino IDE. Результат програмування наведено нижче.

```
const int stepPin1 = 5;
const int dirPin1 = 2;
const int enPin1 = 8;

const int stepPin2 = 6;
const int dirPin2 = 3;
const int enPin2 = 9;

#include "HX711.h"
#define calibration_factor -44100 //This value is
obtained using the SparkFun_HX711_Calibration sketch
#define LOADCELL_DOUT_PIN 7
#define LOADCELL_SCK_PIN 4
HX711 scale;

const int trigPin = 10;// trig pin of the module
const int echoPin = 11;// echo pin of the module
long t;// time
```

```

int dist; // distance

void setup() {
pinMode(stepPin1,OUTPUT);
pinMode(dirPin1,OUTPUT);
pinMode(enPin1,OUTPUT);
digitalWrite(enPin1,HIGH);
digitalWrite(dirPin1,HIGH);
pinMode(stepPin2,OUTPUT);
pinMode(dirPin2,OUTPUT);
pinMode(enPin2,OUTPUT);
digitalWrite(enPin2,HIGH);
pinMode(trigPin, OUTPUT);// Set trig pin as OUTPUT
pinMode(echoPin, INPUT);// Set echo pin as INPUT
pinMode (12, OUTPUT);
digitalWrite(12, LOW);
pinMode (13, OUTPUT);
digitalWrite(13, LOW);
delay(2000);
digitalWrite(enPin1,LOW);
digitalWrite(enPin2,LOW);
Serial.begin(9600);
scale.begin(LOADCELL_DOUT_PIN, LOADCELL_SCK_PIN);
scale.set_scale(calibration_factor); //This value is
obtained by using the SparkFun_HX711_Calibration sketch
scale.tare();
}

void loop() {
Serial.println("Put product!");
Serial.print("Reading: ");
Serial.print(scale.get_units()*0.1, 3);
//scale.get_units() returns a float
Serial.print(" kg"); //You can change this to kg but
you'll need to refactor the calibration_factor

```

```

delay (100);
if (scale.get_units()*0.1 < 0.03){
    digitalWrite(dirPin1,HIGH);
    for(int x = 0; x < 10500; x++) {           // 1050 це 1 мм
        digitalWrite(stepPin1,HIGH);
        delayMicroseconds(500);
        digitalWrite(stepPin1,LOW);
        delayMicroseconds(500);
    }
    delay(1000);
    digitalWrite(dirPin1,LOW);
    for(int x = 0; x < 10500; x++) {
        digitalWrite(stepPin1,HIGH);
        delayMicroseconds(500);
        digitalWrite(stepPin1,LOW);
        delayMicroseconds(500);
    }
}

//if (scale.get_units() * 0.1 >= 0.03 &&
scale.get_units() * 0.1 <= 0.035){ // if
(scale.get_units()*0.1 > 0.03){
float value = scale.get_units() * 0.1; // Обчислимо
значення один раз
if (value > 0.02 && value < 0.06) {
    digitalWrite(dirPin2,LOW);
    for(int x = 0; x < 6400; x++) {           // 80 це 1 мм
        digitalWrite(stepPin2,HIGH);
        delayMicroseconds(1000);
        digitalWrite(stepPin2,LOW);
        delayMicroseconds(1000);
    }
}

delay(1000);

```

```

    digitalWrite(dirPin1,HIGH);
for(int x = 0; x < 10500; x++) {           // 1050 цe 1 MM
    digitalWrite(stepPin1,HIGH);
    delayMicroseconds(500);
    digitalWrite(stepPin1,LOW);
    delayMicroseconds(500);
}
delay(1000);
    digitalWrite(dirPin1,LOW);
    for(int x = 0; x < 10500; x++) {
        digitalWrite(stepPin1,HIGH);
        delayMicroseconds(400);
        digitalWrite(stepPin1,LOW);
        delayMicroseconds(400);
    }
    delay(3000);
digitalWrite(trigPin, LOW);// Set trig pin as LOW to
clear the pin
delayMicroseconds(2);// Give a delay of 2ms
digitalWrite(trigPin, HIGH);// Set trig pin as HIGH
delayMicroseconds(10); // Give a delay of 10ms
digitalWrite(trigPin, LOW);// Set trig pin as LOW
t= pulseIn(echoPin, HIGH);// Get the value of time from
the module using pulseIn() function
dist=t*0.034/2;// Calculate distance using the formula
Speed=Distance/Time
Serial.print("Distance of object: ");
Serial.println(dist);// Print the distance on the serial
monitor
    if (dist > 5 && dist < 10) { // Check if dist is
between 5 and 9
        digitalWrite(12, HIGH);    // Turn pin 12 ON
        delay(3000);
        digitalWrite(12, LOW);
    } else {

```

```

    digitalWrite(12, LOW);    // Turn pin 12 OFF
    (optional: ensure it's LOW if not in range)
}

    if (dist > 12 && dist < 20) { // Check if dist is
between 12 and 20
        digitalWrite(13, HIGH);    // Turn pin 13 ON
    } else {
        digitalWrite(13, LOW);    // Turn pin 13 OFF
    (optional)
    }
delay (5000);
    digitalWrite(dirPin2,HIGH);
    for(int x = 0; x < 6400; x++) {
        digitalWrite(stepPin2,HIGH);
        delayMicroseconds(1000);
        digitalWrite(stepPin2,LOW);
        delayMicroseconds(1000);
    }
delay(1000);
}

```

```

// сценарій 2
float value1 = scale.get_units() * 0.1; // Обчислимо
значення один раз
if (value1 > 0.072 && value1 < 0.250) {
    digitalWrite(dirPin2,LOW);
    for(int x = 0; x < 10400; x++) { // 80 це 1 мм
        digitalWrite(stepPin2,HIGH);
        delayMicroseconds(1000);
        digitalWrite(stepPin2,LOW);
        delayMicroseconds(1000);
    }
}

```

```

delay(1000);
    digitalWrite(dirPin1,HIGH);
for(int x = 0; x < 10500; x++) {           // 1050 цe 1 MM
    digitalWrite(stepPin1,HIGH);
    delayMicroseconds(500);
    digitalWrite(stepPin1,LOW);
    delayMicroseconds(500);
}

```

```

delay(1000);
    digitalWrite(dirPin1,LOW);
for(int x = 0; x < 10500; x++) {
    digitalWrite(stepPin1,HIGH);
    delayMicroseconds(400);
    digitalWrite(stepPin1,LOW);
    delayMicroseconds(400);
}

```

```

    delay(3000);
digitalWrite(trigPin, LOW);// Set trig pin as LOW to
clear the pin
delayMicroseconds(2);// Give a delay of 2ms
digitalWrite(trigPin, HIGH);// Set trig pin as HIGH
delayMicroseconds(10); // Give a delay of 10ms
digitalWrite(trigPin, LOW);// Set trig pin as LOW
t= pulseIn(echoPin, HIGH);// Get the value of time from
the module using pulseIn() function
dist=t*0.034/2;// Calculate distance using the formula
Speed=Distance/Time
Serial.print("Distance of object: ");
Serial.println(dist);// Print the distance on the serial
monitor
    if (dist > 5 && dist < 10) { // Check if dist is
between 5 and 9

```

```

    digitalWrite(12, HIGH);    // Turn pin 12 ON
    delay(3000);
    digitalWrite(12, LOW);
} else {
    digitalWrite(12, LOW);    // Turn pin 12 OFF
(optional: ensure it's LOW if not in range)
}
    if (dist > 12 && dist < 20) { // Check if dist is
between 12 and 20
        digitalWrite(13, HIGH);    // Turn pin 13 ON
        delay(3000);
        digitalWrite(13, LOW);
    } else {
        digitalWrite(13, LOW);    // Turn pin 13 OFF
(optional)
    }

delay (5000);
    digitalWrite(dirPin2,HIGH);
    for(int x = 0; x < 10400; x++) {
        digitalWrite(stepPin2,HIGH);
        delayMicroseconds(1000);
        digitalWrite(stepPin2,LOW);
        delayMicroseconds(1000);
    }

delay(1000);
}

}

```

Дана прошивка через USB кабель записується в мікроконтролер плати Arduino UNO

#### 4.4 Складання усіх елементів роботизованого станду

Виготовивши усі необхідні деталі та отримавши усі електронні компоненти – можна приступати до збирання роботизованої системи. Результати виконаних робіт представлено на рисунку 4.2.

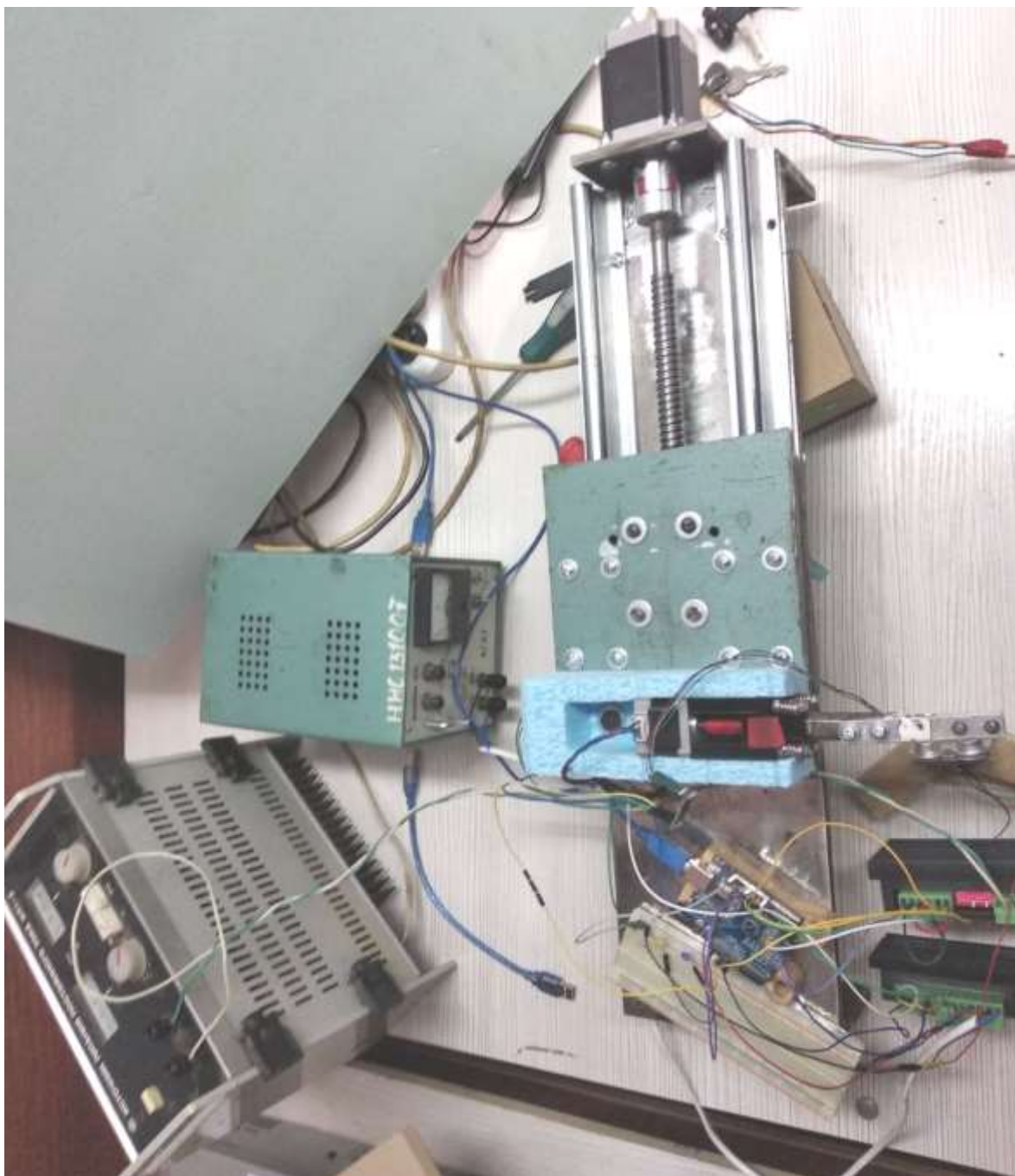


Рис. 4.2 – Складений роботизований стенд з зворотнім зв'язком

Після запуску даного станда усі проектовані та запрограмовані рухи були успішно виконані.

#### **4.5 Висновки щодо розділу**

В даному розділі описаний алгоритм роботи роботизованого стенду, розроблено електричну схему згідно якої підключаються усі попередньо обрані електричні елементи.

Після виготовлення усіх необхідних деталей та після отримання необхідних електричних елементів було зібрано на успішно випробувано роботизовану систему з реалізацією зворотного зв'язку.

## ВИСНОВКИ

В першому розділі проведено огляд принципу роботи та теорія автоматизації. Також проаналізовано матеріал щодо типів зворотного зв'язку роботизованих систем. Опанувавши дану інформацію виникла ідея розробити роботизовану систему в якій реалізовано зворотній зв'язок, тобто виконання операцій системи буде залежати від стану виконання кожного етапу функціонування.

В другому розділі здійснено моделювання усіх деталей роботизованого комплексу та виконано їх збірку в один функціональний механізм. Переконавшись в тому, що усі елементи добре розташовуються один відносно одного і не перетинаються – то можна переходити до етапу в якому можна проектувати технологію їх виготовлення в реальному житті.

В третьому розділі описано проведення налагодження та виготовлення деяких деталей, а саме:

- деталь типу лоток (виготовляється на фрезерному верстаті з ЧПУ);
- деталь типу насадка (виготовляється на 3D принтері);
- деталь типу висувна гайка (виготовляється на універсальних верстатах класичним способом).

Також здійснено підбір усіх необхідних електронних компонентів з яких буде складатися роботизований комплекс з зворотнім зв'язком.

В даному розділі описаний алгоритм роботи роботизованого стенду, розроблено електричну схему згідно якої підключаються усі попередньо обрані електричні елементи.

Після виготовлення усіх необхідних деталей та після отримання необхідних електричних елементів було зібрано на успішно випробувано роботизовану систему з реалізацією зворотного зв'язку.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Матеріали за посиланням <https://www.britannica.com/technology/automation/Modern-developments#ref24841>
2. Матеріали за посиланням <https://www.linkedin.com/advice/3/what-different-types-feedback-control-systems-robotics-rlqif>.
3. Довідкові матеріали програмного забезпечення FeatureCAM.
4. Довідкові матеріали програмного забезпечення SolidWorks.
5. Довідкові матеріали програмного забезпечення Creality Slicer.
6. Довідкові матеріали програмного забезпечення Arduino IDE.
7. Datasheet Arduino UNO
8. Datasheet драйвера ТВ6600
9. Datasheet драйвера DM556
10. Datasheet тензодатчика з платою HX711
11. Datasheet ультразвукового датчика HC-SR04