



**YILDIZ TECHNICAL UNIVERSITY  
FACULTY OF MECHANICAL ENGINEERING**

**ROBOT ARM CONTROL  
WITH HAPTIC FEEDBACK**

2006A601 Kadir Kadiroğlu

1806A038 Oğuzhan Can

1906A602 Özkan Yaşar

**PREPARED IN THE FIELD OF MECHATRONICAL ENGINEERING DEPARTMENT  
MECHATRONICAL SYSTEM DESIGN REPORT**

**Project Advisor:** Assoc. Prof. Hüseyin Ayhan Yavaşoğlu

ISTANBUL, 2024

**TABLE OF CONTENTS**

	Page
REVISION HISTORY .....	v
SYMBOL LIST .....	1
ABBREVIATION LIST.....	2
FIGURE LIST .....	3
LIST OF TABLES .....	5
1. INTRODUCTION .....	7
1.1 Purpose .....	7
1.2 Motivation.....	8
1.3 Scope.....	9
1.4 Literature Review .....	11
1.5 Assumptions .....	13
2. REQUIREMENT SPECIFICATIONS.....	15
2.1 Market Requirements.....	15
2.2 Technical Requirements .....	17
2.3 Design Specifications .....	19
3. THEORETICAL BACKGROUND INFORMATION.....	20
3.1 Lagrange Dynamic Equations & Calculations .....	20
3.2 Bilateral Teleoperation Calculations .....	31
3.3 Power Calculations and Battery Selection for Electronics .....	31
4. DESIGN.....	33
4.1 Mechanical Design .....	33
4.1.1 Customization .....	34
4.1.2 Stress Analysis.....	35
Stress Analysis for the Normal Robot Arm (Master Arm).....	35
Constraints of the Master Arm .....	36
Loads of the Master Arm.....	37
Contacts of the Master Arm.....	38
Mesh of the Master Arm.....	39
Results of the Master Arm.....	41
Stress Analysis for the Generative Robot Arm (Slave Arm).....	46

Constraints of the Slave Arm.....	47
Loads of the Slave Arm.....	48
Contacts of the Slave Arm.....	49
Mesh of the Slave Arm.....	50
Results of the Slave Arm .....	52
Stress Analysis Comparison of the Normal & Generative Designs .....	56
4.1.3     Material Selection .....	57
4.1.4     Mechanical Simulation .....	58
4.1.5     Conceptual Design.....	60
4.1.6     Mechanical Design Realizations.....	66
4.2       Electronical Design.....	69
4.2.1    Component Selection.....	69
4.2.2    Serial Communication (SPI, I2C, WIRELESS) Map of the Robot Arm Circuit ..	69
4.2.3    Simulation/Electronical Design .....	70
Master Arm Electronic Circuit .....	70
Slave Arm Electronic Circuit .....	71
4.2.4    Electronical Realization.....	72
4.3       Software Algorithm .....	75
4.3.1   Algorithm Flow Chart .....	75
4.3.2   Codes .....	77
Master Arm Code .....	77
Master Arm I2C Slave Code .....	85
Slave Arm Code .....	87
Slave Arm I2C Slave Code.....	93
4.4       Control .....	95
4.4.1   Modelling.....	95
4.4.2   Control Method & Optimization .....	97
4.4.3   Control Method & Optimization Realization .....	98
4.4.4   Simulink Simulation Output Graphs .....	99
4.4.5   Control Output Graphs in The Real Prototype .....	101
4.5       Overall System View .....	103
4.5.1   Model of the Overall System.....	103
4.5.2   Overall System View in Real .....	104
5.       WORK SCHEDULE .....	105
6.       BUDGET .....	107
6.1       Material List.....	107
6.2       Service Procurements .....	107

6.3	External Support Application .....	107
CONCLUSION .....	108	
APPENDIX .....	110	
ÖZGEÇMİŞ .....	118	

## REVISION HISTORY

Date	Rev. No	Definition	Author(s)
05/01/2024	1.0	Introduction Document	Kadir Kadiroğlu Öguzhan Can Özkan Yaşar
07/01/2024	1.1	Small Improvements	Kadir Kadiroğlu Öguzhan Can Özkan Yaşar
24/05/2024	2.0	Updates for Realization	Kadir Kadiroğlu Öguzhan Can Özkan Yaşar

## SYMBOL LIST

$\tau$	Torque Matrix [ $N \cdot m$ ]
$\omega$	Angular Velocity [ $\frac{rad}{sec}$ ]
$\sigma$	Sub-Equations
$C$	Torque Centrifugal and Coriolis Matrix [ $N \cdot m$ ]
$D$	Inertial Matrix [ $kg \cdot m^2$ ]
$G$	Gravitation Matrix [ $\frac{m}{s^2}$ ]
$I$	Inertia [ $kg \cdot m^2$ ]
$J$	Jacobian
$m$	Mass [ $kg$ ]
$O_c$	Center of Links [ $m$ ]
$J^T$	Transpose of Jacobian
$\partial$	Partial Derivative
$l$	Length [ $m$ ]
$g$	Gravity [ $\frac{m}{s^2}$ ]
$\theta$	Joint Angle [ $rad$ ]
$\dot{\theta}$	Joint Angular Velocity [ $\frac{rad}{s}$ ]
$\ddot{\theta}$	Joint Angular Acceleration [ $\frac{rad}{s^2}$ ]
$q$	General Joint Angle Parameter [ $rad$ ]
$\dot{q}$	General Joint Angular Velocity [ $\frac{rad}{s}$ ]
$\ddot{q}$	General Joint Angular Acceleration [ $\frac{rad}{s^2}$ ]

### **ABBREVIATION LIST**

RF	Radio-Frequency
PID	Proportional Integral Derivative
DC	Direct Current
LED	Light Emitting Diode
W.r.t.	With Respect To

## FIGURE LIST

Figure 1.1 Bilateral Teleoperation Example.....	7
Figure 1.2 Illustration of Our Robot Arms .....	8
Figure 1.3 Teleoperation System Illustration .....	12
Figure 2.1 Global Market Representation .....	15
Figure 3.1 - Joint Angles & Link Centers Illustration .....	20
Figure 3.2 Length & Distance Parameters in Robot Arm .....	29
Figure 3.3 Compact Design of the Battery According to our Calculations.....	32
Figure 4.1 Mechanical Design of Master & Slave Arm in Fusion 360 .....	33
Figure 4.2 Lengths & Distances of the Robot Arm.....	34
Figure 4.3 Exploded View of the Robot Arm Assembly .....	34
Figure 4.4 Technical Drawing of the Robot Arm.....	35
Figure 4.5 Constraints of the Master Arm.....	36
Figure 4.6 Gravity Force for the Master Arm .....	37
Figure 4.7 Load for the Master Arm.....	37
Figure 4.8 Contacts of the Master Arm .....	38
Figure 4.9 Offset Bonded .....	38
Figure 4.10 Mesh of the Master Arm .....	39
Figure 4.11 Local Mesh Control .....	39
Figure 4.12 Local Mesh Control .....	40
Figure 4.13 Local Mesh Control .....	40
Figure 4.14 Mesh & Loads .....	40
Figure 4.15 Materials of the Master Arm .....	41
Figure 4.16 Point Mass for Servo Motor.....	42
Figure 4.17 Point Mass for Servo Motor.....	42
Figure 4.18 Result Summary .....	43
Figure 4.19 Safety Factor .....	44
Figure 4.20 Stress (von Mises) .....	44
Figure 4.21 1 <sup>st</sup> Principal .....	45
Figure 4.22 3 <sup>rd</sup> Principal .....	45
Figure 4.23 Displacement.....	46
Figure 4.24 Constraints for the Slave Arm.....	47
Figure 4.25 Gravity Force for the Slave Arm.....	48
Figure 4.26 Load for the Slave Arm.....	48
Figure 4.27 Contacts of the Slave Arm .....	49
Figure 4.28 Contacts for the Slave Arm .....	50
Figure 4.29 Mesh of the Slave Arm .....	50
Figure 4.30 Local Mesh Control .....	51
Figure 4.31 Result Summary .....	52
Figure 4.32 Safety Factor .....	53
Figure 4.33 Stress (von Mises) .....	53
Figure 4.34 1 <sup>st</sup> Principal .....	54
Figure 4.35 3 <sup>rd</sup> Principal .....	54
Figure 4.36 Displacement.....	55
Figure 4.37 Normal & Generative Design Comparison .....	56

Figure 4.38 PLA Material Properties Shown in Fusion 360 Material Properties Page.....	57	
Figure 4.39 Nylon 12 Properties .....	58	
Figure 4.40 Master Robot Arm Multibody Diagram in Simulink.....	59	
Figure 4.41 Slave Robot Arm Multibody Diagram in Simulink .....	59	
Figure 4.42 Multibody Robot Arms Animation Output in Simulink .....	60	
Figure 4.43 Second Link Generative Design Process .....	60	
Figure 4.44 Objectives & Limits .....	61	
Figure 4.45 Structural Constraints.....	61	
Figure 4.46 Conceptual Designs.....	62	
Figure 4.47 Second Link Generative Design Output .....	62	
Figure 4.48 Third Link Generative Design Process .....	63	
Figure 4.49 Safety Factor .....	63	
Figure 4.50 Obstacle Geometries for Third Link .....	64	
Figure 4.51 Generative Design Output for Third Link & Stress Distribution.....	64	
Figure 4.52 The Assembly of Generative Design Parts & Its Unique Design .....	65	
Figure 4.53 Master Arm & its Circuit	Figure 4.54 Master Arm & its Circuit .....	66
Figure 4.55 The Arms & Wiring Process	Figure 4.56 Soldering Process.....	66
Figure 4.57 Mounting on Wooden Board	Figure 4.58 Arrangement of the Arms .....	67
Figure 4.59 Final Position of the Arms	Figure 4.60 Example of Lifting an Object.....	68
Figure 4.61 Example of Lifting an Object.....	68	
Figure 4.62 Serial Communication Map .....	69	
Figure 4.63 Controller Arm (Master Arm) Circuit Design on Proteus .....	70	
Figure 4.64 Controlled Arm (Slave Arm) Circuit Design on Proteus .....	71	
Figure 4.65 The Layout of The Circuit	Figure 4.66 The Layout of The Circuit .....	72
Figure 4.67 The Layout of The Wires	Figure 4.68 The Layout of The Wires .....	72
Figure 4.69 The Layout of The Wires	Figure 4.70 Soldering from Underside .....	73
Figure 4.71 Embedding Encoder in Servo	Figure 4.72 Embedding Encoder in Servo .....	73
Figure 4.73 Embedded-Encoder Servo	Figure 4.74 Wiring Process of The Arms .....	74
Figure 4.75 Master & Slave Arms	Figure 4.76 Soldering Process.....	74
Figure 4.77 Master Arm & its Circuit .....	75	
Figure 4.78 Flow Chart Algorithm.....	76	
Figure 4.79 Robot Arms Haptic Feedback Simscape Simulation .....	95	
Figure 4.80 Master Arm Multibody Diagram .....	96	
Figure 4.81 Slave Arm Multibody Diagram.....	96	
Figure 4.82 Joint Actuation & Sensing Types in Simscape .....	97	
Figure 4.83 PID Control Block Scheme.....	98	
Figure 4.84 Loop Tuning Iteration Process .....	98	
Figure 4.85 PID Coefficients Found by Optimization .....	98	
Figure 4.86 Underdamped System Response .....	99	
Figure 4.87 Delay Between the Arm Motions Output.....	99	
Figure 4.88 Master First Joint Angle – Slave First Joint Angle Comparison Output .....	100	
Figure 4.89 Master Second Joint Angle – Slave Second Joint Angle Comparison Output ...	100	
Figure 4.90 Master Third Joint Angle – Slave Third Joint Angle Comparison Output .....	101	
Below graph illustrates the output as the Slave Arm attempts to move instantaneously from its current position at 90 degrees to the 0 position of the Master Arm. It can be likened to providing a step response to the Slave Arm joint.4.91 .....	101	
Figure 5.1 Road Map .....	106	

**LIST OF TABLES**

Table 2-1 Requirements Specification Table .....	19
Table 4-1 Component List.....	69
Table 5-1 - Gantt Chart of The Project.....	105

## SUMMARY

This project introduces a haptic feedback-controlled robotic arm designed to optimize laboratory processes. Focused on reducing human intervention, improving efficiency, and ensuring adaptability, the system incorporates force feedback integration, precise control algorithms, and safety features. The project contributes to laboratory automation with a versatile, safe, and efficient solution, showcasing adaptability for diverse setups.

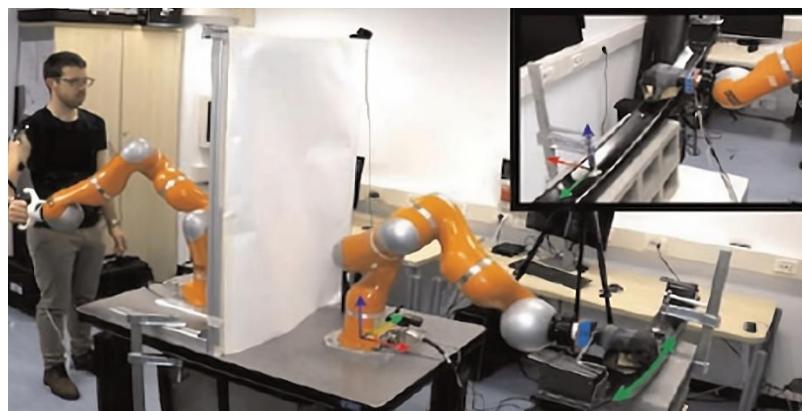
*Keywords: Haptic feedback, Robotic arm, Laboratory automation, Control algorithms, Force feedback integration.*

## 1. INTRODUCTION

This section will cover the purpose of the project, motivation, scope of operation, previous examples in the literature, ways of benefiting from these examples, assumptions and considerations regarding the operation and design.

### 1.1 Purpose

The aim of our project is to revolutionize laboratory processes by developing and implementing an advanced haptic feedback-controlled robotic arm. Through the utilization of cutting-edge haptic technology, our system seeks to minimize direct human-object interaction, enhance automation, and deliver a realistic and intuitive user experience[1]. The primary objectives encompass reducing risks associated with human-object interaction, ensuring human safety, and transferring manual skills to the controlled arm via the controller arm. Furthermore, our project aims to optimize operational efficiency while ensuring seamless integrability and adaptability within laboratory settings[2].



**Figure 1.1 Bilateral Teleoperation Example**

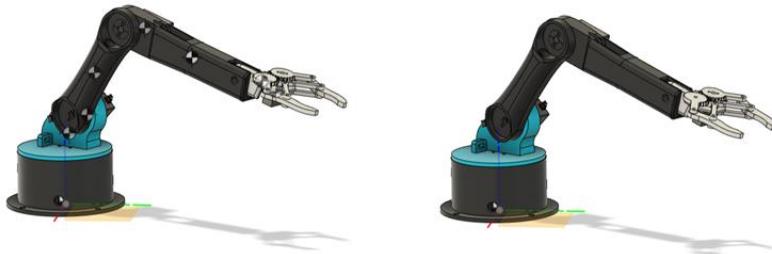
With a focus on achieving minimal latency, maximum precision, and effortless integration, our project strives to create a safer and more efficient laboratory environment.

By addressing challenges inherent in laboratory operations, we aim to elevate the overall quality of research and experimentation[3]. The application of haptic feedback-controlled robotic arms extends beyond laboratories, offering the potential for widespread benefits in various industries. Through improved safety, enhanced efficiency, and increased employee satisfaction, our project seeks to set a new standard for automation solutions, positively impacting both laboratory settings and broader industrial applications.

## 1.2 Motivation

The motivation behind our project stems from a deep-seated commitment to advancing laboratory technologies and automation, coupled with a keen awareness of the challenges inherent in traditional laboratory processes[4]. Recognizing the limitations and risks associated with direct human-object interaction, as well as the potential for human error, we are motivated to create a transformative solution that harnesses the power of haptic feedback-controlled robotic arms.

The desire to minimize human intervention in laboratory tasks, enhance overall operational efficiency, and create a more immersive and realistic user experience propels our team forward. We are motivated by the prospect of significantly reducing the potential for accidents and errors, thereby elevating the safety standards within laboratory environments[5]. Additionally, our project aims to address inefficiencies and streamline workflows, allowing researchers and professionals to focus on the core aspects of their work.



**Figure 1.2 Illustration of Our Robot Arms**

Moreover, the motivation extends beyond the confines of laboratory settings. We are driven by the broader vision of introducing a versatile technology that can find applications across various industries. The potential benefits in terms of improved safety, increased efficiency, and enhanced employee satisfaction serve as catalysts for our endeavor[6].

By developing a haptic feedback-controlled robotic arm that is not only effective but also adaptable to diverse environments, we aspire to make a lasting impact on the landscape of automation and contribute to the evolution of technology in a way that positively influences human-machine interactions[7].

### 1.3 Scope

The scope of our project encompasses the comprehensive development and implementation of a haptic feedback-controlled robotic arm system tailored for laboratory use[8]. This includes a multifaceted approach covering hardware design, software development, and integration strategies to create a seamless and effective solution. The key components of the project scope are outlined below:

#### 1. Hardware Development:

- Design and fabrication of a robotic arm with haptic feedback capabilities.
- Integration of sensors for precise positioning and force feedback.
- Implementation of a control system for real-time interaction and feedback.

#### 2. Software Integration:

- Creation of algorithms for haptic feedback to simulate realistic interactions.
- Integration with existing laboratory systems and protocols.

#### 3. Latency Optimization:

- Minimization of latency to ensure real-time responsiveness of the system.
- Calibration and synchronization of sensors and actuators for maximum precision.

#### 4. Operational Efficiency Enhancement:

- Implementation of automation features to streamline laboratory processes.
- Optimization of the robotic arm's movements for efficient and accurate task execution.

## 5. Safety Features:

- Incorporation of safety protocols to mitigate risks associated with human-machine interactions.
- Emergency stop mechanisms and fail-safe procedures to ensure user and equipment safety.

## 6. Adaptability and Integrability:

- Design of the robotic arm system to be easily adaptable to various laboratory setups.
- Compatibility with different experimental setups and equipment commonly used in laboratories.

## 7. Testing and Validation:

- Rigorous testing of the system under diverse laboratory conditions.
- Validation of haptic feedback effectiveness and precision in simulated and real-world scenarios.

## 8. Documentation and Training:

- Creation of comprehensive documentation for system setup, operation, and troubleshooting.
- Development of training materials for users to maximize the benefits of the haptic feedback-controlled robotic arm.

## 9. Scalability and Future Developments:

- Consideration of scalability for potential expansion or integration with other robotic systems.
- Identification of areas for future enhancements and developments based on user feedback and technological advancements.

By addressing these aspects within the defined scope, our project aims to deliver a state-of-the-art haptic feedback-controlled robotic arm system that not only meets the immediate needs of laboratory automation but also sets the stage for future advancements and applications in diverse industries[9].

#### 1.4 Literature Review

A literature review provides a comprehensive overview of existing research and developments related to the project's focus. Here's a brief literature review for a haptic feedback-controlled robotic arm in laboratory settings:

##### 1. Haptic Feedback in Robotics:

- The incorporation of haptic feedback in robotic systems has been the subject of extensive research. Studies by Burdea and Coiffet (2003) and Colgate et al. (1996) have explored the significance of haptic feedback in improving user interaction and control precision in various robotic applications.

##### 2. Robotic Arm Technologies:

- Recent advancements in robotic arm technologies, as highlighted in works by Siciliano and Khatib (2016) and Craig (2005), showcase the evolution of robotic arms from basic manipulators to sophisticated systems capable of intricate tasks. These studies provide insights into the design considerations and kinematics of robotic arms.

##### 3. Laboratory Automation:

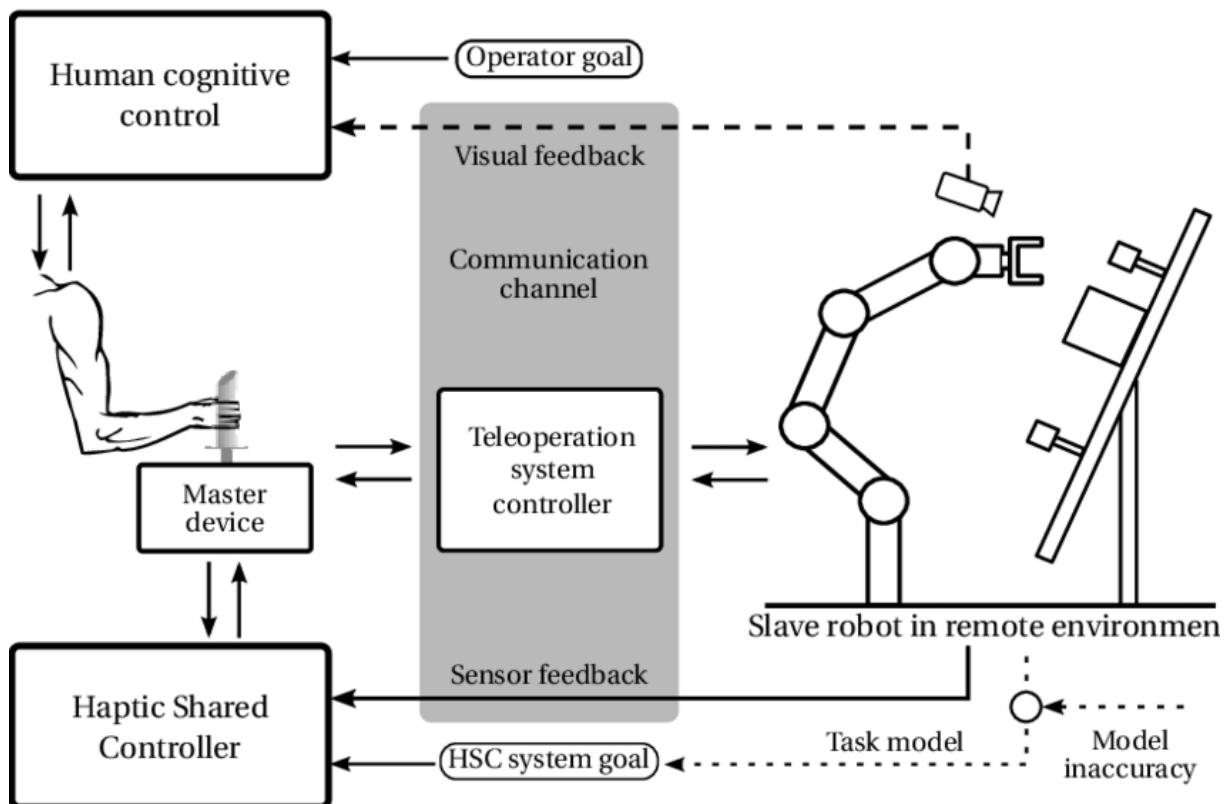
- Research by Lee and Tomizuka (2000) and Choudhury et al. (2017) emphasizes the growing importance of laboratory automation for increased efficiency and reduced human intervention. Integration of robotic systems, particularly in laboratory settings, has shown potential for minimizing errors and enhancing productivity.

##### 4. Human-Robot Interaction and Safety:

- Human-robot interaction and safety have been significant concerns. Studies by Fong et al. (2003) and Mainprice and Berenson (2013) delve into the challenges and solutions related to safe human-robot collaboration, providing insights into the design of safety features and protocols.

## 5. Latency Optimization in Robotic Systems:

- Addressing latency issues in robotic systems is crucial for real-time responsiveness. Research by Okamura et al. (1999) and Colgate and Brown (1994) explore techniques for latency reduction in haptic interfaces, which can be applicable to the development of haptic feedback-controlled robotic arms.



**Figure 1.3 Teleoperation System Illustration**

## 6. Applications of Haptic Feedback in Medicine:

- The medical field has seen successful integration of haptic feedback in robotic surgery systems (Taylor et al., 1999). Lessons learned from these applications can inform the development of haptic feedback systems in laboratory settings.

## 7. Industrial Applications of Robotic Arms:

- Literature on the use of robotic arms in industrial settings, such as the works by Aspragathos and Arabatzis (2003) and Nof (1999), provides valuable insights into the scalability, adaptability, and challenges associated with robotic systems, which can be relevant for our project's goals.

## 8. Integration Challenges and Solutions:

- The integration of robotic systems in laboratory environments presents challenges. Studies by Khusainov et al. (2002) and Ribeiro et al. (2015) discuss strategies for overcoming integration challenges and achieving seamless collaboration between robotic systems and laboratory workflows.

This literature review provides a foundational understanding of key concepts and findings related to haptic feedback, robotic arms, laboratory automation, and human-robot interaction. Drawing on these insights, our project aims to contribute to the ongoing discourse by developing a haptic feedback-controlled robotic arm specifically tailored for laboratory use.

## 1.5 Assumptions

### 1. Technology Compatibility:

The selected haptic feedback technology and sensors are compatible with the robotic arm's design and can be seamlessly integrated without significant technical constraints.

### 2. User Training and Adaptation:

Users will receive sufficient training and guidance to adapt to the haptic feedback-controlled interface, ensuring effective and safe utilization within laboratory settings.

### 3. Laboratory Environment Suitability:

The designed robotic arm system is suitable for a variety of laboratory environments, taking into consideration different setups, equipment, and working conditions.

### 4. Availability of Necessary Resources:

Adequate resources, including funding, materials, and skilled personnel, will be available throughout the project timeline to support the development, testing, and refinement of the haptic feedback-controlled robotic arm.

**5. Compliance with Safety Standards:**

The implemented safety features and protocols meet or exceed relevant industry and laboratory safety standards, ensuring a secure working environment and minimizing potential risks.

**6. Collaborative Research Opportunities:**

The project may open opportunities for collaborative research and feedback from laboratory professionals, allowing for iterative improvements and refinements based on real-world usage.

**7. Technology Scalability:**

The developed haptic feedback-controlled robotic arm system is designed with scalability in mind, allowing for potential future expansions or adaptations to different robotic platforms and applications.

**8. Acceptance of Automation in Laboratory Settings:**

Laboratory professionals are receptive to the integration of automation technologies, including haptic feedback-controlled robotic arms, and recognize the potential benefits in terms of efficiency and safety.

**9. Availability of Skilled Technical Support:**

Skilled technical support, whether internal or external, is available to address any unforeseen technical challenges, glitches, or issues that may arise during the development and deployment phases.

**10. Regulatory Compliance:**

The project adheres to relevant regulatory requirements and ethical guidelines governing the development and implementation of robotic systems, ensuring ethical and responsible use.

## 2. REQUIREMENT SPECIFICATIONS

### 2.1 Market Requirements

#### 1. Performance:

The system must exhibit high precision and responsiveness in executing tasks, ensuring optimal performance in laboratory applications.

#### 2. Functionality:

The robotic arm should be versatile, accommodating various laboratory tasks and procedures, thereby enhancing its overall functionality.

#### 3. Cost:

The cost of the haptic feedback-controlled robotic arm system must be competitive, aligning with market expectations and ensuring affordability for potential users.



**Figure 2.1 Global Market Representation**

**4. Energy Resources and Efficiency:**

The system should be designed with energy-efficient components and mechanisms to minimize power consumption and contribute to sustainable practices[10].

**5. Environmental Impact:**

The manufacturing, operation, and disposal of the robotic arm should adhere to environmentally friendly practices, minimizing the overall environmental impact.

**6. Legal Considerations:**

The system must comply with all relevant legal regulations and standards governing robotic systems, ensuring adherence to ethical and legal frameworks.

**7. Health and Safety:**

Safety features must be prioritized to ensure the well-being of users, minimizing potential risks associated with the operation of the robotic arm in laboratory settings.

**8. Maintenance Requirements:**

The system should have minimal maintenance requirements, reducing downtime and associated costs for end-users.

**9. Reliability and Availability:**

The robotic arm must be highly reliable, with minimal downtime, ensuring its availability for use whenever required in laboratory processes.

**10. Manufacturability:**

Design considerations should facilitate ease of manufacturing, allowing for cost-effective and scalable production of the haptic feedback-controlled robotic arm.

**11. Usability:**

The system interface should be user-friendly, enabling ease of operation for individuals with varying levels of technical expertise in laboratory environments.

**12. Accessibility:**

The robotic arm should be easily accessible and adaptable to different laboratory setups, accommodating diverse research requirements.

### 13. Political Factors:

Consideration of political factors involves compliance with regulations, policies, and geopolitical considerations that may impact the development and deployment of the robotic arm.

### 14. Social and Cultural Implications:

The design and implementation of the system should take into account social and cultural factors, ensuring acceptance and usability in diverse cultural contexts.

These market requirements serve as a foundational framework for guiding the development and deployment of the haptic feedback-controlled robotic arm, aligning the project with the expectations and needs of the target market.

## 2.2 Technical Requirements

Technical Requirements	Description
Force Feedback Integration	<p>Maximum load capacity of 300 g and a maximum distance of 0.45 m are adhered to, thereby not surpassing the operator's natural force application capacity.</p> <p>The ability to adjust force transmission and position transmission ratios through different modes.</p>
Enabling Remote Data Transmission	<p>Fast (2.4 GHz) and reliable data transmission from a distance of 100 meters using RF modules.</p>
Precision and Force Transfer	<p>Achieving <math>\pm 2</math> cm accuracy in linear movements, <math>\pm 1</math> degree angular accuracy, and <math>\pm 0.5</math> Newton force transmission accuracy</p>

Optimization of Latency Duration	Minimizing the delay(approximately 35ms) between the operator's manipulations and the response of the controlled arm.
Security Protocols and Initialization Process	Safely determining the initial positions of the arms.  Creating safety protocols and completing the initialization process within a maximum of 5 seconds.
Development of Control Algorithms	Implementing PID control algorithms to accurately transmit the operator's manipulations to the controller arm for haptic feedback integration.  Applying digital filters to dampen hand tremors.(3Hz cut-off freq. -40dB/decade roll-off rate)
Integration and Compatibility	Facilitating easy integration of the robot arm with different devices  Capability to be integrated into different platforms using standard attachments or adapters

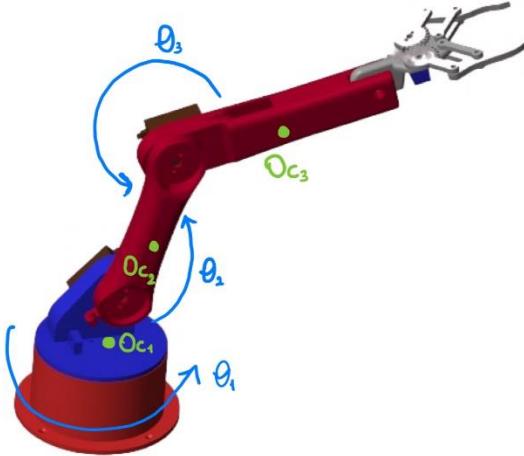
## 2.3 Design Specifications

**Table 2-1 Requirements Specification Table**

Market Requirements	Technical Requirements	Description
2,7,12	1. Force Feedback Integration	The robotic arm should be able to perform various laboratory tasks by adjusting force and position transmission ratios in different modes, enhancing its overall functionality. Safety features are prioritized to protect user well-being, and accessibility ensures adaptability to different laboratory setups.
1	2. Precision and Force Transfer	The system must achieve $\pm 2$ cm accuracy in linear movements, $\pm 1$ degree angular accuracy, and $\pm 0.5$ Newton force transmission accuracy to ensure optimal performance in laboratory applications.
1	3. Optimization of Latency Duration	Minimizing the delay between the operator's manipulations and the response of the controlled arm contributes to optimal performance in laboratory applications.
7	4. Security Protocols and Initialization Process	Safely determining the initial positions of the arms and creating safety protocols within a maximum of 5 seconds contribute to user safety in laboratory settings.
11	5. Development of Control Algorithms	Control algorithms contribute to the user-friendly interface, enabling ease of operation for individuals with varying levels of technical expertise in laboratory environments.
12	6. Integration and Compatibility	Facilitating easy integration of the robot arm with different devices and platforms using standard attachments or adapters ensures adaptability to diverse laboratory setups.

### 3. THEORETICAL BACKGROUND INFORMATION

#### 3.1 Lagrange Dynamic Equations & Calculations



**Figure 3.1 - Joint Angles & Link Centers Illustration**

$$\tau = D(q)\ddot{q} + C(q, \dot{q}) + G(q)$$

$$D(q) = \sum_{i=1}^3 (m_i \cdot J_{v_i}^T \cdot J_{v_i} + J_{\omega_i}^T \cdot I_i \cdot J_{\omega_i})$$

$l_{c1}$  = First link group center,       $l_{c2}$  = Second link group center,

$l_{c3}$  = Third link group center

$$l_{c1} = 0, \quad l_{c2} = l_{c2}, \quad l_{c3} = \sqrt{\frac{m_{l,3} l_{c3}^2 + m_{gs} l_{gs}^2 + m_{load} l_{load}^2 + m_s l s_3^2}{m_{gs} + m_{l,3} + m_{load} + m_s}}$$

$m_{l,1}$  = mass of the first link,       $m_{l,2}$  = mass of the second link

$m_{l,3}$  = mass of the third link,       $m_{gs}$  = mass of the gripper servo

$m_s$  = mass of the servos,       $m_{load}$  = mass of the load

$\theta_1 = \text{First joint angle}, \quad \theta_2 = \text{Second joint angle}, \quad \theta_3 = \text{Third joint angle}$

$$O_{C_1}^0 = \text{center of first link group}, \quad O_{C_1}^0 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$O_{C_2}^0 = \text{center of second link group}, \quad O_{C_2}^0 = \begin{bmatrix} lc_2 \cos(\theta_1) \cos(\theta_2) \\ -lc_2 \cos(\theta_2) \sin(\theta_1) \\ lc_2 \sin(\theta_2) \end{bmatrix}$$

$O_{C_3}^0 = \text{center of third link group},$

$$O_{C_3}^0 = \begin{bmatrix} \cos(\theta_1) (lc_3 \cos(\theta_2 + \theta_3) + lc_2 \cos(\theta_2)) \\ (-\sin(\theta_1) (lc_3 \cos(\theta_2 + \theta_3) + lc_2 \cos(\theta_2))) \\ lc_3 \sin(\theta_2 + \theta_3) + lc_2 \sin(\theta_2) \end{bmatrix}$$

$J_{V_{C_1}} = \text{First link group velocity jacobian},$

$$J_{V_{C_1}} = \begin{bmatrix} \frac{\partial O_{C_1}^0}{\partial \theta_1} & \frac{\partial O_{C_1}^0}{\partial \theta_2} & \frac{\partial O_{C_1}^0}{\partial \theta_3} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$d_1 = m_1 \cdot J_{v_1}^T \cdot J_{v_1}, \quad d_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$J_{V_{C_2}} = \text{Second link group velocity jacobian},$

$$J_{V_{C_2}} = \begin{bmatrix} \frac{\partial O_{C_2}^0}{\partial \theta_1} & \frac{\partial O_{C_2}^0}{\partial \theta_2} & \frac{\partial O_{C_2}^0}{\partial \theta_3} \end{bmatrix}$$

$$J_{V_{C_2}} = \begin{bmatrix} -lc_2 \cos(\theta_2) \sin(\theta_1) & -lc_2 \cos(\theta_1) \sin(\theta_2) & 0 \\ -lc_2 \cos(\theta_1) \cos(\theta_2) & lc_2 \sin(\theta_1) \sin(\theta_2) & 0 \\ 0 & lc_2 \cos(\theta_2) & 0 \end{bmatrix}$$

$$d_2 = m_2 \cdot J_{v_2}^T \cdot J_{v_2}$$

$$d_2 = \begin{bmatrix} m_2 lc_2^2 \cos(\theta_1)^2 \cos(\theta_2)^2 + m_2 lc_2^2 \cos(\theta_2)^2 \sin(\theta_1)^2 & 0 & 0 \\ 0 & m_2 lc_2^2 \cos(\theta_1)^2 \sin(\theta_2)^2 + m_2 lc_2^2 \cos(\theta_2)^2 + m_2 lc_2^2 \sin(\theta_1)^2 \sin(\theta_2)^2 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$J_{V_{C_3}}$  = Third link group velocity jacobian,

$$J_{V_{C_3}} = \begin{bmatrix} \frac{\partial O_{C_3}^0}{\partial \theta_1} & \frac{\partial O_{C_3}^0}{\partial \theta_2} & \frac{\partial O_{C_3}^0}{\partial \theta_3} \end{bmatrix}$$

$$J_{V_{C_3}} = \begin{bmatrix} -\sin(\theta_1) \sigma_1 & -\cos(\theta_1) \sigma_2 & -lc_3 \sin(\theta_2 + \theta_3) \cos(\theta_1) \\ -\cos(\theta_1) \sigma_1 & \sin(\theta_1) \sigma_2 & lc_3 \sin(\theta_2 + \theta_3) \sin(\theta_1) \\ 0 & \sigma_1 & lc_3 \cos(\theta_2 + \theta_3) \end{bmatrix}$$

Where,

$$\sigma_1 = lc_3 \cos(\theta_2 + \theta_3) + lc_2 \cos(\theta_2)$$

$$\sigma_2 = lc_3 \sin(\theta_2 + \theta_3) + lc_2 \sin(\theta_2)$$

$$d_3 = m_3 \cdot J_{v_3}^T \cdot J_{v_3},$$

$$d_3 = \begin{bmatrix} m_3 \cos(\theta_1)^2 \sigma_4^2 + m_3 \sin(\theta_1)^2 \sigma_4^2 & 0 & 0 \\ 0 & m_3 \sigma_4^2 + m_3 \sin(\theta_1)^2 \sigma_3^2 + m_3 \cos(\theta_1)^2 \sigma_3^2 & \sigma_2 \\ 0 & \sigma_2 & m_3 l c_3^2 \cos(\theta_2 + \theta_3)^2 + m_3 l c_3^2 \sigma_1 \cos(\theta_1)^2 + m_3 l c_3^2 \sigma_1 \sin(\theta_1)^2 \end{bmatrix}$$

Where,

$$\sigma_1 = \sin(\theta_2 + \theta_3)^2$$

$$\sigma_2 = l c_3 m_3 \sin(\theta_2 + \theta_3) \sigma_3 \cos(\theta_1)^2 + l c_3 m_3 \sin(\theta_2 + \theta_3) \sigma_3 \sin(\theta_1)^2 + l c_3 m_3 \cos(\theta_2 + \theta_3) \sigma_4$$

$$\sigma_3 = l c_3 \sin(\theta_2 + \theta_3) + l c_2 \sin(\theta_2)$$

$$\sigma_4 = l c_3 \cos(\theta_2 + \theta_3) + l c_2 \cos(\theta_2)$$

$$I_1 = Ic_1 + Ics_1$$

Where,

$$Ic_1 = \text{First link moment of inertia w.r.t its center}$$

$$Ics_1 = \text{First Joint servo moment of inertia w.r.t its center}$$

$$I_2 = Ic_2$$

Where,

$$Ic_2 = \text{Second link moment of inertia w.r.t its center}$$

$$I_3 = I_{c3} + m_{l,3} \left( l_{c3} - \sqrt{\frac{m_{l,3} l_{c3}^2 + m_{gs} l_{gs}^2 + m_{load} l_{load}^2 + m_s l_{s3}^2}{m_{gs} + m_{l,3} + m_{load} + m_s}} \right)^2$$

Where,

$I_{c3}$  = Third link moment of inertia w.r.t its center

$J_{\omega_1}$  = First link group angular velocity jacobian,

$$J_{\omega_1} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

$$d_4 = J_{\omega_1}^T \cdot I_1 \cdot J_{\omega_1},$$

$$d_4 = \begin{bmatrix} I_{c1} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$J_{\omega_2}$  = Second link group angular velocity jacobian,

$$J_{\omega_2} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

$$d_5 = J_{\omega_2}^T \cdot I_2 \cdot J_{\omega_2}$$

$$d_5 = \begin{bmatrix} I_{c2} & 0 & 0 \\ 0 & I_{c2} & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$J_{\omega_3}$  = Third link group angular velocity jacobian,

$$J_{\omega_3} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

$$d_6 = J_{\omega_3}^T \cdot I_3 \cdot J_{\omega_3}$$

$$d_6 = \begin{bmatrix} Ic_3 & 0 & 0 \\ 0 & Ic_3 & Ic_3 \\ 0 & Ic_3 & Ic_3 \end{bmatrix}$$

$D(q)$  = Inertial Matrix,

$$D(q) = \sum_{i=1}^3 (m_i \cdot J_{v_i}^T \cdot J_{v_i} + J_{\omega_i}^T \cdot I_i \cdot J_{\omega_i}) = d_1 + d_2 + d_3 + d_4 + d_5 + d_6$$

$$D(q) = \begin{bmatrix} d_{11} & d_{12} & d_{13} \\ d_{21} & d_{22} & d_{23} \\ d_{31} & d_{32} & d_{33} \end{bmatrix}$$

Where,

$$d_{11} = Ic_1 + Ic_2 + Ic_3 + m_3 \cos(\theta_1)^2 \sigma_4^2 + m_3 \sin(\theta_1)^2 \sigma_4^2 + Ic_2^2 m_2 \cos(\theta_1)^2 \cos(\theta_2)^2 + Ic_2^2 m_2 \cos(\theta_2)^2 \sin(\theta_1)^2$$

$$d_{12} = 0$$

$$d_{13} = 0$$

$$d_{21} = 0$$

$$d_{22}$$

$$= Ic_2 + Ic_3 + m_3 \sigma_4^2 + m_3 \sin(\theta_1)^2 \sigma_3^2 + Ic_2^2 m_2 \cos(\theta_2)^2 + m_3 \cos(\theta_1)^2 \sigma_3^2 + Ic_2^2 m_2 \cos(\theta_1)^2 \sin(\theta_2)^2 + Ic_2^2 m_2 \sin(\theta_1)^2 \sin(\theta_2)^2$$

$$d_{23} = \sigma_2$$

$$d_{31} = 0$$

$$d_{32} = \sigma_2$$

$$d_{33} = m_3 l c_3^2 \cos(\theta_2 + \theta_3)^2 + m_3 l c_3^2 \sigma_1 \cos(\theta_1)^2 + m_3 l c_3^2 \sigma_1 \sin(\theta_1)^2 + I c_3$$

where

$$\sigma_1 = \sin(\theta_2 + \theta_3)^2$$

$$\sigma_2 = l c_3 m_3 \sin(\theta_2 + \theta_3) \sigma_3 \cos(\theta_1)^2 + l c_3 m_3 \sin(\theta_2 + \theta_3) \sigma_3 \sin(\theta_1)^2 + I c_3 + l c_3 m_3 \cos(\theta_2 + \theta_3) \sigma_4$$

$$\sigma_3 = l c_3 \sin(\theta_2 + \theta_3) + l c_2 \sin(\theta_2)$$

$$\sigma_4 = l c_3 \cos(\theta_2 + \theta_3) + l c_2 \cos(\theta_2)$$

Christoffel symbols,

$$c_{kij} = \frac{1}{2} \left( \frac{\partial(d_{kj})}{\partial\theta_i} + \frac{\partial(d_{ki})}{\partial\theta_j} - \frac{\partial(d_{ij})}{\partial\theta_k} \right)$$

Since there are 27 coefficients in the equations, instead of writing each coefficient individually, C matrix will be used.

$$C_1 = c_{111} + c_{112} + c_{113} + c_{121} + c_{122} + c_{123} + c_{131} + c_{132} + c_{133}$$

$$C_2 = c_{211} + c_{212} + c_{213} + c_{221} + c_{222} + c_{223} + c_{231} + c_{232} + c_{233}$$

$$C_3 = c_{311} + c_{312} + c_{313} + c_{321} + c_{322} + c_{323} + c_{331} + c_{332} + c_{333}$$

*C = Centrifugal and Coriolis Matrix*

$$\mathcal{C} = \begin{bmatrix} C_1 \\ C_2 \\ C_3 \end{bmatrix}$$

$$\mathcal{C} =$$

$$\begin{bmatrix} -2\theta_1\theta_3(\sigma_6 + \sigma_5) - 2\theta_1\theta_2(\sigma_8 + \sigma_7 + \sigma_3 + \sigma_2) \\ \theta_1^2(\sigma_8 + \sigma_7 + \sigma_3 + \sigma_2) + \theta_2^2(\sigma_8 - m_3\sigma_9\sigma_{10} + \sigma_7 - lc_2^2m_2\cos(\theta_2)\sin(\theta_2) + \sigma_3 + \sigma_2) + \theta_3^2\sigma_4 + 2\theta_2\theta_3\sigma_4 \\ \theta_3\sigma_1 + \theta_2(\sigma_6 + \sigma_5 - lc_3m_3\cos(\theta_2 + \theta_3)\sigma_{10}) + \theta_1(\sigma_6 + \sigma_5) + 2\theta_2\theta_3\sigma_1 \end{bmatrix}$$

Where,

$$\sigma_1 = m_3\cos(\theta_2 + \theta_3)\sin(\theta_2 + \theta_3)lc_3^2\cos(\theta_1)^2 + m_3\cos(\theta_2 + \theta_3)\sin(\theta_2 + \theta_3)lc_3^2\sin(\theta_1)^2 - m_3\cos(\theta_2 + \theta_3)\sin(\theta_2 + \theta_3)lc_3^2$$

$$\sigma_2 = lc_2^2m_2\cos(\theta_2)\sin(\theta_1)^2\sin(\theta_2)$$

$$\sigma_3 = lc_2^2m_2\cos(\theta_1)^2\cos(\theta_2)\sin(\theta_2)$$

$$\sigma_4 = lc_3m_3\cos(\theta_2 + \theta_3)\sigma_{10}\cos(\theta_1)^2 + lc_3m_3\cos(\theta_2 + \theta_3)\sigma_{10}\sin(\theta_1)^2 - lc_3m_3\sin(\theta_2 + \theta_3)\sigma_9$$

$$\sigma_5 = lc_3m_3\sin(\theta_2 + \theta_3)\sigma_9\sin(\theta_1)^2$$

$$\sigma_6 = lc_3m_3\sin(\theta_2 + \theta_3)\sigma_9\cos(\theta_1)^2$$

$$\sigma_7 = m_3\sin(\theta_1)^2\sigma_9\sigma_{10}$$

$$\sigma_8 = m_3\cos(\theta_1)^2\sigma_9\sigma_{10}$$

$$\sigma_9 = lc_3\cos(\theta_2 + \theta_3) + lc_2\cos(\theta_2)$$

$$\sigma_{10} = lc_3\sin(\theta_2 + \theta_3) + lc_2\sin(\theta_2)$$

Gravitation Matrix,

$$G(q) = -(J_{v_1}^T \cdot m_1 + J_{v_2}^T \cdot m_2 + J_{v_3}^T \cdot m_3) \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix}$$

$$G(q) = \begin{bmatrix} 0 \\ g(m_3(lc_3\cos(\theta_2 + \theta_3) + lc_2\cos(\theta_2)) + lc_2m_2\cos(\theta_2)) \\ glc_3m_3\cos(\theta_2 + \theta_3) \end{bmatrix}$$

$\tau$  = Torque Matrix,

$$\tau = D(q)\ddot{q} + \mathcal{C}(q, \dot{q}) + G(q)$$

$\ddot{q}$  = Angular Acceleration Vector,

$$\ddot{q} = \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix}$$

$$\tau = \begin{bmatrix} d_{11} & d_{12} & d_{13} \\ d_{21} & d_{22} & d_{23} \\ d_{31} & d_{32} & d_{33} \end{bmatrix} \cdot \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix} + \begin{bmatrix} C_1 \\ C_2 \\ C_3 \end{bmatrix} + G = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix}$$

$\tau_1$  = First Joint Torque Equation from the  $\tau$  Matrix

$$\tau_1 =$$

$$\begin{aligned} & \theta_1 (\text{Ic}_1 + \text{Ic}_2 + \text{Ic}_3 + m_3 \cos(\theta_1)^2 \sigma_{11}^2 + m_3 \sin(\theta_1)^2 \sigma_{11}^2 + \text{lc}_2^2 m_2 \cos(\theta_1)^2 \cos(\theta_2)^2 \\ & + \text{lc}_2^2 m_2 \cos(\theta_2)^2 \sin(\theta_1)^2) - 2 \theta_1 \theta_3 (\sigma_8 + \sigma_7) - 2 \theta_1 \theta_2 (\sigma_{10} + \sigma_9 + \sigma_3 + \sigma_2) \end{aligned}$$

$\tau_2$  = Second Joint Torque Equation from the  $\tau$  Matrix

$$\tau_2 =$$

$$\begin{aligned} & \theta_1^2 (\sigma_{10} + \sigma_9 + \sigma_3 + \sigma_2) \\ & + \theta_2^2 (\sigma_{10} - m_3 \sigma_{11} \sigma_{12} + \sigma_9 - \text{lc}_2^2 m_2 \cos(\theta_2) \sin(\theta_2) + \sigma_3 + \sigma_2) + \theta_3 \sigma_5 \\ & + g (m_3 \sigma_{11} + \text{lc}_2 m_2 \cos(\theta_2)) + \theta_3^2 \sigma_6 \\ & + \theta_2 (\text{Ic}_2 + \text{Ic}_3 + m_3 \sigma_{11}^2 + m_3 \sin(\theta_1)^2 \sigma_{12}^2 + \text{lc}_2^2 m_2 \cos(\theta_2)^2 + m_3 \cos(\theta_1)^2 \sigma_{12}^2 \\ & + \text{lc}_2^2 m_2 \cos(\theta_1)^2 \sin(\theta_2)^2 + \text{lc}_2^2 m_2 \sin(\theta_1)^2 \sin(\theta_2)^2) + 2 \theta_2 \theta_3 \sigma_6 \end{aligned}$$

$\tau_3 = \text{Third Joint Torque Equation from the } \tau \text{ Matrix}$

$$\tau_3 =$$

$$\begin{aligned} & \theta_3 (m_3 l_{c3}^2 \cos(\theta_2 + \theta_3)^2 + m_3 l_{c3}^2 \sigma_4 \cos(\theta_1)^2 + m_3 l_{c3}^2 \sigma_4 \sin(\theta_1)^2 + I_{c3}) + \theta_3^2 \sigma_1 \\ & + \theta_2^2 (\sigma_8 + \sigma_7 - l_{c3} m_3 \cos(\theta_2 + \theta_3) \sigma_{12}) + \theta_2 \sigma_5 + \theta_1^2 (\sigma_8 + \sigma_7) + 2 \theta_2 \theta_3 \sigma_1 \\ & + g l_{c3} m_3 \cos(\theta_2 + \theta_3) \end{aligned}$$

Design criteria for the maximum (the most challenging scenario) pose of the arm as shown below:

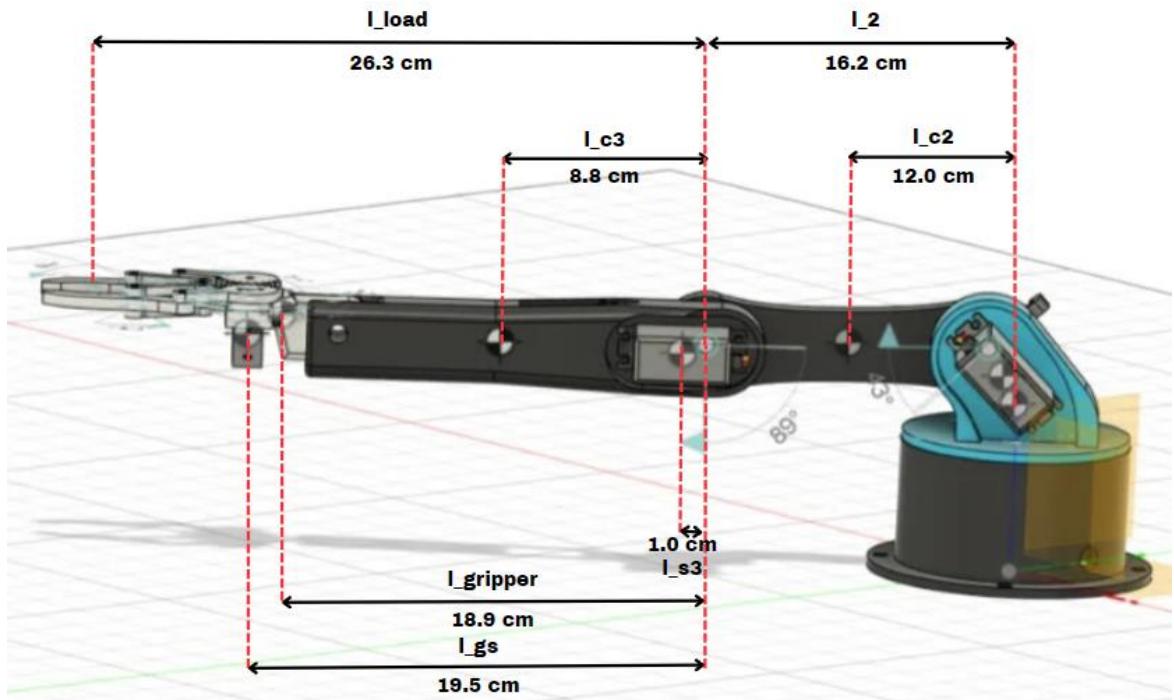


Figure 3.2 Length & Distance Parameters in Robot Arm

$$\theta_1 = 0 \text{ rad}, \theta_2 = 0 \text{ rad}, \theta_3 = 0 \text{ rad}$$

$$\dot{\theta}_1 = 1 \frac{\text{rad}}{\text{s}}, \dot{\theta}_2 = 1 \frac{\text{rad}}{\text{s}}, \dot{\theta}_3 = 1 \frac{\text{rad}}{\text{s}}$$

$$\ddot{\theta}_1 = 0.81 \frac{\text{rad}}{\text{s}^2}, \ddot{\theta}_2 = 0.8 \frac{\text{rad}}{\text{s}^2}, \ddot{\theta}_3 = 0.8 \frac{\text{rad}}{\text{s}^2}$$

$$l_{c1} = 0 \text{ cm}, l_{c2} = 11.97 \text{ cm}, l_{c3} = 0.09, l_2 = 16.2 \text{ cm}$$

$$l_{s1} = 0 \text{ cm}, l_{s2} = 0 \text{ cm}, l_{s3} = 0.97 \text{ cm},$$

$$l_{\text{load}} = 26.3 \text{ cm}, l_{\text{gs}} = 19.5 \text{ cm}, l_g = 18.9 \text{ cm}$$

$$m_{l_2} = 83 \text{ g}, m_{l_3} = 145.4 \text{ g}$$

$$m_{\text{load}} = 300 \text{ g}, m_s = 55 \text{ g}, m_{\text{gs}} = 25 \text{ g}$$

$$Ic_1 = 0.000072 \text{ kg.m}^2, Ic_2 = 0.0001454 \text{ kg.m}^2, Ic_3 = 0.0071 \text{ kg.m}^2,$$

$$Ics_1 = 0.00006 \text{ kg.m}^2$$

$$\tau = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} = \begin{bmatrix} 0.05 \\ 1.25 \\ 0.85 \end{bmatrix} \text{ Nm}$$

Based on these torque values:

- We've selected the needed actuators for each joint on the robot arms.
- We've conducted the stress analysis on the mechanical design part.

### 3.2 Bilateral Teleoperation Calculations

*For the Slave Arm:*

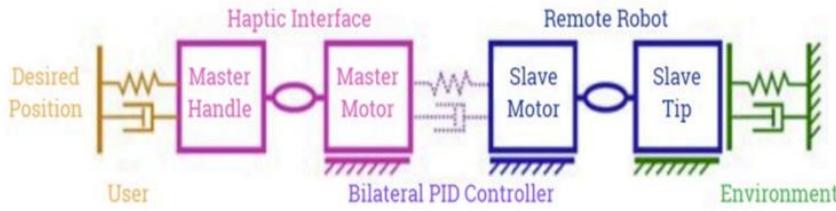
$$f_s(t) = k_{ps}(x_m - x_s) + k_{ds}(\dot{x}_m - \dot{x}_s) + k_{is} \int_0^t (x_m - x_s) dt$$

*For the Master Arm:*

$$f_m(t) = k_{pm}(x_s - x_m) + k_{dm}(\dot{x}_s - \dot{x}_m) + k_{im} \int_0^t (x_s - x_m) dt$$

*Bilateral Teleoperation (Position Forward, Force Feedback):*

$$f_m(t) = f_{env}$$



### 3.3 Power Calculations and Battery Selection for Electronics

$$V_{Mg995} = 7.2V, I_{Mg995R_{stall-current}} = 1.2A$$

$$V_{SG90} = 7.2V, I_{SG90_{stall-current}} = 250 mA$$

$$V_{AS5600} = 3.3V, I_{AS5600_{max-current}} = 10 mA$$

$$V_{NRF24L01} = 3.3V, I_{NRF24L01_{max-current}} = 115 mA$$

$$V_{Raspberry-Pi-Pico} = 3.3V, I_{Raspberry-Pi-Pico_{max-current}} = 300 mA$$

In our electronic circuit, the highest voltage requirement comes from the servo motors, which operate at 7.2V. Consequently, we designed our battery system to supply this voltage, using 7.4V provided by 2S (two-cell) batteries.

$$V_{Battery} = 7.4V$$

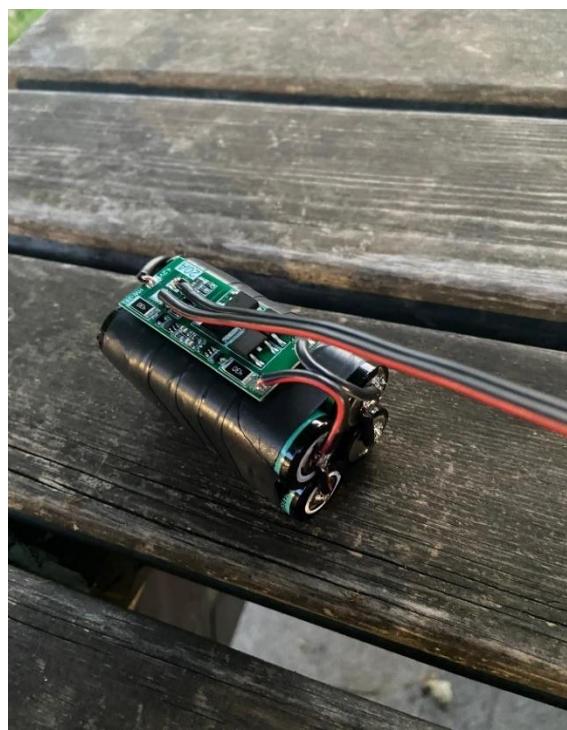
$$3 * I_{Mg995R_{stall-current}} + 1 * I_{SG90_{stall-current}} + 4 * I_{AS5600_{max-current}} + 2 * I_{NRF24L01_{max-current}} + 2 * I_{Raspberry-Pi-Pico_{max-current}} = 4.72A$$

According to our calculations, the maximum current requirement for our device in the project amounts to 4.72 Amperes. Therefore, we tailored our battery selection to accommodate this maximum current by configuring two parallel-connected batteries, each rated at 3.2 Amperes, resulting in a total power output of 6.4 ampere-hours. Additionally, to ensure balanced charging and discharging, we implemented a Battery Management System into the battery design, as the batteries are also connected in series. Additionally, as a result of this configuration, a single arm will be able to operate at maximum performance for approximately 80 minutes with 47.36 Watts of power drawn.

$$I_{Battery} = 6400mAh$$

$$P_{Battery} = I_{Battery} * V_{Battery}$$

$$P_{Battery} = 47.36W$$

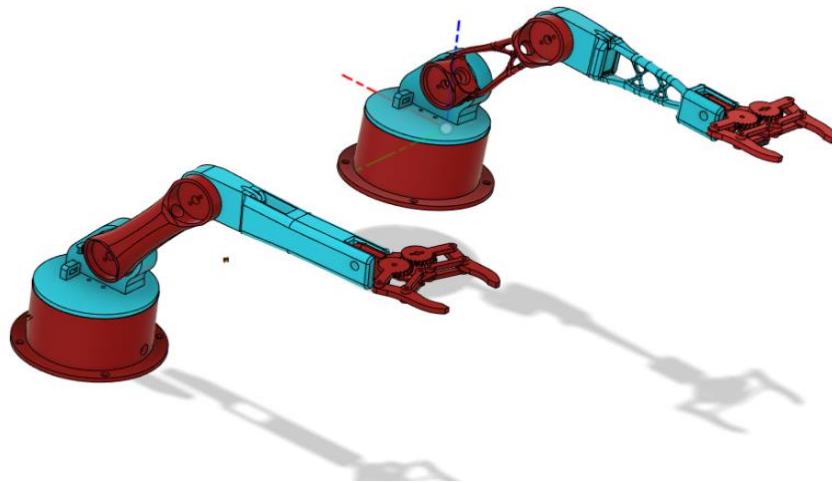


**Figure 3.3 Compact Design of the Battery According to our Calculations**

## 4. DESIGN

### 4.1 Mechanical Design

Length values associated with torque values were defined based on dynamic equations, material selection (PLA), selected actuators, and the design criteria due to the desired technical requirements. Using this information, robot arms were designed and drawn in Autodesk Fusion 360, as shown below. The controller arm, depicted on the left, serves as the master, while the one on the right is the slave to be controlled. The robot parts designed will be 3D printed with PLA material using a 3D printer.



**Figure 4.1 Mechanical Design of Master & Slave Arm in Fusion 360**

#### 4.1.1 Customization

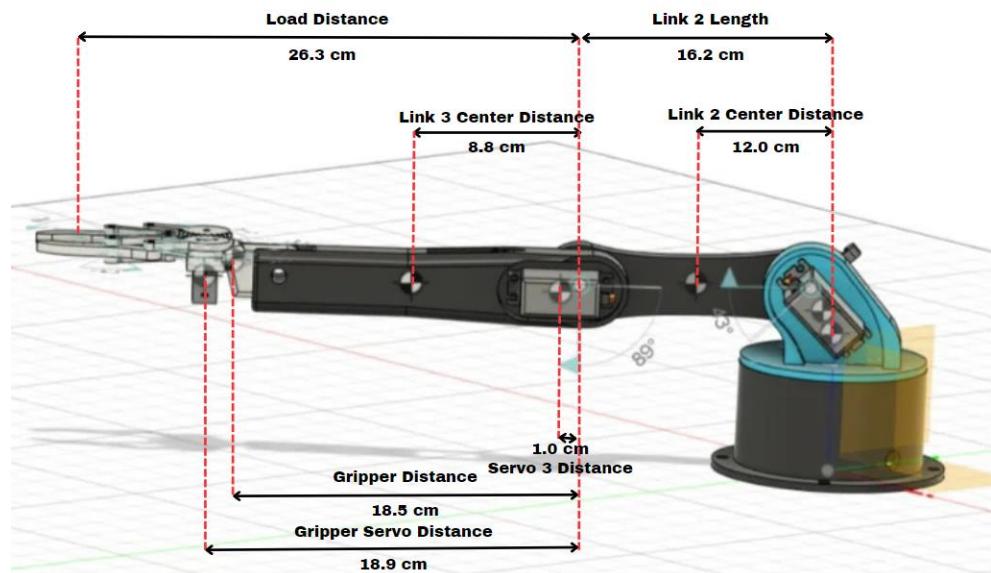
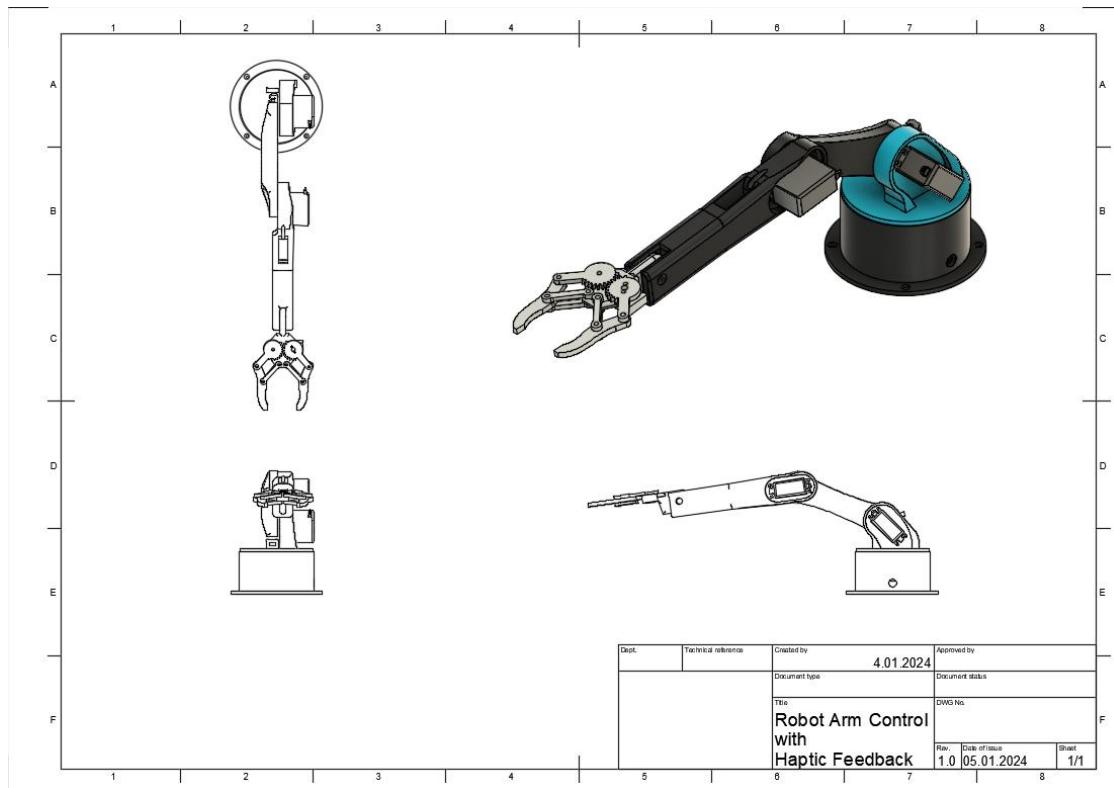


Figure 4.2 Lengths & Distances of the Robot Arm



Figure 4.3 Exploded View of the Robot Arm Assembly



**Figure 4.4 Technical Drawing of the Robot Arm**

#### 4.1.2 Stress Analysis

The torques obtained using dynamic equations, and the weight forces of each component are applied to the joints in the robot arm. This way, the stress analysis of the resulting stresses in the components was successfully performed in the Simulation section of the Autodesk Fusion 360 program. The desired safe results were achieved through a few iterations by changing the design dimensions.

#### Stress Analysis for the Normal Robot Arm (Master Arm)

In this segment, constraints, loads, contacts, meshing, and the outcomes of stress analysis will be delved into. The application of constraints, such as fixed boundaries or prescribed displacements, to the model to simulate real-world conditions will be explored. Additionally,

the examination of the loads, which are represented by external forces or pressures acting on the structure or component under analysis, will be conducted. Contacts will be considered in terms of the interaction between different parts of the model, and how they are brought into contact and affect each other. Meshing will involve the division of the geometry into discrete elements to facilitate numerical analysis. Finally, the stress distribution across the model will be analyzed, and the implications of the results obtained from the stress analysis will be discussed.

## Constraints of the Master Arm

<input type="checkbox"/> Load Case1								
<input type="checkbox"/> Constraints								
<input type="checkbox"/> Fixed1								
<table border="1"><tr><td>Type</td><td>Fixed</td></tr><tr><td>Ux</td><td>Fixed</td></tr><tr><td>Uy</td><td>Fixed</td></tr><tr><td>Uz</td><td>Fixed</td></tr></table>	Type	Fixed	Ux	Fixed	Uy	Fixed	Uz	Fixed
Type	Fixed							
Ux	Fixed							
Uy	Fixed							
Uz	Fixed							
<input type="checkbox"/> Selected Entities								



Figure 4.5 Constraints of the Master Arm

## Loads of the Master Arm

### □ Loads

#### □ Gravity

Type	Gravity
Magnitude	9.807 m / s <sup>2</sup>
X Value	0.00 m / s <sup>2</sup>
Y Value	0.00 m / s <sup>2</sup>
Z Value	-9.807 m / s <sup>2</sup>

#### □ Selected Entities



Figure 4.6 Gravity Force for the Master Arm

### □ Force1

Type	Force
Magnitude	3.00 N
X Value	0.272 N
Y Value	0.266 N
Z Value	-2.976 N
Force Per Entity	No

#### □ Selected Entities



Figure 4.7 Load for the Master Arm

## Contacts of the Master Arm

### □ Contacts

#### □ Bonded

Name
[S] Bonded11 [grip link 1:4  Gripper 1:1]
[S] Bonded12 [grip link 1:4  Gripper base:1]
[S] Bonded13 [grip link 1:3  Gripper 1:2]
[S] Bonded14 [grip link 1:3  Gripper base:1]
[S] Bonded15 [grip link 1:2  Gripper 1:1]
[S] Bonded16 [grip link 1:2  Gripper base:1]
[S] Bonded17 [grip link 1:1  Gripper 1:2]
[S] Bonded18 [grip link 1:1  Gripper base:1]
[S] Bonded19 [gear2:1  Gripper 1:1]
[S] Bonded20 [gear2:1  Gripper base:1]
[S] Bonded21 [gear1:1  gear2:1]
[S] Bonded22 [gear1:1  gear2:1]
[S] Bonded23 [gear1:1  Gripper 1:2]
[S] Bonded24 [gear1:1  Gripper base:1]
[S] Bonded25 [gear1:1  Gripper base:1]
[S] Bonded26 [Arm 02 v3:1  Gripper base:1]
[S] Bonded27 [Arm 02 v3:1  Gripper base:1]
[S] Bonded28 [Arm 02 v3:1  Gripper base:1]
[S] Bonded29 [Arm 02 v3:1  Gripper base:1]
[S] Bonded30 [Arm 02 v3:1  Gripper base:1]
[S] Bonded31 [Arm 02 v3:1  Gripper base:1]
[S] Bonded32 [Arm 02 v3:1  Gripper base:1]
[S] Bonded40 [Arm 01:1(Body2)  Arm 01:1(Body1)]
[S] Bonded46 [Base:1  Waist:1]

#### □ Offset Bonded

Name
[M] [S] Offset Bonded47 [Waist:1  Arm 01:1(Body1)]
[M] [S] Offset Bonded48 [Arm 01:1(Body1)  Arm 02 v3:1]

Figure 4.8 Contacts of the Master Arm



Figure 4.9 Offset Bonded

## Mesh of the Master Arm

**Mesh**

Average Element Size (% of model size)	
Solids	10
Scale Mesh Size Per Part	No
Average Element Size (absolute value)	-
Element Order	Parabolic
Create Curved Mesh Elements	No
Max. Turn Angle on Curves (Deg.)	60
Max. Adjacent Mesh Size Ratio	1.5
Max. Aspect Ratio	10
Minimum Element Size (% of average size)	20

**Mesh**

Type	Nodes	Elements
Solids	100890	57875

**Adaptive Mesh Refinement**

Number of Refinement Steps	0
Results Convergence Tolerance (%)	20
Portion of Elements to Refine (%)	10
Results for Baseline Accuracy	von Mises Stress

Figure 4.10 Mesh of the Master Arm

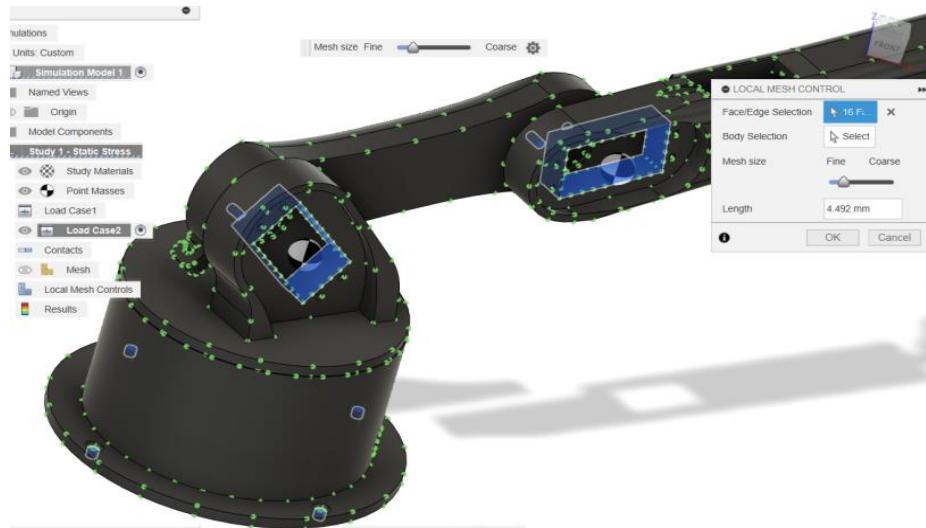
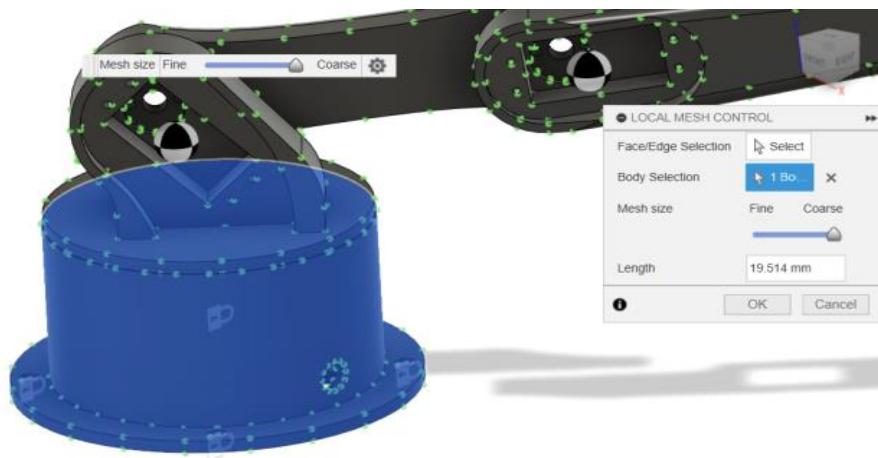
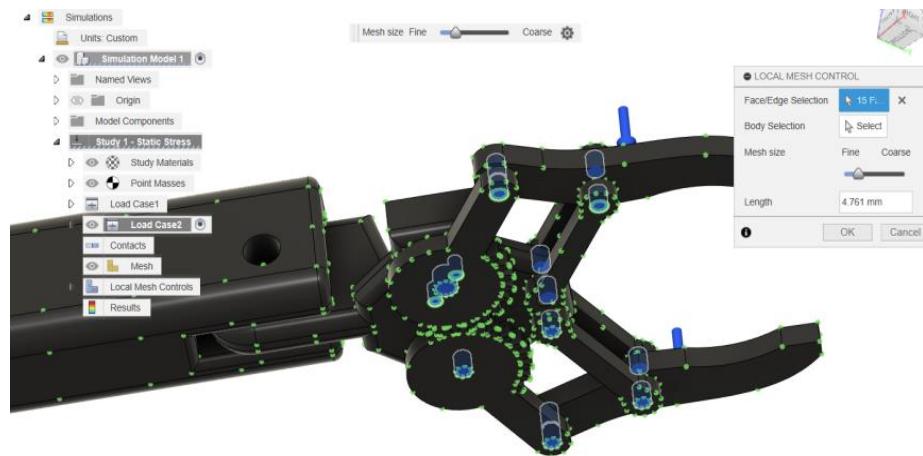


Figure 4.11 Local Mesh Control



**Figure 4.12 Local Mesh of the Base**



**Figure 4.13 Local Mesh of the Gripper**



**Figure 4.14 Mesh & Loads**

## Results of the Master Arm

### □ Materials

Component	Material	Safety Factor
Base:1	Nylon 12 (with Formlabs Fuse 1 3D Printer)	Yield Strength
Waist:1	Nylon 12 (with Formlabs Fuse 1 3D Printer)	Yield Strength
Arm 01:1/Body2	Nylon 12 (with Formlabs Fuse 1 3D Printer)	Yield Strength
Arm 01:1/Body1	Nylon 12 (with Formlabs Fuse 1 3D Printer)	Yield Strength
Arm 02 v3:1	Nylon 12 (with Formlabs Fuse 1 3D Printer)	Yield Strength
Gripper Assembly:1/gear1:1	Nylon 12 (with Formlabs Fuse 1 3D Printer)	Yield Strength
Gripper Assembly:1/gear2:1	Nylon 12 (with Formlabs Fuse 1 3D Printer)	Yield Strength
Gripper Assembly:1/grip link 1:1	Nylon 12 (with Formlabs Fuse 1 3D Printer)	Yield Strength
Gripper Assembly:1/grip link 1:2	Nylon 12 (with Formlabs Fuse 1 3D Printer)	Yield Strength
Gripper Assembly:1/grip link 1:3	Nylon 12 (with Formlabs Fuse 1 3D Printer)	Yield Strength
Gripper Assembly:1/grip link 1:4	Nylon 12 (with Formlabs Fuse 1 3D Printer)	Yield Strength
Gripper Assembly:1/Gripper base:1	Nylon 12 (with Formlabs Fuse 1 3D Printer)	Yield Strength
Gripper Assembly:1/Gripper 1:1	Nylon 12 (with Formlabs Fuse 1 3D Printer)	Yield Strength
Gripper Assembly:1/Gripper 1:2	Nylon 12 (with Formlabs Fuse 1 3D Printer)	Yield Strength

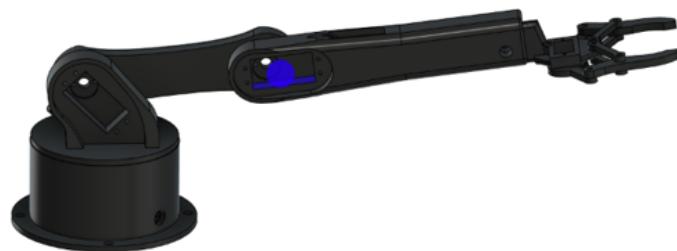
### □ Nylon 12 (with Formlabs Fuse 1 3D Printer)

Density	1.015E-06 kg / mm <sup>3</sup>
Young's Modulus	1850.00 MPa
Poisson's Ratio	0.35
Yield Strength	46.00 MPa
Ultimate Tensile Strength	50.00 MPa
Thermal Conductivity	3.500E-04 W / (mm C)
Thermal Expansion Coefficient	1.275E-04 / C
Specific Heat	1830.00 J / (kg C)

Figure 4.15 Materials of the Master Arm

**Point Mass1**

Type	Point Mass (Auto)
Mass	0.055 kg

 **Selected Entities**

**Figure 4.16 Point Mass for Servo Motor**

 **Point Mass2**

Type	Point Mass (Auto)
Mass	0.055 kg

 **Selected Entities**

**Figure 4.17 Point Mass for Servo Motor**

---

**□ Results**
**□ Result Summary**

Name	Minimum	Maximum
<b>Safety Factor</b>		
Safety Factor (Per Body)	15.00	15.00
<b>Stress</b>		
von Mises	2.384E-06 MPa	3.048 MPa
1st Principal	-1.298 MPa	2.941 MPa
3rd Principal	-4.663 MPa	0.639 MPa
Normal XX	-3.421 MPa	1.59 MPa
Normal YY	-2.541 MPa	1.99 MPa
Normal ZZ	-2.087 MPa	1.253 MPa
Shear XY	-0.825 MPa	0.559 MPa
Shear YZ	-0.478 MPa	0.994 MPa
Shear ZX	-0.465 MPa	1.293 MPa
<b>Displacement</b>		
Total	0.00 mm	1.225 mm
X	-0.026 mm	0.078 mm
Y	-0.002 mm	0.193 mm
Z	-1.216 mm	0.031 mm
<b>Reaction Force</b>		
Total	0.00 N	1.984 N
X	-1.533 N	1.477 N
Y	-1.449 N	1.31 N
Z	-0.97 N	1.365 N
<b>Strain</b>		
Equivalent	0.00	0.003
1st Principal	0.00	0.002
3rd Principal	-0.003	0.00
Normal XX	-0.001	7.340E-04
Normal YY	-8.495E-04	6.566E-04
Normal ZZ	-4.175E-04	3.586E-04
Shear XY	-0.001	8.162E-04
Shear YZ	-6.976E-04	0.001
Shear ZX	-6.782E-04	0.002
<b>Contact Pressure</b>		
Total	0.00 MPa	0.687 MPa
X	-0.30 MPa	0.223 MPa
Y	-0.302 MPa	0.226 MPa
Z	-0.418 MPa	0.648 MPa
<b>Contact Force</b>		
Total	0.00 N	7.693 N
X	-4.96 N	3.217 N
Y	-3.276 N	3.586 N
Z	-7.659 N	6.19 N

**Figure 4.18 Result Summary**

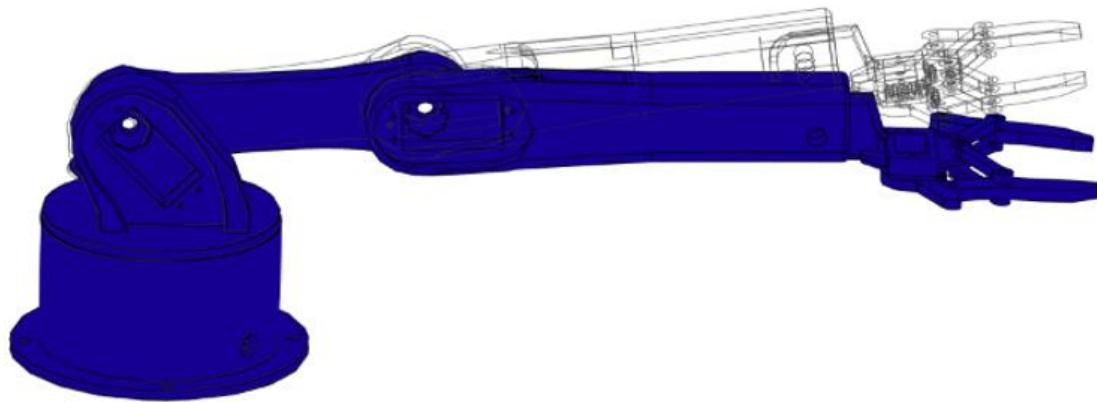
Safety Factor Safety Factor (Per Body)0.00  8.00

Figure 4.19 Safety Factor

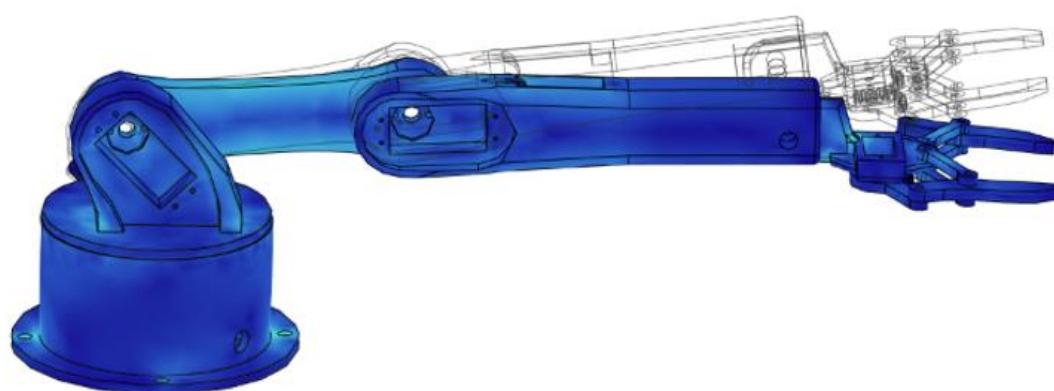
 Stress von Mises[MPa] 0.00  3.048

Figure 4.20 Stress (von Mises)

⊖ 1st Principal  
[MPa] -1.298 2.941

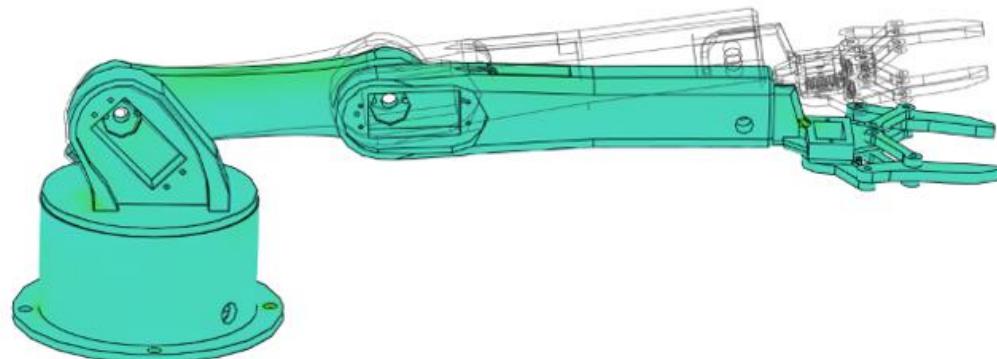


Figure 4.21 1<sup>st</sup> Principal

⊖ 3rd Principal  
[MPa] -4.663 0.639

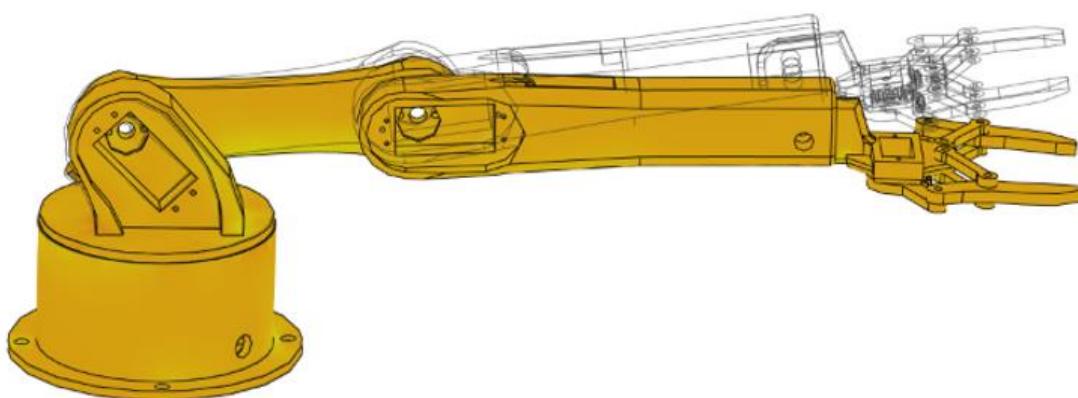
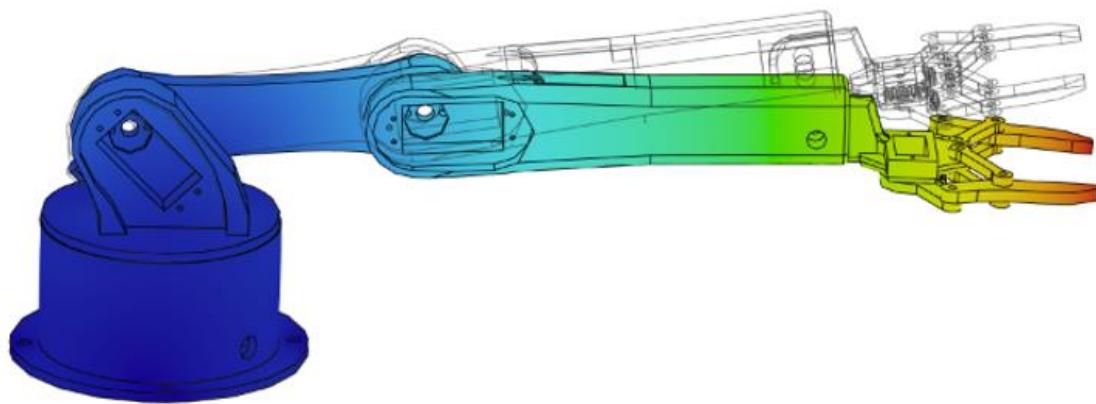


Figure 4.22 3<sup>rd</sup> Principal

**□ Displacement****□ Total**[mm] 0.00  1.225**Figure 4.23 Displacement****Stress Analysis for the Generative Robot Arm (Slave Arm)**

In this segment, constraints, loads, contacts, meshing, and the outcomes of stress analysis will be delved into. The application of constraints, such as fixed boundaries or prescribed displacements, to the model to simulate real-world conditions will be explored. Additionally, the examination of the loads, which are represented by external forces or pressures acting on the structure or component under analysis, will be conducted. Contacts will be considered in terms of the interaction between different parts of the model, and how they are brought into contact and affect each other. Meshing will involve the division of the geometry into discrete elements to facilitate numerical analysis. Finally, the stress distribution across the model will be analyzed, and the implications of the results obtained from the stress analysis will be discussed.

## Constraints of the Slave Arm

### ▫ Constraints

#### ▫ Fixed1

Type	Fixed
Ux	Fixed
Uy	Fixed
Uz	Fixed

#### ▫ Selected Entities



Figure 4.24 Constraints for the Slave Arm

## Loads of the Slave Arm

### □ Loads

#### □ Gravity

Type	Gravity
Magnitude	9.807 m / s <sup>2</sup>
X Value	0.00 m / s <sup>2</sup>
Y Value	0.00 m / s <sup>2</sup>
Z Value	-9.807 m / s <sup>2</sup>

#### □ Selected Entities



Figure 4.25 Gravity Force for the Slave Arm

### □ Force1

Type	Force
Magnitude	3.00 N
X Value	0.00 N
Y Value	0.00 N
Z Value	-3.00 N
Force Per Entity	No

#### □ Selected Entities



Figure 4.26 Load for the Slave Arm

## Contacts of the Slave Arm

### □ Contacts

#### □ Bonded

Name
[S] Bonded1 [Gripper base:1  arm2_pla_v1_2 v1:1]
[S] Bonded2 [Gripper base:1  arm2_pla_v1_2 v1:1]
[S] Bonded3 [Gripper base:1  arm2_pla_v1_2 v1:1]
[S] Bonded4 [Gripper base:1  Servo Motor Micro 9g v1:1(Body2)]
[S] Bonded5 [Gripper base:1  Servo Motor Micro 9g v1:1(Body1)]
[S] Bonded6 [grip link 1:4  Gripper 1:1]
[S] Bonded7 [grip link 1:4  Gripper base:1]
[S] Bonded8 [grip link 1:3  Gripper 1:2]
[S] Bonded9 [grip link 1:3  Gripper base:1]
[S] Bonded10 [grip link 1:2  Gripper 1:1]
[S] Bonded11 [grip link 1:2  Gripper base:1]
[S] Bonded12 [grip link 1:1  Gripper 1:2]
[S] Bonded13 [grip link 1:1  Gripper base:1]
[S] Bonded14 [gear2:1  Gripper 1:1]
[S] Bonded15 [gear2:1  Gripper base:1]
[S] Bonded16 [gear1:1  gear2:1]
[S] Bonded17 [gear1:1  Gripper 1:2]
[S] Bonded18 [gear1:1  Gripper base:1]
[S] Bonded19 [Base:1  Waist:1]

#### □ Offset Bonded

Name
[M] [S] Offset Bonded22 [Waist:1  generative arm2_2 v3:1]
[M] [S] Offset Bonded23 [generative arm2_2 v3:1  arm2_pla_v1_2 v1:1]

Figure 4.27 Contacts of the Slave Arm

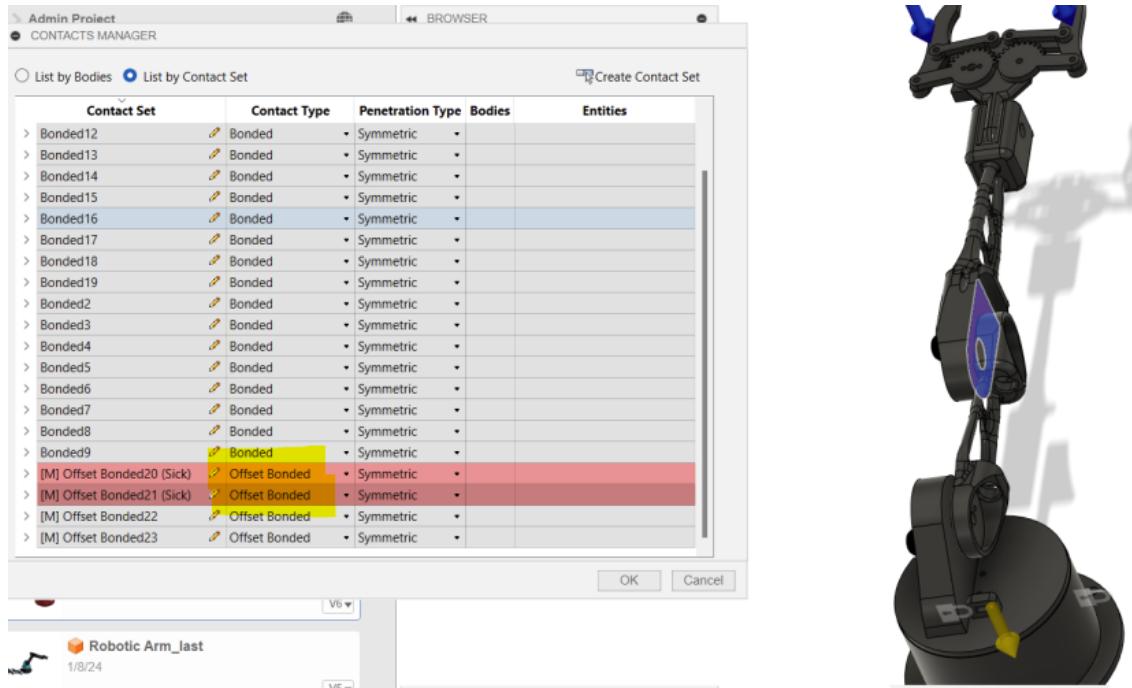


Figure 4.28 Contacts for the Slave Arm

## Mesh of the Slave Arm

Mesh	
Average Element Size (% of model size)	
Solids	10
Scale Mesh Size Per Part	No
Average Element Size (absolute value)	-
Element Order	Parabolic
Create Curved Mesh Elements	Yes
Max. Turn Angle on Curves (Deg.)	60
Max. Adjacent Mesh Size Ratio	1.5
Max. Aspect Ratio	10
Minimum Element Size (% of average size)	20

Adaptive Mesh Refinement	
Number of Refinement Steps	0
Results Convergence Tolerance (%)	20
Portion of Elements to Refine (%)	10
Results for Baseline Accuracy	von Mises Stress

Mesh		
Type	Nodes	Elements
Solids	118921	68945

Figure 4.29 Mesh of the Slave Arm

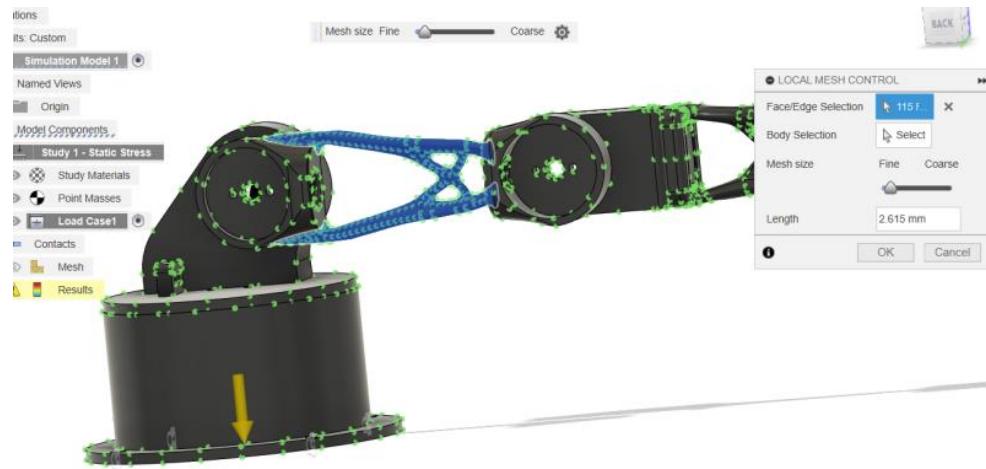


Figure 4.30 Local Mesh Control

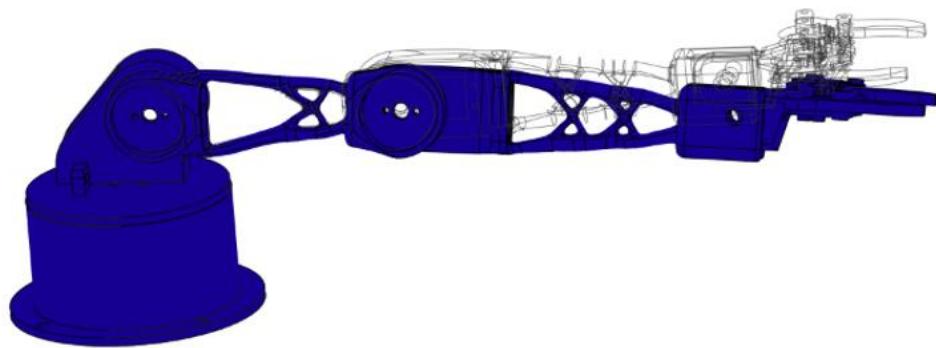
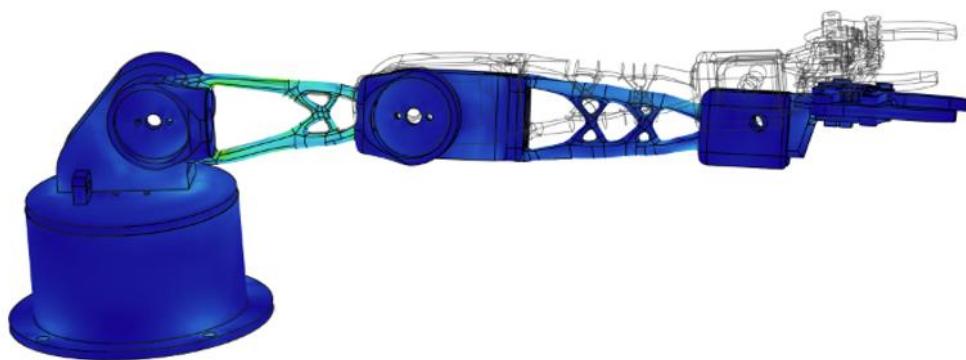
## Results of the Slave Arm

### □ Results

#### □ Result Summary

Name	Minimum	Maximum
<b>Safety Factor</b>		
Safety Factor (Per Body)	5.497	15.00
<b>Stress</b>		
von Mises	4.104E-05 MPa	8.368 MPa
1st Principal	-1.086 MPa	9.564 MPa
3rd Principal	-9.439 MPa	0.97 MPa
Normal XX	-9.069 MPa	9.252 MPa
Normal YY	-2.345 MPa	2.147 MPa
Normal ZZ	-2.083 MPa	2.295 MPa
Shear XY	-2.122 MPa	1.86 MPa
Shear YZ	-1.018 MPa	0.852 MPa
Shear ZX	-2.516 MPa	1.733 MPa
<b>Displacement</b>		
Total	0.00 mm	5.338 mm
X	-0.545 mm	0.331 mm
Y	-2.383 mm	0.319 mm
Z	-4.803 mm	0.37 mm
<b>Reaction Force</b>		
Total	0.00 N	1.978 N
X	-1.403 N	1.373 N
Y	-1.15 N	1.365 N
Z	-0.679 N	1.224 N
<b>Strain</b>		
Equivalent	0.00	0.005
1st Principal	-7.090E-06	0.005
3rd Principal	-0.005	2.045E-05
Normal XX	-0.004	0.004
Normal YY	-0.001	0.001
Normal ZZ	-0.002	0.002
Shear XY	-0.003	0.003
Shear YZ	-0.001	0.001
Shear ZX	-0.004	0.003
<b>Contact Pressure</b>		
Total	0.00 MPa	1.182 MPa
X	-1.022 MPa	1.011 MPa
Y	-0.889 MPa	0.761 MPa
Z	-0.989 MPa	0.746 MPa
<b>Contact Force</b>		
Total	0.00 N	6.723 N
X	-3.52 N	3.685 N
Y	-2.461 N	2.23 N
Z	-4.369 N	6.403 N

Figure 4.31 Result Summary

Safety Factor Safety Factor (Per Body)  
0.00  8.00**Figure 4.32 Safety Factor** Stress von Mises  
[MPa] 0.00  8.368**Figure 4.33 Stress (von Mises)**

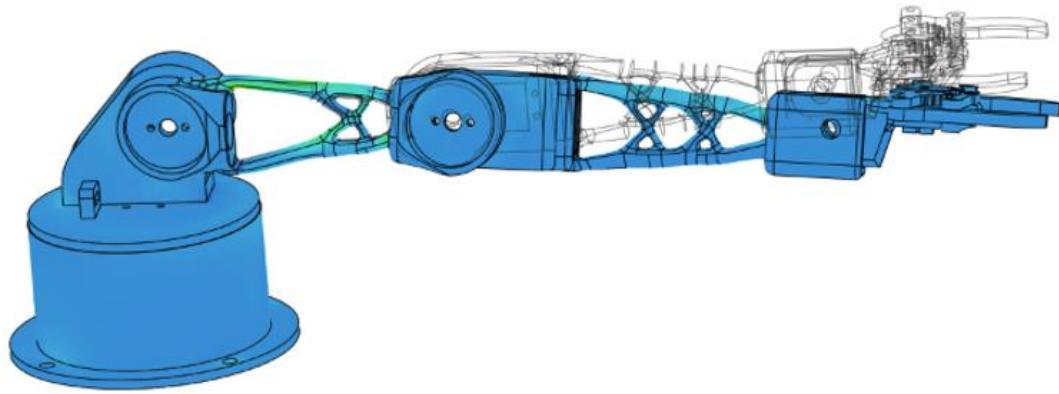


Figure 4.34 1<sup>st</sup> Principal

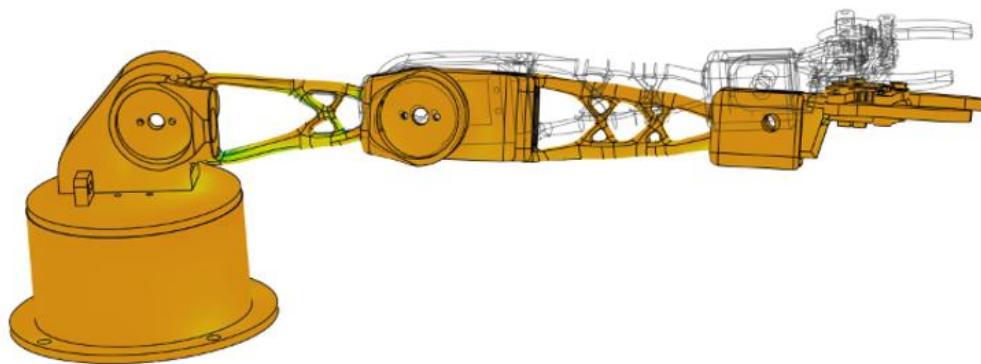


Figure 4.35 3<sup>rd</sup> Principal

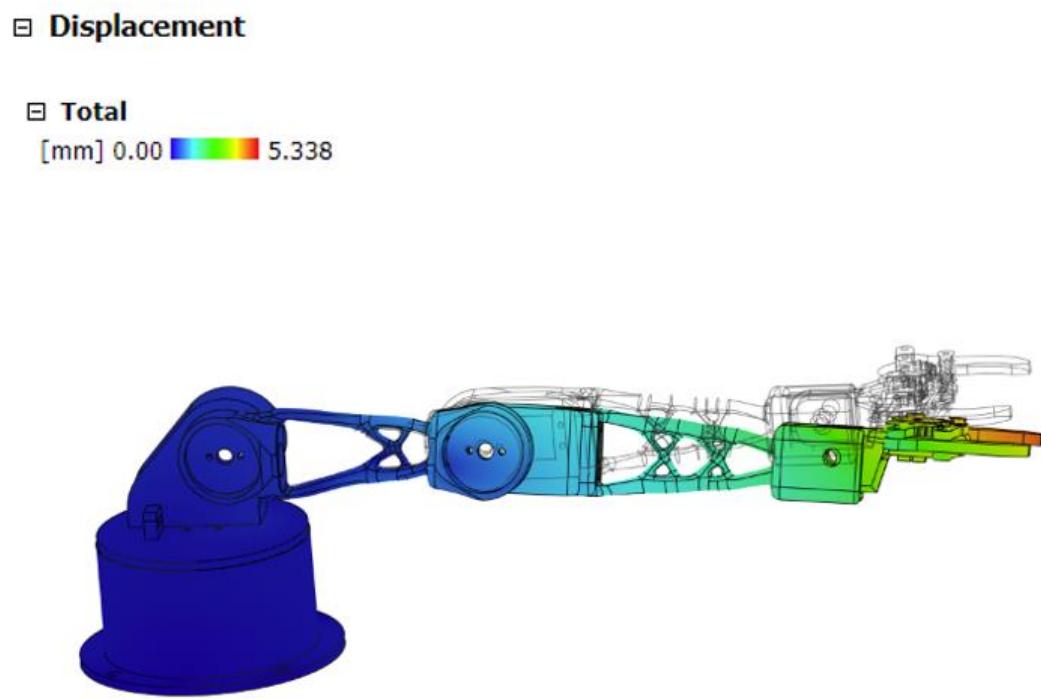


Figure 4.36 Displacement

### Stress Analysis Comparison of the Normal & Generative Designs

As observed below, the outputs on the left pertain to the normal arm (Master Arm), whereas those on the right depict the outcomes derived from the arm crafted using generative design (Slave Arm). In the process of Generative Design, mass optimization was prioritized while maintaining relatively unchanged stiffness. The strength level aligns with the desired criteria.

#### Result Summary

Name	Minimum	Maximum
<b>Safety Factor</b>		
Safety Factor (Per Body)	15.00	15.00
<b>Stress</b>		
von Mises	2.384E-06 MPa	3.048 MPa
1st Principal	-1.298 MPa	2.941 MPa
3rd Principal	-4.663 MPa	0.639 MPa
Normal XX	-3.421 MPa	1.59 MPa
Normal YY	-2.541 MPa	1.99 MPa
Normal ZZ	-2.087 MPa	1.253 MPa
Shear XY	-0.825 MPa	0.559 MPa
Shear YZ	-0.478 MPa	0.994 MPa
Shear ZX	-0.465 MPa	1.293 MPa
<b>Displacement</b>		
Total	0.00 mm	1.225 mm
X	-0.026 mm	0.078 mm
Y	-0.002 mm	0.193 mm
Z	-1.216 mm	0.031 mm
<b>Reaction Force</b>		
Total	0.00 N	1.984 N
X	-1.533 N	1.477 N
Y	-1.449 N	1.31 N
Z	-0.97 N	1.365 N
<b>Strain</b>		
Equivalent	0.00	0.003
1st Principal	0.00	0.002
3rd Principal	-0.003	0.00
Normal XX	-0.001	7.340E-04
Normal YY	-8.495E-04	6.566E-04
Normal ZZ	-4.175E-04	3.586E-04
Shear XY	-0.001	8.162E-04
Shear YZ	-6.976E-04	0.001
Shear ZX	-6.782E-04	0.002
<b>Contact Pressure</b>		
Total	0.00 MPa	0.687 MPa
X	-0.30 MPa	0.223 MPa
Y	-0.302 MPa	0.226 MPa
Z	-0.418 MPa	0.648 MPa
<b>Contact Force</b>		
Total	0.00 N	7.693 N
X	-4.96 N	3.217 N
Y	-3.276 N	3.586 N
Z	-7.659 N	6.19 N

#### Result Summary

Name	Minimum	Maximum
<b>Safety Factor</b>		
Safety Factor (Per Body)	5.497	15.00
<b>Stress</b>		
von Mises	4.104E-05 MPa	8.368 MPa
1st Principal	-1.086 MPa	9.564 MPa
3rd Principal	-9.439 MPa	0.97 MPa
Normal XX	-9.069 MPa	9.252 MPa
Normal YY	-2.345 MPa	2.147 MPa
Normal ZZ	-2.083 MPa	2.295 MPa
Shear XY	-2.122 MPa	1.86 MPa
Shear YZ	-1.018 MPa	0.852 MPa
Shear ZX	-2.516 MPa	1.733 MPa
<b>Displacement</b>		
Total	0.00 mm	5.338 mm
X	-0.545 mm	0.331 mm
Y	-2.383 mm	0.319 mm
Z	-4.803 mm	0.37 mm
<b>Reaction Force</b>		
Total	0.00 N	1.978 N
X	-1.403 N	1.373 N
Y	-1.15 N	1.365 N
Z	-0.679 N	1.224 N
<b>Strain</b>		
Equivalent	0.00	0.005
1st Principal	-7.090E-06	0.005
3rd Principal	-0.005	2.045E-05
Normal XX	-0.004	0.004
Normal YY	-0.001	0.001
Normal ZZ	-0.002	0.002
Shear XY	-0.003	0.003
Shear YZ	-0.001	0.001
Shear ZX	-0.004	0.003
<b>Contact Pressure</b>		
Total	0.00 MPa	1.182 MPa
X	-1.022 MPa	1.011 MPa
Y	-0.889 MPa	0.761 MPa
Z	-0.989 MPa	0.746 MPa
<b>Contact Force</b>		
Total	0.00 N	6.723 N
X	-3.52 N	3.685 N
Y	-2.461 N	2.23 N
Z	-4.369 N	6.403 N

Figure 4.37 Normal & Generative Design Comparison

#### 4.1.3 Material Selection

The preference for using PLA material in robot arm design has several advantages. Firstly, its low cost allows for effective budget management in the project. Additionally, PLA material is an environmentally friendly option; its biodegradable feature and sourcing from renewable resources contribute positively to environmental sustainability. Compatibility with 3D printers and a wide range of color options can simplify the design process. Its lightweight structure provides design flexibility and allows for easy printing at low processing temperatures. Its durability makes it suitable for general use and prototype stages. When these advantages come together, it is evident that PLA is an economical and user-friendly option for robot arm design. Therefore, we chose PLA material in our design process. The features of the PLA material added to Fusion 360 can be seen below.

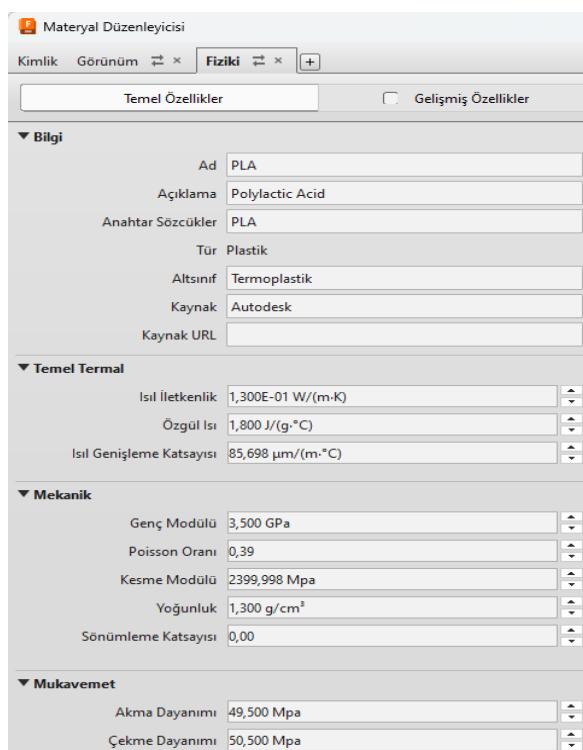


Figure 4.38 PLA Material Properties Shown in Fusion 360 Material Properties Page

In generative design, NYLON 12 material was utilized. The preference for using NYLON 12 material in our generative design robot offers several advantages. NYLON 12 is chosen for its superior strength, durability, and resistance to wear and tear, which are crucial factors in ensuring the reliability and longevity of the robot arm. Additionally, NYLON 12's flexibility allows for intricate designs and complex geometries, enhancing the overall performance and functionality of the robot. Its high melting point and thermal stability make it suitable for applications requiring exposure to high temperatures. Furthermore, NYLON 12's chemical resistance and low moisture absorption contribute to its suitability for various environmental conditions. Overall, NYLON 12 offers a balance of strength, flexibility, and durability, making it an ideal choice for our generative design robot.

#### ⊖ Nylon 12 (with Formlabs Fuse 1 3D Printer)

Density	1.015E-06 kg / mm <sup>3</sup>
Young's Modulus	1850.00 MPa
Poisson's Ratio	0.35
Yield Strength	46.00 MPa
Ultimate Tensile Strength	50.00 MPa
Thermal Conductivity	3.500E-04 W / (mm C)
Thermal Expansion Coefficient	1.275E-04 / C
Specific Heat	1830.00 J / (kg C)

**Figure 4.39 Nylon 12 Properties**

#### 4.1.4 Mechanical Simulation

The design, originally created in SolidWorks, was transferred to Fusion 360. Subsequently, the design was transferred to Autodesk Inventor, which contains the plugin for connection with MATLAB. In Autodesk Inventor, joints and constraints in the robot arm were redefined. After the necessary codes for the Simscape Multibody Link Plugin were entered into the MATLAB console, the required Multibody Plugin was installed in Inventor. This allowed our design, along with joints and constraints, to be exported as a compatible XML file for MATLAB Simulink. The XML file in Simulink was opened with the help of the Multibody plugin, enabling the multibody design shown below to be obtained. In this way, simulations needed in Simscape can be utilized in the control section.

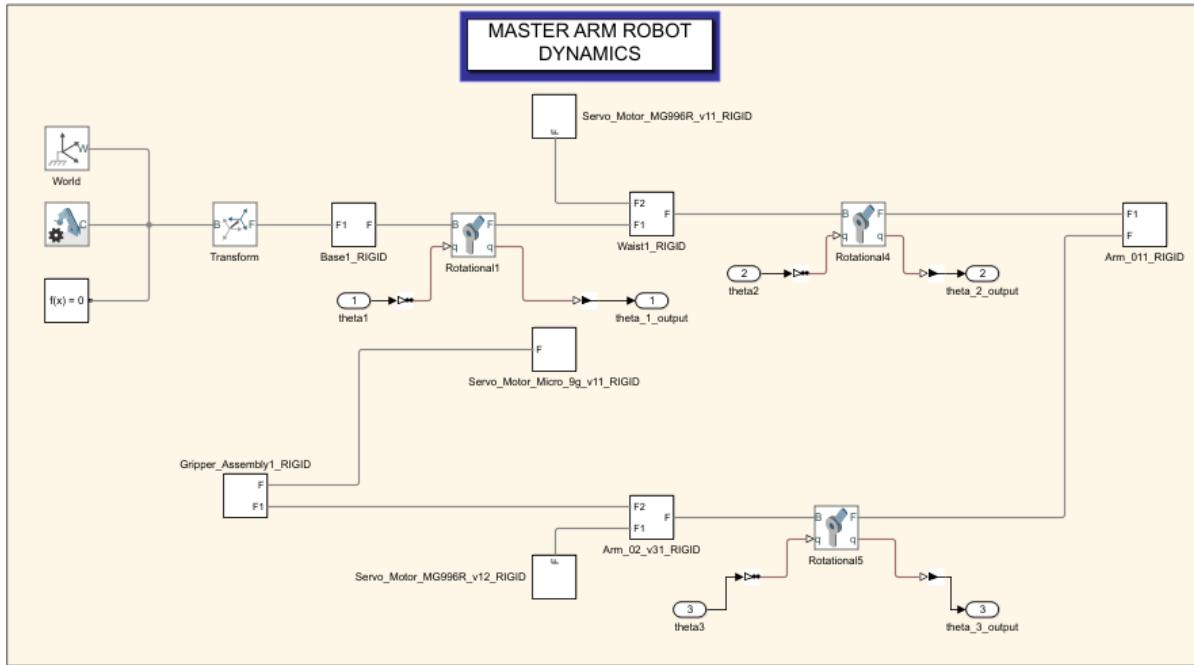


Figure 4.40 Master Robot Arm Multibody Diagram in Simulink

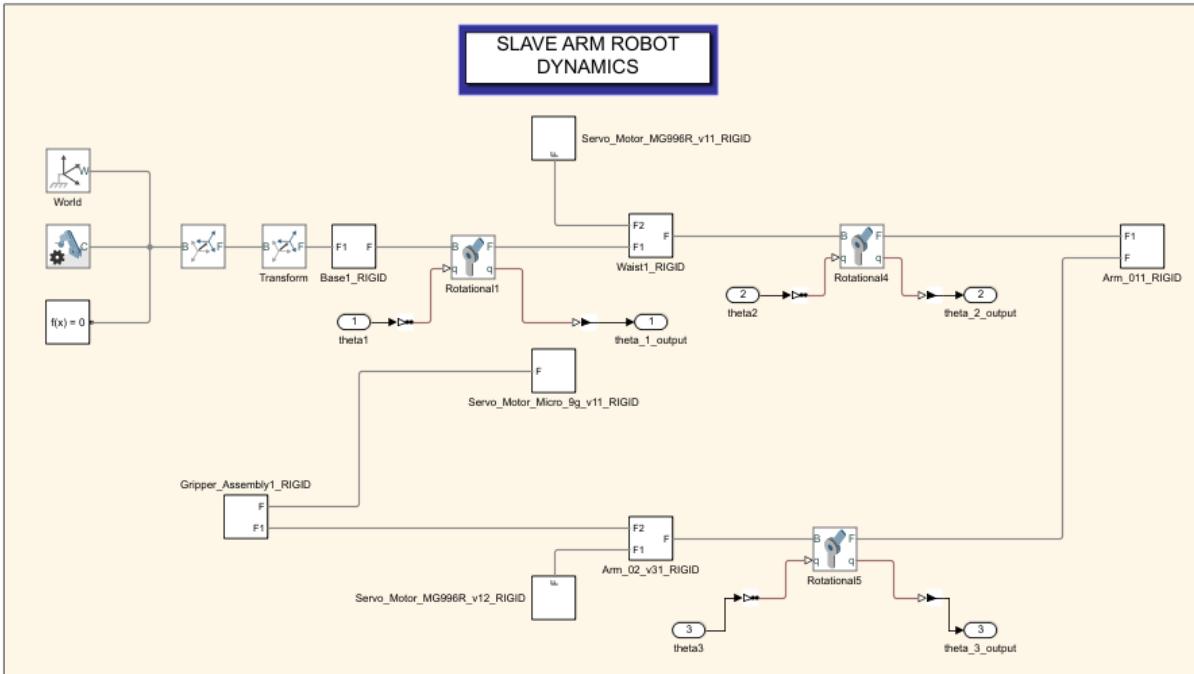
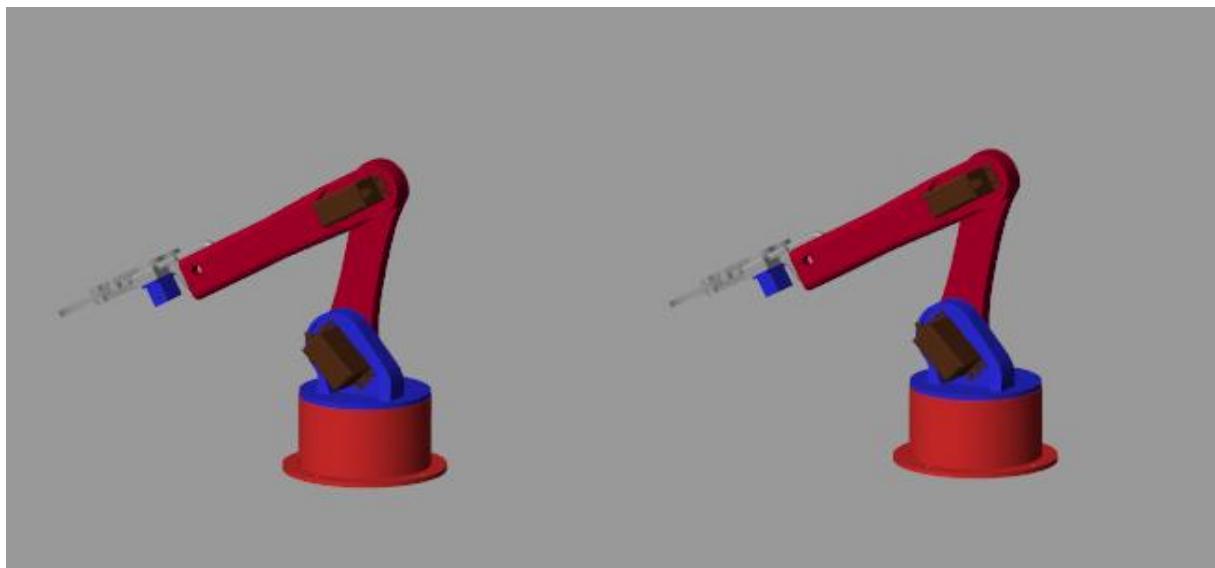


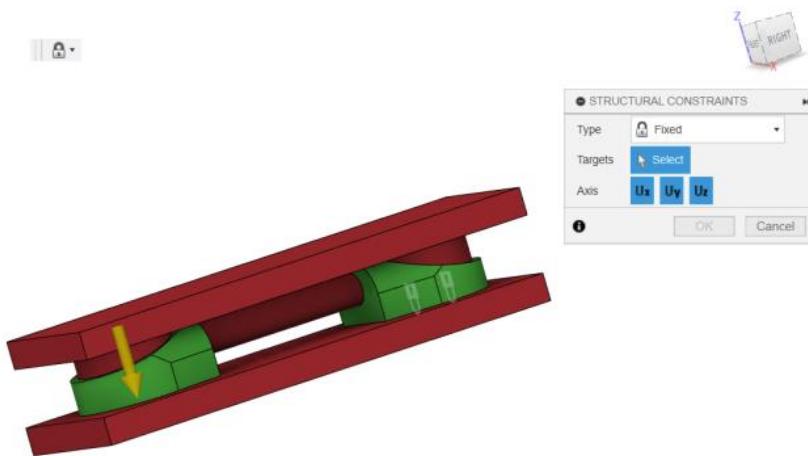
Figure 4.41 Slave Robot Arm Multibody Diagram in Simulink



**Figure 4.42 Multibody Robot Arms Animation Output in Simulink**

#### 4.1.5 Conceptual Design

Utilizing Fusion 360 generative design, we crafted conceptual designs and pinpointed a design that met the specified technical criteria. The groundbreaking generative design process facilitated significant savings of 98 grams of PLA material per robot arm. This translated into a remarkable cost reduction of 300 TL per pair of arms, alongside a notable increase in load capacity by 98 grams. Additionally, the optimized design exhibited lower power consumption and a substantial improvement in overall performance when compared to the initial design.



**Figure 4.43 Second Link Generative Design Process**

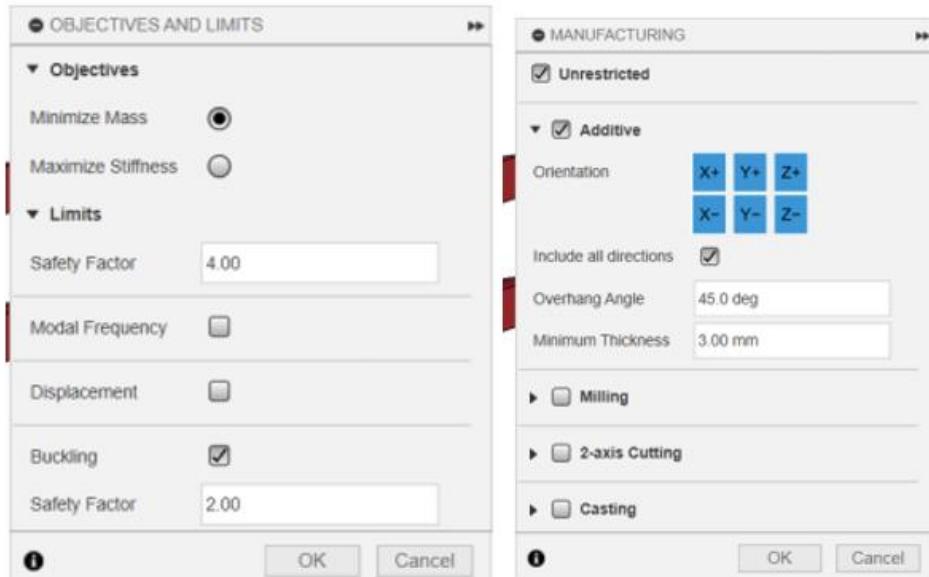


Figure 4.44 Objectives & Limits & Manufacturing

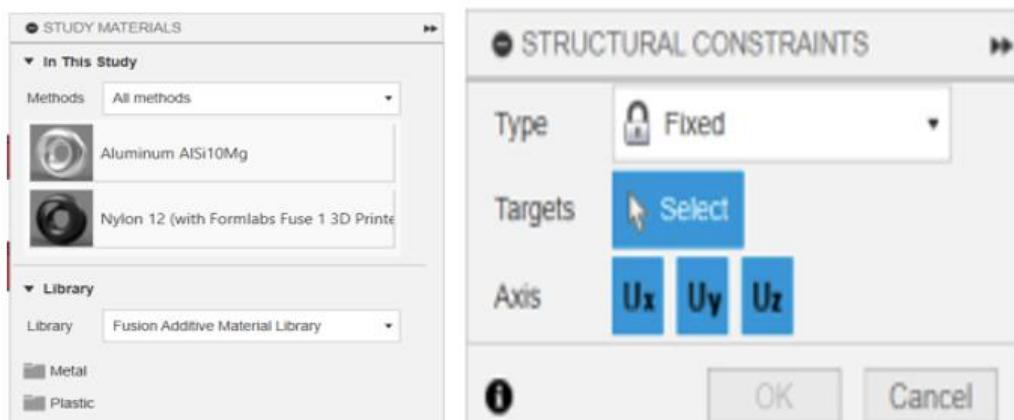
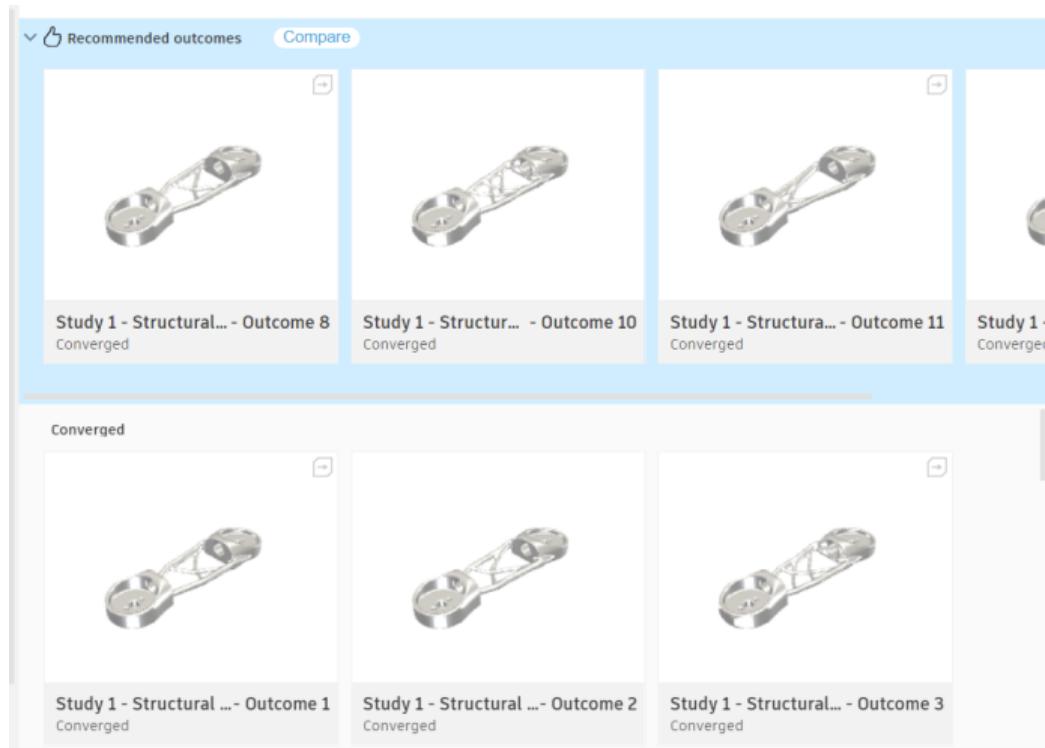


Figure 4.45 Study Materials & Structural Constraints

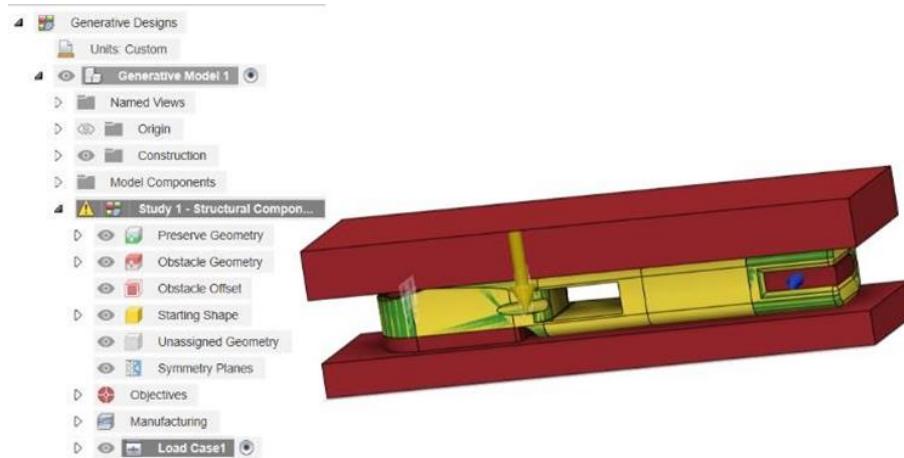


**Figure 4.46 Conceptual Designs**

In this picture conceptual designs are being generated according to design criteria.



**Figure 4.47 Second Link Generative Design Output**



**Figure 4.48 Third Link Generative Design Process**



**Figure 4.49 Safety Factor & Preserved Geometries**

In this picture one of the design criteria which is safety factor to be 4 is shown and the preserve geometries of third link can be seen which are green color.

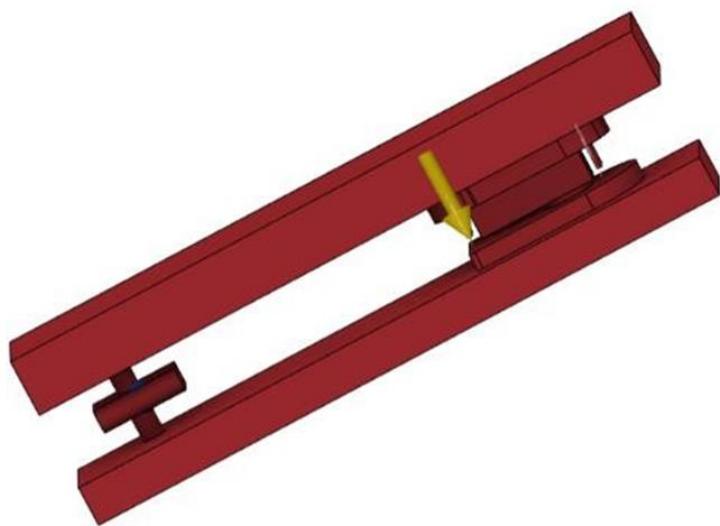
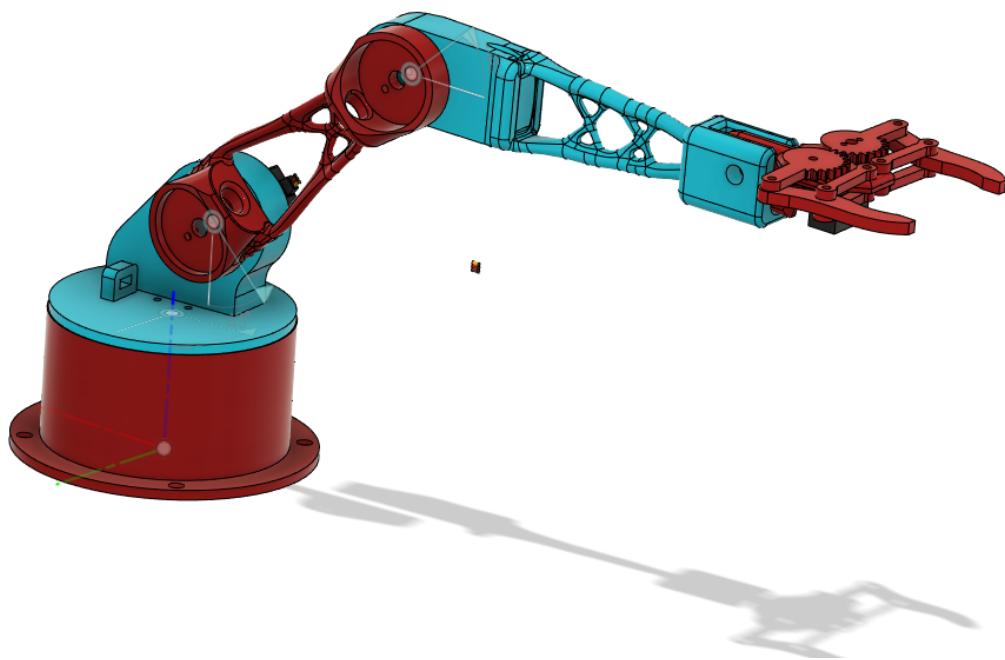


Figure 4.50 Obstacle Geometries for Third Link



Figure 4.51 Generative Design Output for Third Link & Stress Distribution



**Figure 4.52 The Assembly of Generative Design Parts & Its Unique Design**

#### 4.1.6 Mechanical Design Realizations



Figure 4.53 Master Arm



Figure 4.54 Master Arm & its Circuit

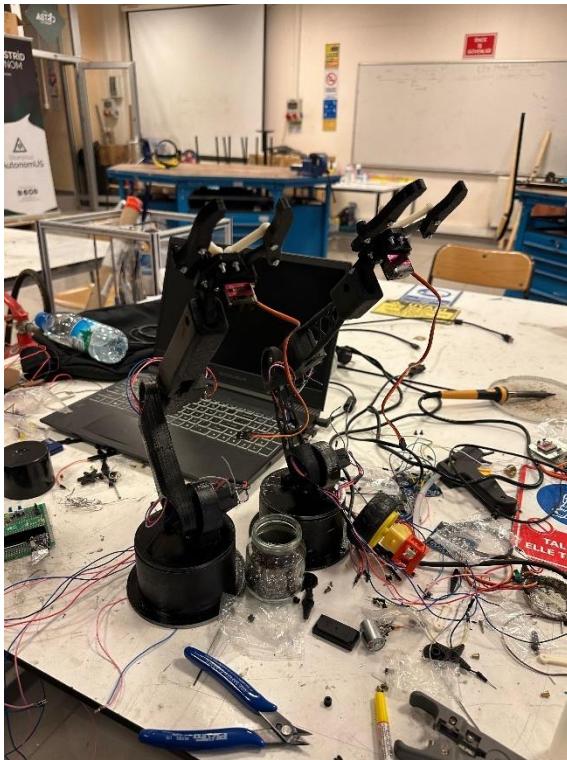


Figure 4.55 The Arms & Wiring Process

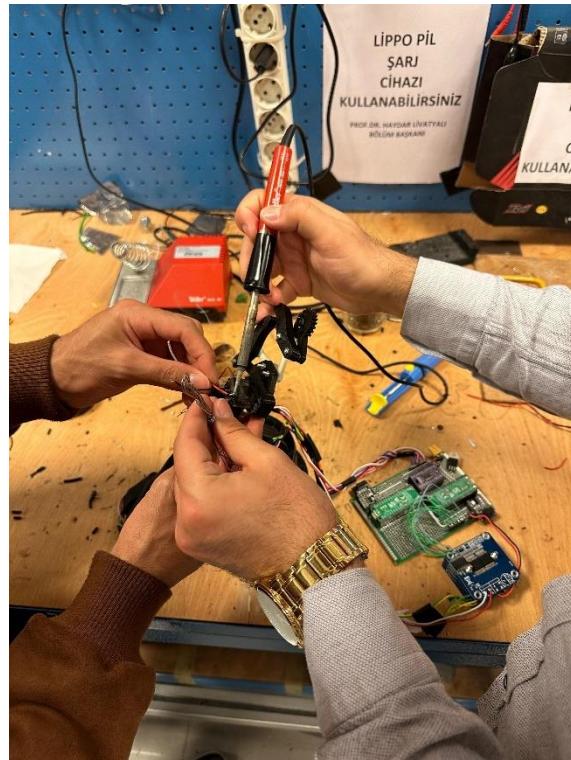
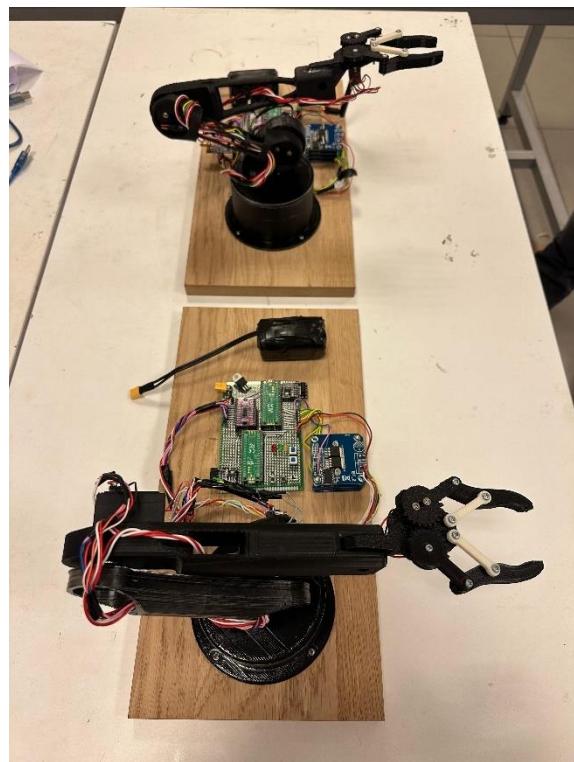
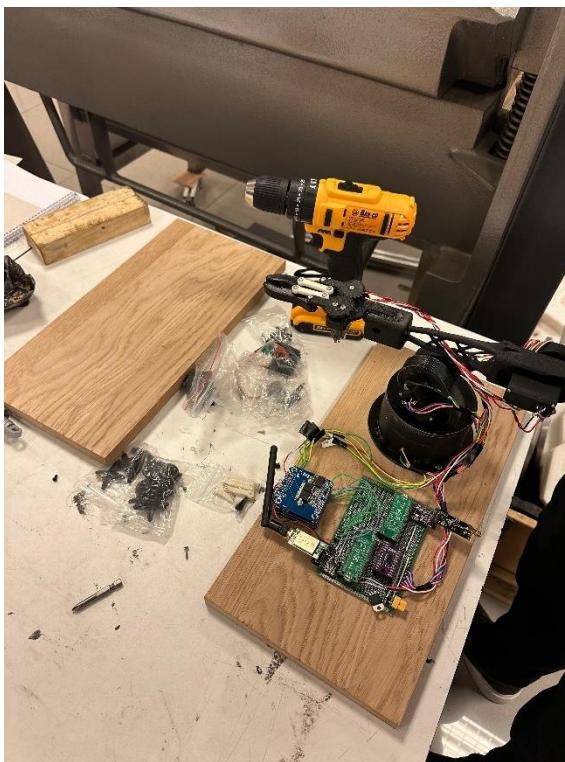
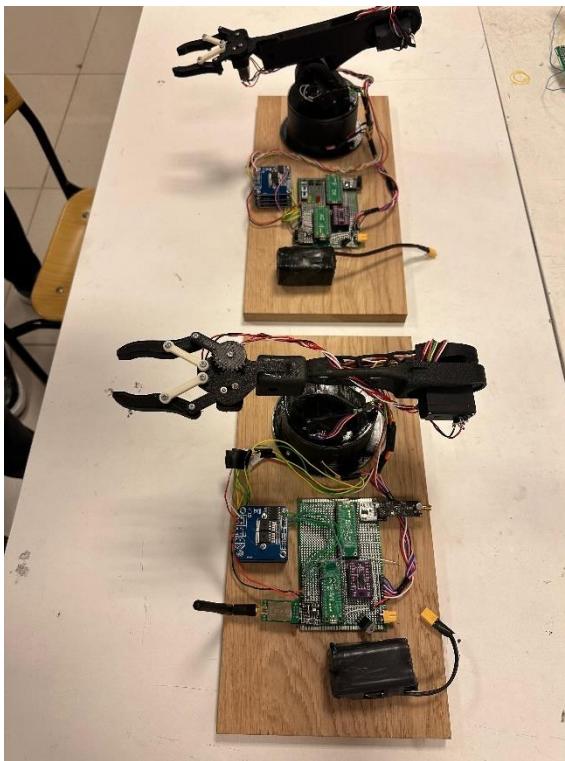


Figure 4.56 Soldering Process



**Figure 4.57 Mounting on Wooden Board    Figure 4.58 Arrangement of the Arms**



**Figure 4.58 Final Position of the Arms**

**Figure 4.59 Final Position of the Arms**

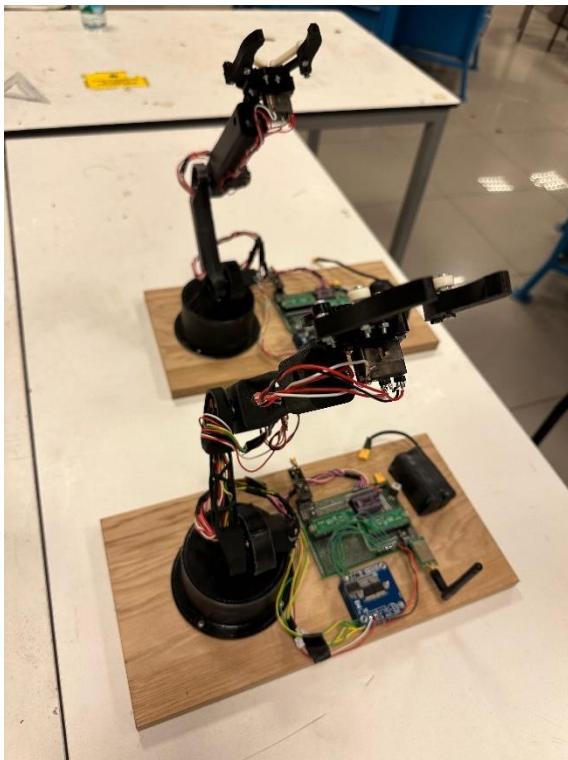


Figure 4.59 Final Position of the Arms



Figure 4.60 Example of Lifting an Object

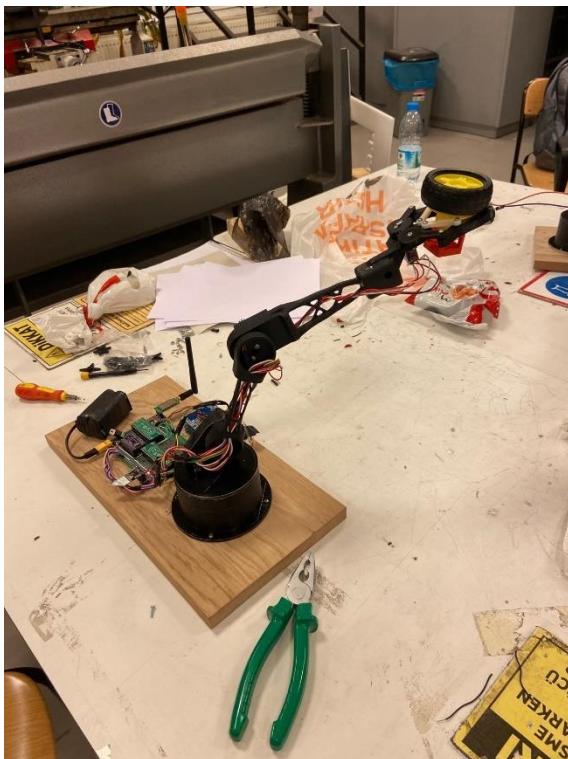


Figure 4.61 Example of Lifting an Object

## 4.2 Electronical Design

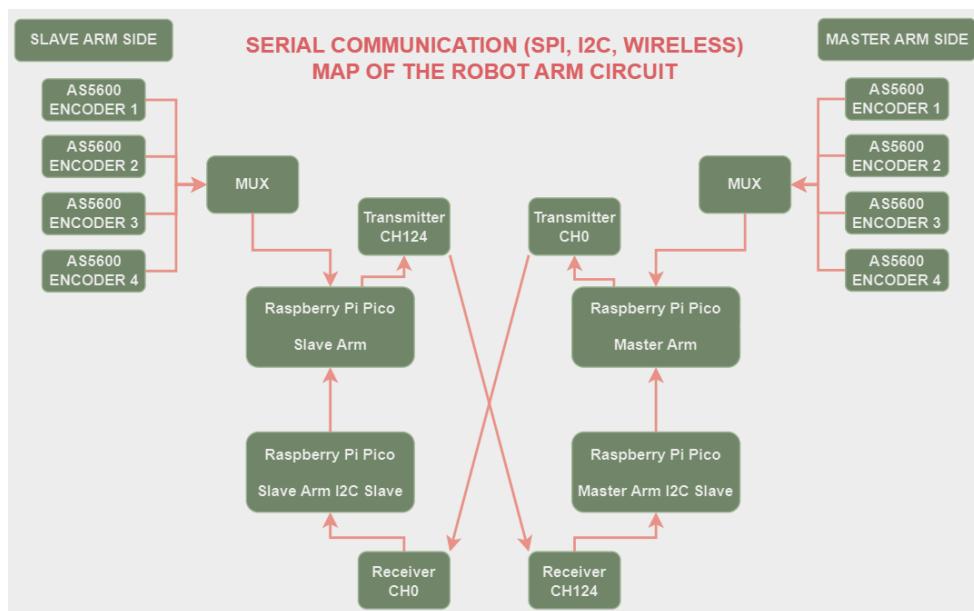
### 4.2.1 Component Selection

Below the components and the descriptions for the robot arms are listed.

**Table 4-1 Component List**

COMPONENT	DESCRIPTION	BASE QTY.
Battery	Orion 18650 3.7V 3200mah 3c Rechargeable Li-ion	8
Servo Motor	6 pcs. MG995R 13kg & 2 pcs. SG90 9G Mini	8
Magnetic Encoder	AS5600 (12-bit resolution)	8
Motor Driver	BTS7960 40A	8
Voltage Regulator	LM7805 3.3V	2
Microcontroller	2 pcs. Raspberry Pi Pico & 2 pcs. Raspberry Pico W	4
Wireless Communication Module	NRF24L01+PA+LNA SMA Antenna 2.4GHz 5km	4
Multiplexer	I2C TCA9548A	2
Mechanical Parts	Will be built-in 3D Printer (PLA Filament)(QTY in kg)	2
LED & Button	3 pcs. LED & 6 pcs. button	9
Wiring	For the connections (QTY in m)	10

### 4.2.2 Serial Communication (SPI, I2C, WIRELESS) Map of the Robot Arm Circuit



**Figure 4.62 Serial Communication Map**

#### 4.2.3 Simulation/Electronical Design

A circuit diagram was created in Proteus to establish connections between electronic components and modules. On the left side in the figure below, there is the circuit diagram of the controller arm (master arm). On the right side in the figure below, the circuit diagram of the controlled arm (slave arm) is present. The connections between these two arms are established wirelessly using nrf24l01 transceiver modules. In the master arm's circuit diagram, unlike the slave arm, there are buttons, potentiometers, and mode indicator LEDs for various mode options selections.

#### Master Arm Electronic Circuit

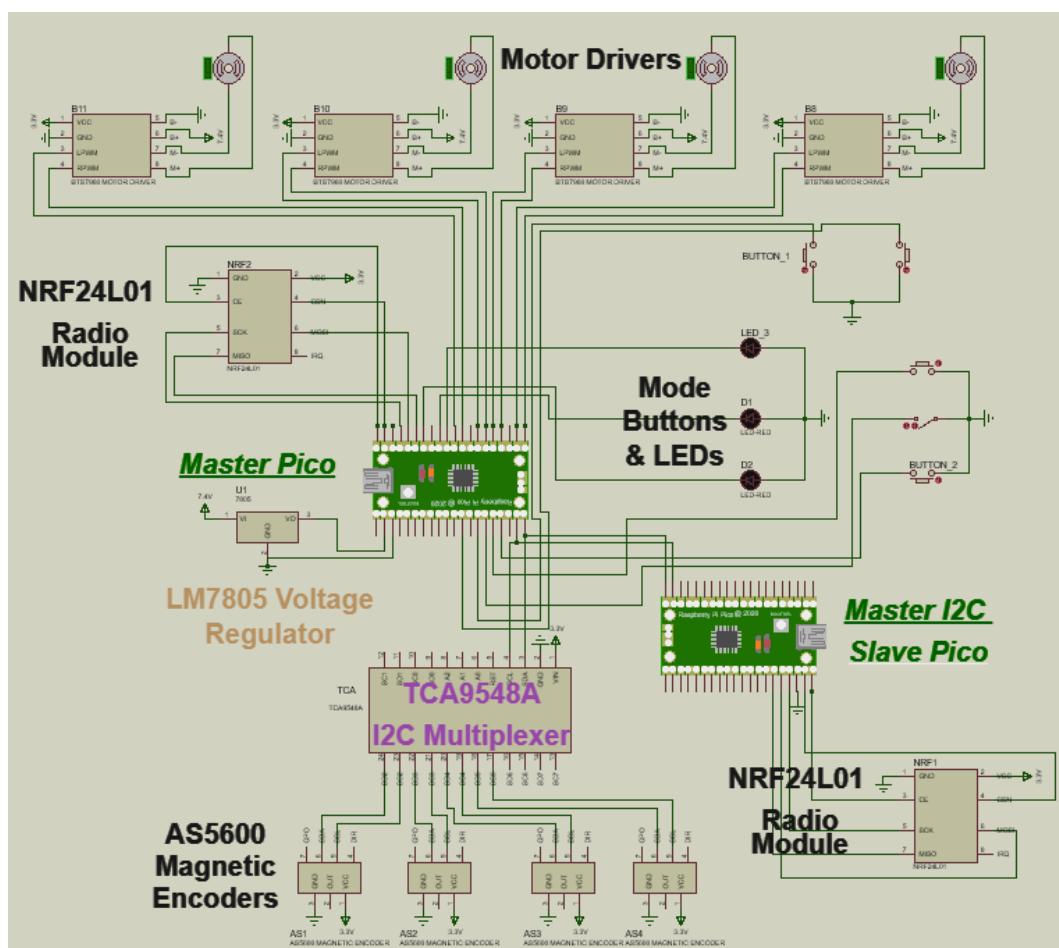


Figure 4.63 Controller Arm (Master Arm) Circuit Design on Proteus

## Slave Arm Electronic Circuit

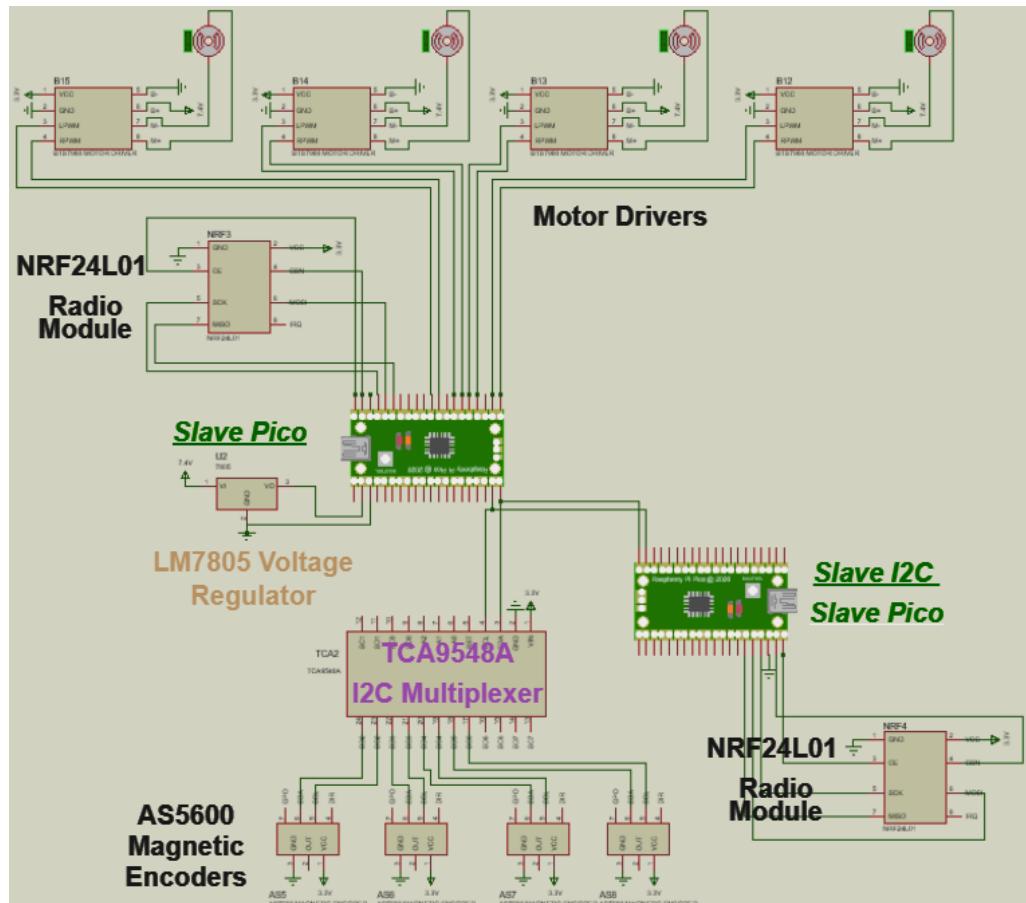


Figure 4.64 Controlled Arm (Slave Arm) Circuit Design on Proteus

#### 4.2.4 Electronical Realization

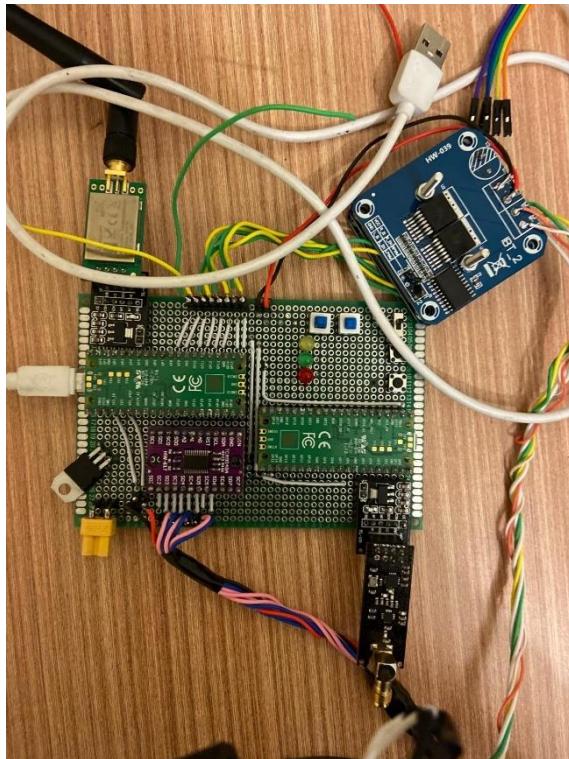


Figure 4.65 The Layout of The Circuit

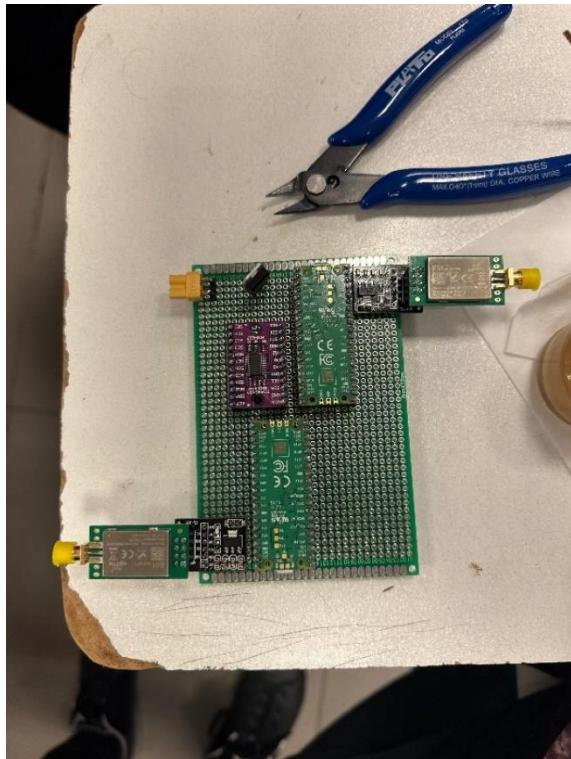


Figure 4.66 The Layout of The Circuit

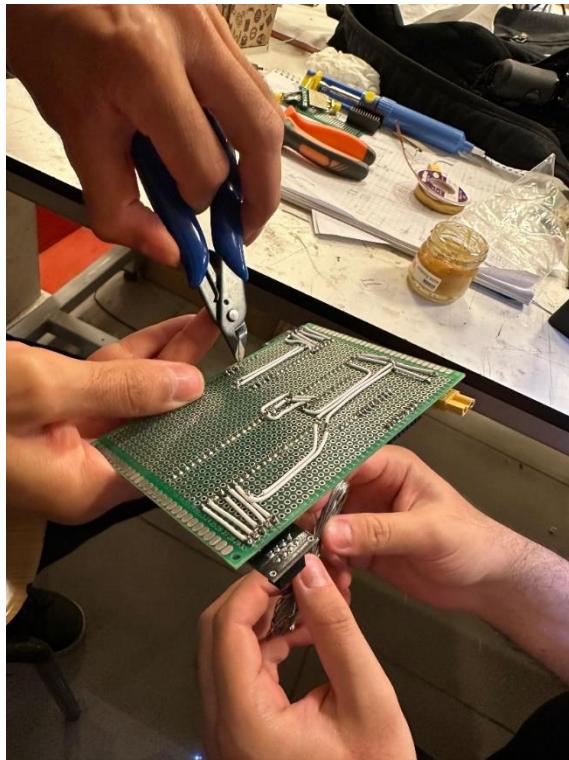


Figure 4.67 Wiring Process

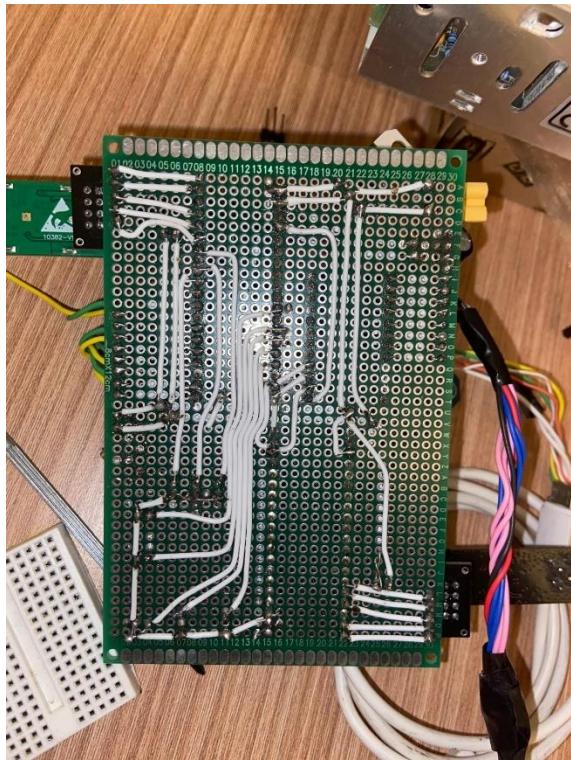
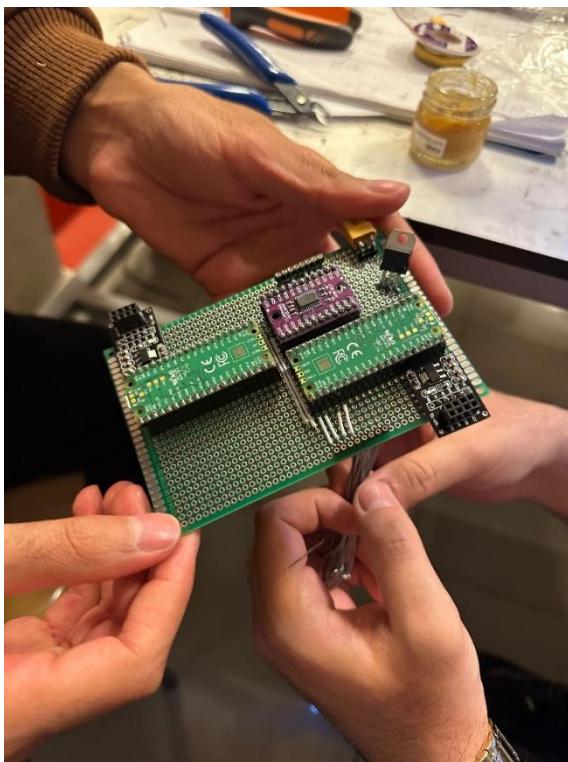
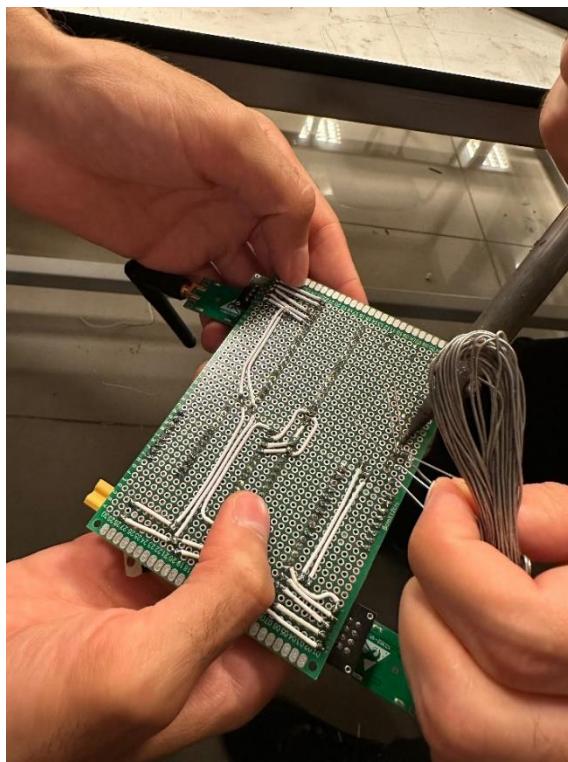


Figure 4.68 The Layout of The Wires



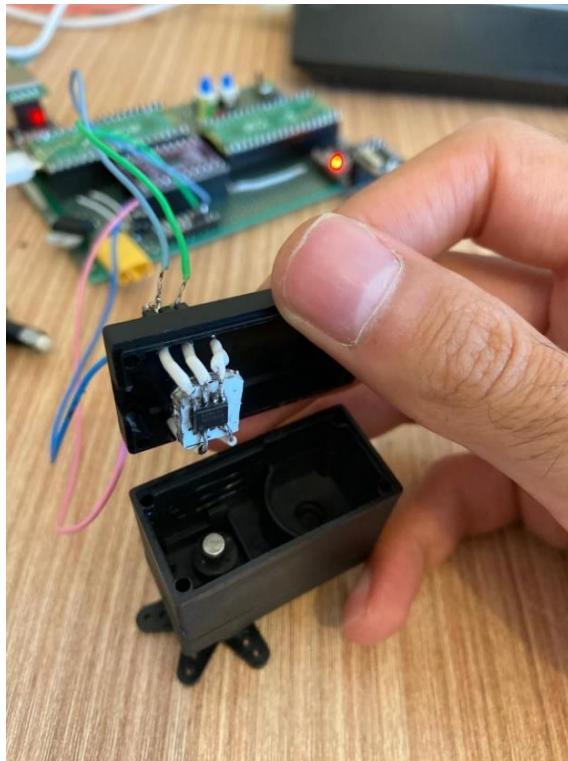
**Figure 4.69** The Layout of The Wires



**Figure 4.70** Soldering from Underside



**Figure 4.71** Embedding Encoder in Servo



**Figure 4.72** Embedding Encoder in Servo

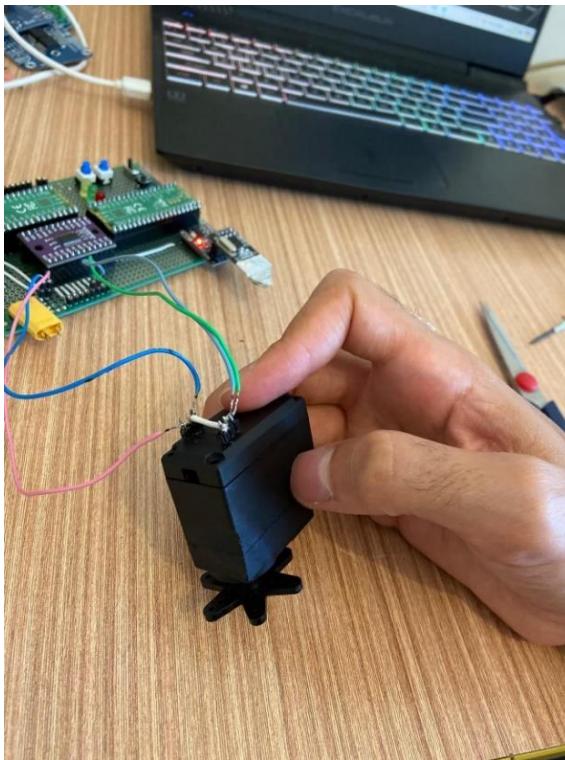


Figure 4.73 Embedded-Encoder Servo

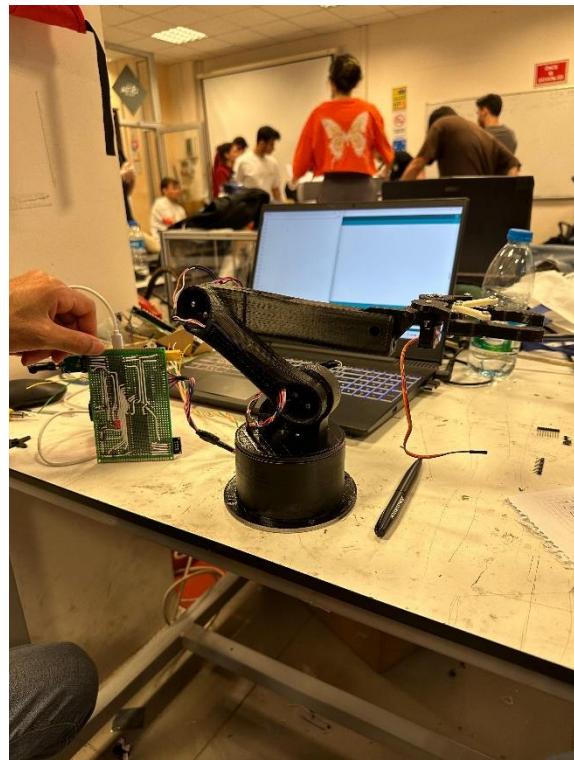


Figure 4.74 Wiring Process of The Arms

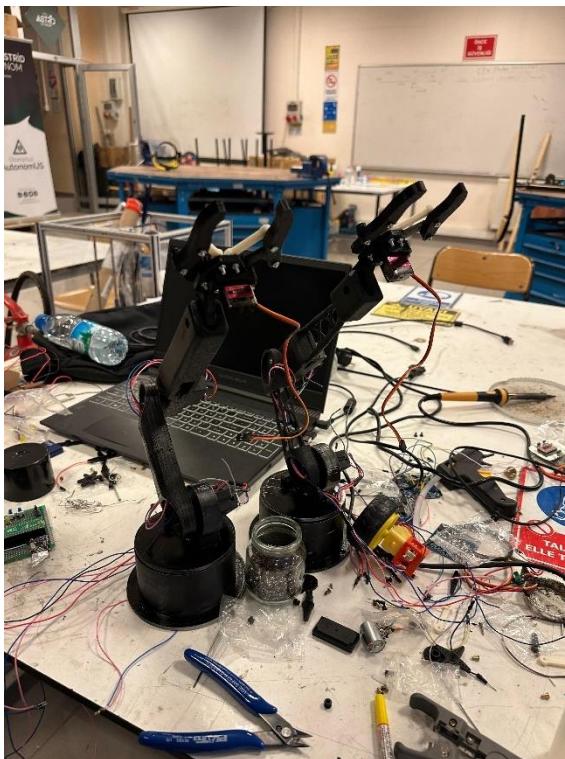


Figure 4.75 Master & Slave Arms

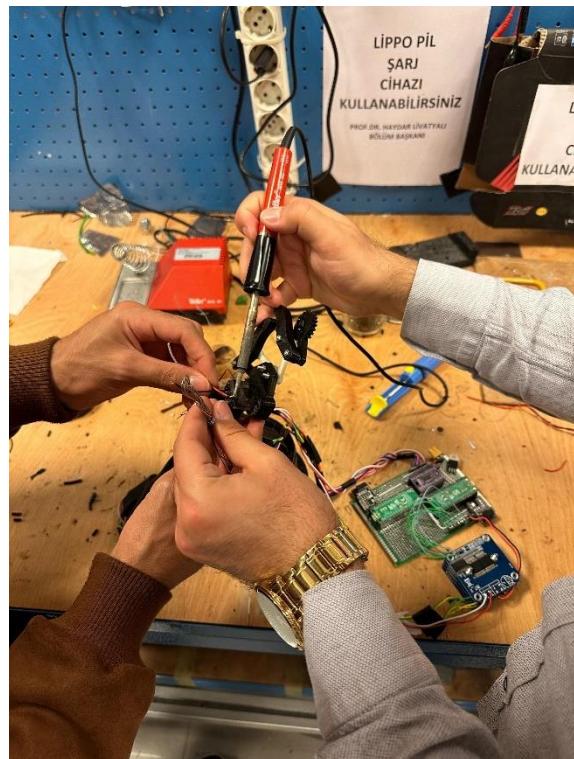


Figure 4.76 Soldering Process

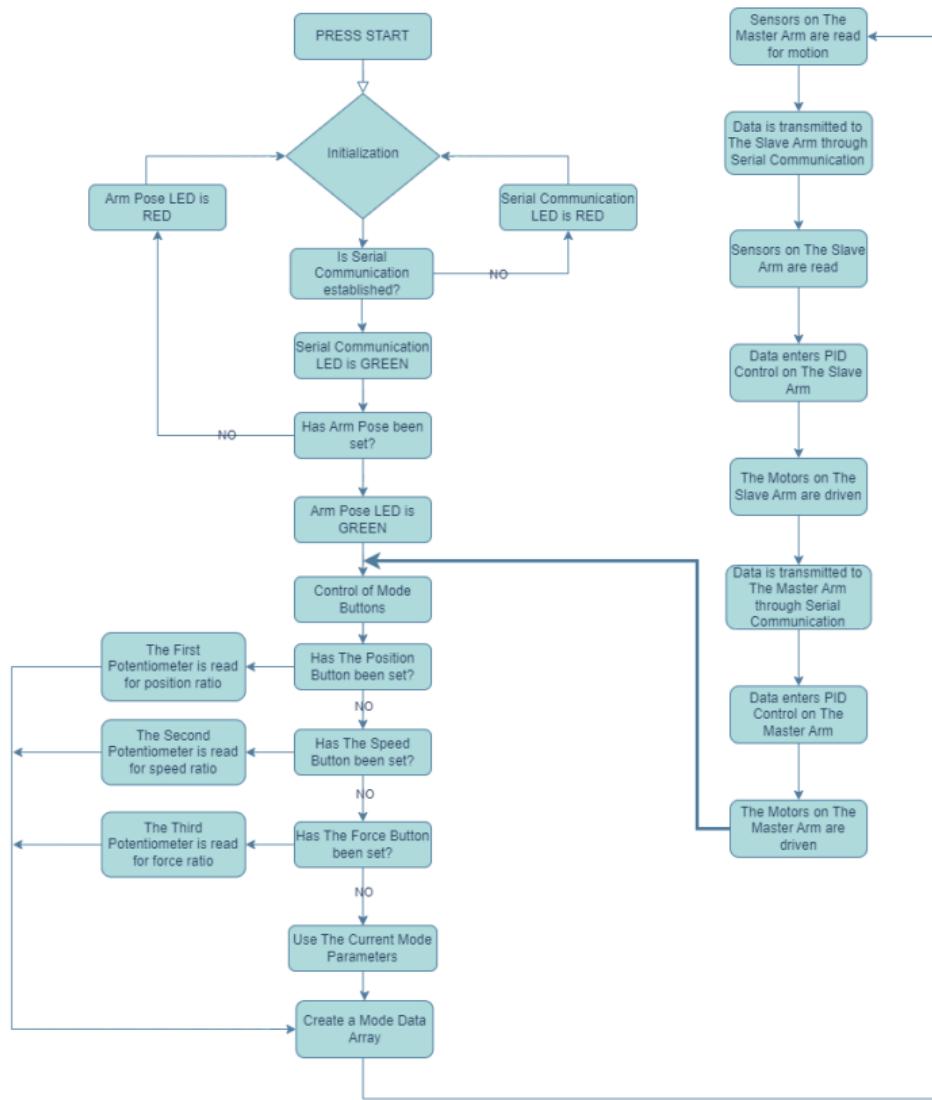


**Figure 4.77 Master Arm & its Circuit**

## 4.3 Software Algorithm

### 4.3.1 Algorithm Flow Chart

After the user presses the Start button, the initialization procedure will be initiated immediately. This procedure is necessary to ensure the robot arms can safely move to the desired initial position. The completion status of this procedure is under control. Upon successful completion, the system proceeds to the next step, data communication, and status information is conveyed to the user through LEDs as notifications. Subsequently, data for controlling the arm, which will be operated with physical controls from the operator, is transferred. The system is detailed in the diagram below.

**Figure 4.78 Flow Chart Algorithm**

### 4.3.2 Codes

In the electronic design, as illustrated in the serial communication map, we have prepared four separate code bases, each dedicated to one of the four microcontrollers involved. Hence, we have developed four distinct codes: Master Arm, Master Arm I2C Slave, Slave Arm, and Slave Arm I2C Slave. These codes are organized under their respective headings. For instance, in the code written for the Master Pico, communication occurs via I2C with the Master Arm I2C Slave microcontroller, which, in turn, communicates with the NRF24L01 to receive data from the remote source. This received data, representing the setpoint from the slave arm's encoders, is defined as the setpoint for the master arm. Additionally, data obtained from the encoders on the master arm is utilized as feedback and transmitted to the Slave Pico as setpoints via the NRF24L01 transmitter module connected to the master. These setpoint and feedback data are employed in PID control to manipulate the motors on the arms as desired, ensuring control is achieved. The code also includes necessary libraries for electronic modules. Furthermore, coefficients for force transmission ratios and functions for these modes are incorporated. Angular motion limits are defined for each joint to prevent undesirable positions. Comments are provided in the code for clarity, and optimizations are made to ensure it operates as swiftly as possible. Additionally, the same scenario applies to the Slave Arm as well.

#### Master Arm Code

```
/*
All content and software components contained in this document are Copyright ©
[2024] [Robotic Arm Control With Haptic Feedback] Project,
protected by Kadir Kadiroğlu, Özkan Yaşar, Oğuzhan Can.
Unauthorized copying, distribution, or use in any form is prohibited.
All rights reserved.
*/

#include <Wire.h>
#include <AS5600.h>
#include <SPI.h>
#include <RF24.h>

AS5600 as5600;
RF24 radiotransmitter (0,1);

// Radio Address for Master_Transmitter
const byte addr=240;
```

```
// Define Slave I2C Address
#define SLAVE_ADDR 9
uint8_t byteArray[8];

//Motor Driver Pins
const int motor1Pin1 = 14;
const int motor1Pin2 = 15;
const int motor2Pin1 = 12;
const int motor2Pin2 = 13;
const int motor3Pin1 = 10;
const int motor3Pin2 = 11;
const int motor4Pin1 = 8;
const int motor4Pin2 = 9;

//buttons
const int button_5 =22;
const int button_4 =21;
const int button_3 =20;
const int button_2 =19;
const int button_1 =18;

//leds
const int led_yellow =7;
const int led_green =6;
const int led_red =5;

void i2c_mux(uint8_t bus) {
    Wire.beginTransmission(0x70);
    Wire.write(1 << bus);
    Wire.endTransmission(); }

//pid parameters
int feedback_1;
int feedback_2;
int feedback_3;
int feedback_4;
int setpoint[4];
int rf_write[4];
int prerror=0;
int integral=0;
int prerror2=0;
int integral2=0;
int prerror3=0;
int integral3=0;
int prerror4=0;
int integral4=0;
```

```
//-----  
  
//Sensitivity coefficient  
byte force_coeff;  
unsigned long previousTime = 0;  
byte counter=0;  
byte counter2=0;  
  
bool lastButtonState = HIGH;  
bool lastButtonState_2 = HIGH;  
//-----Setups-----  
  
void setup() {  
  
    // pins setup-----  
    //motor driver pins  
    pinMode(motor1Pin1, OUTPUT);  
    pinMode(motor1Pin2, OUTPUT);  
    pinMode(motor2Pin1, OUTPUT);  
    pinMode(motor2Pin2, OUTPUT);  
    pinMode(motor3Pin1, OUTPUT);  
    pinMode(motor3Pin2, OUTPUT);  
    pinMode(motor4Pin1, OUTPUT);  
    pinMode(motor4Pin2, OUTPUT);  
  
    //buttons  
    pinMode(button_1, INPUT_PULLUP);  
    pinMode(button_2, INPUT_PULLUP);  
    pinMode(button_3, INPUT_PULLUP);  
    pinMode(button_4, INPUT_PULLUP);  
    pinMode(button_5, INPUT_PULLUP);  
  
    //leds  
    pinMode(led_yellow, OUTPUT);  
    pinMode(led_green, OUTPUT);  
    pinMode(led_red, OUTPUT);  
  
    //pwm setup-----  
  
    analogWriteFreq(22000);  
    analogWriteRange(1023);  
    analogWriteResolution(10);  
  
    //spi setup  
    SPI.setSCK(2);  
    SPI.setTX(3);  
    SPI.setRX(4);
```

```
//nrf module setup-----
radiotransmitter.begin();
radiotransmitter.setAutoAck(false);
radiotransmitter.openWritingPipe(addr);
radiotransmitter.setChannel(0);
radiotransmitter.setPALevel(RF24_PA_MAX);
radiotransmitter.setDataRate(RF24_2MBPS);
radiotransmitter.stopListening();

//i2c setup
Wire.setSDA(16);
Wire.setSCL(17);
Wire.begin();
Wire.setClock(1000000);

as5600.begin();
as5600.setDirection(AS5600_CLOCKWISE);

Serial.begin(2000000);

}

void loop() {
bool modes=digitalRead(button_3);
bool currentButtonState = digitalRead(button_1);

if (lastButtonState == HIGH && currentButtonState == LOW) {
  if(modes){if(counter<3){counter++;}}
  if(!modes){if(counter2<3){counter2++;}}
}

lastButtonState = currentButtonState;

bool currentButtonState_2 = digitalRead(button_2);

if (lastButtonState_2 == HIGH && currentButtonState_2 == LOW) {
  if(modes){if(counter>0){counter--;}}
  if(!modes){if(counter2>0){counter2--;}}
}

lastButtonState_2 = currentButtonState_2;

//Serial.print("Button pressed! Counter: ");
//Serial.println(counter);
//Serial.print("Button pressed! Counter2: ");
//Serial.println(counter2);
```

```
if(modes){  
    digitalWrite(led_red, counter == 3 ? HIGH : LOW);  
    digitalWrite(led_green, counter == 2 ? HIGH : LOW);  
    digitalWrite(led_yellow, counter == 1 ? HIGH : LOW);  
}  
  
else{digitalWrite(led_red, counter2 == 3 ? HIGH : LOW);  
digitalWrite(led_green, counter2 == 2 ? HIGH : LOW);  
digitalWrite(led_yellow, counter2 == 1 ? HIGH : LOW);}  
  
if(counter==0){  
    //unilateral mode  
    force_coeff=200;  
}  
if(counter==1){  
    force_coeff=2;  
}  
if(counter==2){  
    force_coeff=1.6;  
}  
if(counter==3){  
    force_coeff=1.3;  
}  
  
//AS5600 Readings via TCA9548A  
i2c_mux(3);  
feedback_1=as5600.readAngle()/4;  
i2c_mux(4);  
feedback_2=as5600.readAngle()/4;  
i2c_mux(5);  
feedback_3=as5600.readAngle()/4;  
i2c_mux(2);  
feedback_4=as5600.readAngle()/4;  
  
Serial.println(feedback_1);  
unsigned long currentTime = micros();  
unsigned long elapsedTime = currentTime - previousTime;  
// Serial.print("Elapsed Time (in microseconds): ");  
//Serial.println(elapsedTime);  
  
rf_write[0]=feedback_1;  
rf_write[1]=feedback_2;  
rf_write[2]=feedback_3;  
rf_write[3]=feedback_4;  
  
// Transmitting Data via Radio Module
```

```

radiotransmitter.write(&rf_write,sizeof(rf_write));
// Receiving Data via I2C From Slave Pico hat is connected to Master Receiver
Wire.requestFrom(SLAVE_ADDR,sizeof(byteArray));

if (Wire.available() ==sizeof(byteArray)) {
for(int i=0;i<sizeof(byteArray);i++){

byteArray[i]=Wire.read();

}

}

// Converting 8 bytes value Array to 4 integer array
for (int i = 0; i < 4; i++) {
setpoint[i] = (byteArray[i * 2] << 8) | (byteArray[i * 2 + 1]);
}

// Limits of the robot arms for each joint/link in order not to conflict the
other parts of the body
if(setpoint[0]<400){setpoint[0]=400;}
if(setpoint[0]>900){setpoint[0]=900;}

if(setpoint[1]<250){setpoint[1]=250;}
if(setpoint[1]>820){setpoint[1]=820;}

if(setpoint[2]<120){setpoint[2]=120;}
if(setpoint[2]>900){setpoint[2]=900;}

if(setpoint[3]<382){setpoint[3]=382;}
if(setpoint[3]>620){setpoint[3]=620;

Serial.print(',');
//Serial.print("slvdata1 ");
Serial.println(setpoint[0]);
previousTime = currentTime;

//PID 1-----
int error=(setpoint[0]-feedback_1);
//int error=(512-feedback_1); // In order to set the Joint in the middle
angle position

integral= (integral+error);

if (integral>=1023){integral=1023;}
if (integral<=-1023){integral=-1023;}

```

```

int deriv=error-preerror;
preerror=error;
int pwm =8*error+0.1*integral+120*deriv;

if (error<=1 && error>=-1){integral=0;pwm=0; }
if (pwm>=1023){pwm=1023;}
if (pwm<=-1023){pwm=-1023;}

// Motor control part-----
if (pwm >=0) {
    analogWrite(motor1Pin1, pwm/force_coeff);
    analogWrite(motor1Pin2, 0);
}

if(pwm<0)
{
    analogWrite(motor1Pin1, 0);
    analogWrite(motor1Pin2, -pwm/force_coeff);
}
//PWM&MOTOR1 END

//PID 2-----
int error2=(setpoint[1]-feedback_2);
//int error2=(512-feedback_2);
integral2= (integral2+error2);

if (integral2>=1023){integral2=1023;}
if (integral2<=-1023){integral2=-1023;}
int deriv2=error2-preerror2;
preerror2=error2;
int pwm2 =8*error2+0.1*integral2+120*deriv2;
if (error2<=1 && error2>=-1){integral2=0;pwm2=0; }
if (pwm2>=1023){pwm2=1023;}
if (pwm2<=-1023){pwm2=-1023;}

// Motor control part-----
if (pwm2 >=0) {
    analogWrite(motor2Pin2, pwm2/force_coeff);
    analogWrite(motor2Pin1, 0);
}

if(pwm2<0)
{
    analogWrite(motor2Pin2, 0);
    analogWrite(motor2Pin1, -pwm2/force_coeff);
}
//PWM&MOTOR2 END

```

```

//PID 3-----
int error3=(setpoint[2]-feedback_3);
//int error3=(512-feedback_3);
integral3= (integral3+error3);

if (integral3>=1023){integral3=1023;}
if (integral3<=-1023){integral3=-1023;}

int deriv3=error3-prerror3;
prerror3=error3;
int pwm3 =8*error3+0.1*integral3+120*deriv3;

if (error3<=1 && error3>=-1){integral3=0;pwm3=0; }
if (pwm3>=1023){pwm3=1023; }
if (pwm3<=-1023){pwm3=-1023; }

// Motor control part-----
if (pwm3 >=0) {
    analogWrite(motor3Pin1, pwm3/force_coeff);
    analogWrite(motor3Pin2, 0);
}

if(pwm3<0)
{
    analogWrite(motor3Pin1, 0);
    analogWrite(motor3Pin2, -pwm3/force_coeff);
}
//PWM&MOTOR3 END

//PID 4-----
int error4=(setpoint[3]-feedback_4);
//int error4=(512-feedback_4);
integral4= (integral4+error4);

if (integral4>=1023){integral4=1023;}
if (integral4<=-1023){integral4=-1023;}
int deriv4=error4-prerror4;
prerror4=error4;
int pwm4 =9*error4+0.1*integral4+120*deriv4;

if (error4<=1 && error4>=-1){integral4=0;pwm4=0; }
if (pwm4>=1023){pwm4=1023; }
if (pwm4<=-1023){pwm4=-1023; }

```

```

// Motor control part-----
if (pwm4 >=0) {
    analogWrite(motor4Pin1, pwm4);
    analogWrite(motor4Pin2, 0);
}

if(pwm4<0)
{

    analogWrite(motor4Pin1, 0);
    analogWrite(motor4Pin2, -pwm4);

}
//PWM&MOTOR4 END

}

```

### Master Arm I2C Slave Code

```

/*
All content and software components contained in this document are Copyright ©
[2024] [Robotic Arm Control With Haptic Feedback] Project,
protected by Kadir Kadiroğlu, Özkan Yaşar, Oğuzhan Can.
Unauthorized copying, distribution, or use in any form is prohibited.
All rights reserved.
*/

#include <Wire.h>
#include <SPI.h>
#include <RF24.h>
RF24 radioreceiver (0,1);
const byte addr=20;

// Define Slave I2C Address
#define i2c_slave_addr 9
uint8_t byteArray[8];

// data array to get data from rf module wirelessly
int rf_read[4];

void setup() {
//spi setup
SPI.setSCK(2);

```

```
SPI.setTX(3);
SPI.setRX(4);

//nrf module setup as receiver-----
radioreceiver.begin();
radioreceiver.setAutoAck(false);
radioreceiver.openReadingPipe(1, addr);
radioreceiver.setChannel(124);
radioreceiver.setPALevel(RF24_PA_MAX);
radioreceiver.setDataRate(RF24_2MBPS);
radioreceiver.startListening();

//i2c setup
Wire.setSDA(16);
Wire.setSCL(17);
// Initialize I2C communications as Slave
Wire.begin(i2c_slave_addr);
Wire.setClock(1000000);
// Function to run when data requested from master
Wire.onRequest(requestEvent);
}

void requestEvent() {
Wire.write(byteArray, sizeof(byteArray));
}

void loop() {
if (radioreceiver.available()) {

while(radioreceiver.available()){
radioreceiver.read(&rf_read,sizeof(rf_read));
}

}

//Serial.println(rf_read[0]);
// Convert integer array to byte array

for (int i = 0; i < 4; i++) {
byteArray[i * 2] = rf_read[i] >> 8; // Extract high byte (shift right by 8)
byteArray[i * 2 + 1] = rf_read[i] & 0xFF; // Extract low byte (mask with 0xFF)
}
}
```

## Slave Arm Code

```
/*
All content and software components contained in this document are Copyright ©
[2024] [Robotic Arm Control With Haptic Feedback] Project,
protected by Kadir Kadiroğlu, Özkan Yaşar, Oğuzhan Can.
Unauthorized copying, distribution, or use in any form is prohibited.
All rights reserved.
*/

#include <Wire.h>
#include <AS5600.h>
#include <SPI.h>
#include <RF24.h>

AS5600 as5600;
RF24 radiotransmitter (0,1);

// Define rf module address
const byte addr=20;

// Define Slave I2C Address
#define SLAVE_ADDR 9
uint8_t byteArray[8];

//Motor Driver Pins
const int motor1Pin1 = 14;
const int motor1Pin2 = 15;
const int motor2Pin1 = 12;
const int motor2Pin2 = 13;
const int motor3Pin1 = 10;
const int motor3Pin2 = 11;
const int motor4Pin1 = 8;
const int motor4Pin2 = 9;

//Motor Driver Pins
void i2c_mux(uint8_t bus) {
Wire.beginTransmission(0x70);
Wire.write(1 << bus);
Wire.endTransmission();    }

//pid parameters
int feedback_1;
int feedback_2;
int feedback_3;
```

```
int feedback_4;

int setpoint[4];
int rf_write[4];

int preerror=0;
int integral=0;
int preerror2=0;
int integral2=0;
int preerror3=0;
int integral3=0;
int preerror4=0;
int integral4=0;

//-----Setups-----

void setup() {
    // pins setup-----
    pinMode(25, OUTPUT);
    pinMode(motor1Pin1, OUTPUT);
    pinMode(motor1Pin2, OUTPUT);
    pinMode(motor2Pin1, OUTPUT);
    pinMode(motor2Pin2, OUTPUT);
    pinMode(motor3Pin1, OUTPUT);
    pinMode(motor3Pin2, OUTPUT);
    pinMode(motor4Pin1, OUTPUT);
    pinMode(motor4Pin2, OUTPUT);

    //pwm setup-----
    analogWriteFreq(22000);
    analogWriteRange(1023);
    analogWriteResolution(10);

    //spi setup
    SPI.setSCK(2);
    SPI.setTX(3);
    SPI.setRX(4);

    //nrf module setup-----
    radiotransmitter.begin();
    radiotransmitter.setAutoAck(false);
    radiotransmitter.openWritingPipe(addr);
    radiotransmitter.setChannel(124);
    radiotransmitter.setPALevel(RF24_PA_MAX);
    radiotransmitter.setDataRate(RF24_2MBPS);
    radiotransmitter.stopListening();
```

```
//i2c setup
Wire.setSDA(16);
Wire.setSCL(17);
Wire.begin();

//Wire.setClock(400000);
Wire.setClock(1000000);
as5600.begin();
as5600.setDirection(AS5600_CLOCK_WISE);
Serial.begin(2000000);

}

void loop() {
Wire.requestFrom(SLAVE_ADDR,sizeof(byteArray));

if (Wire.available() ==sizeof(byteArray)) {
for(int i=0;i<sizeof(byteArray);i++){

byteArray[i]=Wire.read();
/*
setpoint[i]=Wire.read() << 8;
setpoint[i]|=Wire.read();
*/
}
}

for (int i = 0; i < 4; i++) {
setpoint[i] = (byteArray[i * 2] << 8) | (byteArray[i * 2 + 1]);
}

if(setpoint[0]<400){setpoint[0]=400;}
if(setpoint[0]>900){setpoint[0]=900;}

if(setpoint[1]<250){setpoint[1]=250;}
if(setpoint[1]>820){setpoint[1]=820;}

if(setpoint[2]<120){setpoint[2]=120;}
if(setpoint[2]>900){setpoint[2]=900;}

if(setpoint[3]<430){setpoint[3]=430;}
if(setpoint[3]>640){setpoint[3]=640;

i2c_mux(3);
feedback_1=as5600.readAngle()/4;
i2c_mux(4);
```

```
feedback_2=as5600.readAngle()/4;
i2c_mux(5);
feedback_3=as5600.readAngle()/4;
i2c_mux(2);
feedback_4=as5600.readAngle()/4;

Serial.println(feedback_1);

rf_write[0]=feedback_1;
rf_write[1]=feedback_2;
rf_write[2]=feedback_3;
rf_write[3]=feedback_4;

radiotransmitter.write(&rf_write,sizeof(rf_write));

Serial.print(',');
Serial.println(setpoint[0]);

//PID1-----
int error=(setpoint[0]-feedback_1);
//int error=(512-feedback_1);

integral= (integral+error);

if (integral>=1023){integral=1023;}
if (integral<=-1023){integral=-1023;}

int deriv=error-preerror;
preerror=error;
int pwm =9*error+0.10*integral+120*deriv;

if (error<=1 && error>=-1){integral=0;pwm=0; }
if (pwm>=1023){pwm=1023;}
if (pwm<=-1023){pwm=-1023; }

// Motor control part-----
if (pwm >=0) {
    analogWrite(motor1Pin1, pwm);
    analogWrite(motor1Pin2, 0);
}

if(pwm<0)
{
analogWrite(motor1Pin1, 0);
analogWrite(motor1Pin2, -pwm);
}
```

```

//PWM&MOTOR END

//PID2-----
int error2=(setpoint[1]-feedback_2);
//int error2=(512-feedback_2);
integral2= (integral2+error2);
if (integral2>=1023){integral2=1023;}
if (integral2<=-1023){integral2=-1023;}

int deriv2=error2-preerror2;
preerror2=error2;
int pwm2 =9*error2+0.10*integral2+120*deriv2;

if (error2<=1 && error2>=-1){integral2=0;pwm2=0; }
if (pwm2>=1023){pwm2=1023;}
if (pwm2<=-1023){pwm2=-1023;}

// Motor control part-----
if (pwm2 >=0) {
    analogWrite(motor2Pin1, pwm2);
    analogWrite(motor2Pin2, 0);
}

if(pwm2<0)
{
    analogWrite(motor2Pin1, 0);
    analogWrite(motor2Pin2, -pwm2);
}
//PWM&MOTOR2 END

//PID3-----
int error3=(setpoint[2]-feedback_3);
//int error3=(512-feedback_3);

integral3= (integral3+error3);
if (integral3>=1023){integral3=1023;}
if (integral3<=-1023){integral3=-1023;}
int deriv3=error3-preerror3;
preerror3=error3;
int pwm3 =9*error3+0.10*integral3+120*deriv3;

if (error3<=1 && error3>=-1){integral3=0;pwm3=0; }
if (pwm3>=1023){pwm3=1023;}
if (pwm3<=-1023){pwm3=-1023;}

// Motor control part-----
if (pwm3 >=0) {

```

```
analogWrite(motor3Pin1, pwm3);
analogWrite(motor3Pin2, 0);
}

if(pwm3<0)
{
    analogWrite(motor3Pin1, 0);
    analogWrite(motor3Pin2, -pwm3);
}
//PWM&MOTOR3 END

//PID4-----
int error4=(setpoint[3]-feedback_4);
//int error4=(512-feedback_4);
integral4= (integral4+error4);

if (integral4>=1023){integral4=1023;}
if (integral4<=-1023){integral4=-1023;}

int deriv4=error4-prerror4;
prerror4=error4;
int pwm4 =9*error4+0.10*integral4+120*deriv4;

if (error4<=1 && error4>=-1){integral4=0;pwm4=0; }
if (pwm4>=1023){pwm4=1023;}
if (pwm4<=-1023){pwm4=-1023; }

// Motor control part-----

if (pwm4 >=0) {
    analogWrite(motor4Pin1, pwm4);
    analogWrite(motor4Pin2, 0);
}

if(pwm4<0)
{
    analogWrite(motor4Pin1, 0);
    analogWrite(motor4Pin2, -pwm4);
}
//PWM&MOTOR4 END

}
```

## Slave Arm I2C Slave Code

```
/*
All content and software components contained in this document are Copyright ©
[2024] [Robotic Arm Control With Haptic Feedback] Project,
protected by Kadir Kadiroğlu, Özkan Yaşar, Oğuzhan Can.
Unauthorized copying, distribution, or use in any form is prohibited.
All rights reserved.
*/

#include <Wire.h>
#include <SPI.h>
#include <RF24.h>

RF24 radioreceiver (0,1);
const byte addr=240;

// Define Slave I2C Address
#define i2c_slave_addr 9
uint8_t byteArray[8];

// data array to get data from rf module wirelessly
int rf_read[4];
int anlg;
void setup() {

//spi setup
SPI.setSCK(2);
SPI.setTX(3);
SPI.setRX(4);

//nrf module setup as receiver-----
radioreceiver.begin();
radioreceiver.setAutoAck(false);
radioreceiver.openReadingPipe(1, addr);
radioreceiver.setChannel(0);
radioreceiver.setPALevel(RF24_PA_MAX);
radioreceiver.setDataRate(RF24_2MBPS);
radioreceiver.startListening();

//i2c setup
// Initialize I2C communications as Slave
Wire.setSDA(16);
Wire.setSCL(17);
Wire.begin(i2c_slave_addr);
Wire.setClock(1000000);
```

```
// Function to run when data requested from master
Wire.onRequest(requestEvent);

}

void requestEvent() {
Wire.write(byteArray, sizeof(byteArray));
}

void loop() {

if (radioreceiver.available()) {

while(radioreceiver.available()){

radioreceiver.read(&rf_read,sizeof(rf_read));

}

}

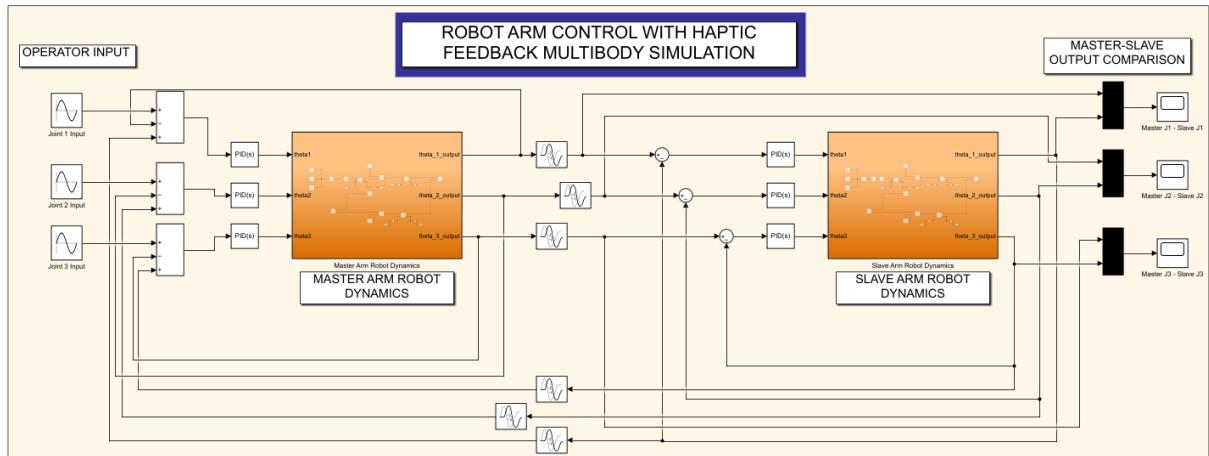
// Convert integer array to byte array
for (int i = 0; i < 4; i++) {
    byteArray[i * 2] = rf_read[i] >> 8; // Extract high byte (shift right by 8)
    byteArray[i * 2 + 1] = rf_read[i] & 0xFF; // Extract low byte (mask with
0xFF)
}

}
```

## 4.4 Control

### 4.4.1 Modelling

The haptic feedback control of robot arms is simulated by the Simulink circuit created using Simscape. In order to observe how the system will be affected by delays arising from communication and code processing times, time transpose blocks have been added to the circuit for each joint angle data. To simulate physical manipulations, sinusoidal signals have been input to the system. For effective control, PID blocks have been added for each joint. The coefficients in the PID blocks have been tuned to ensure the correct operation of the system. Since there are a total of three PID blocks, the loop tuning method has been used to automatically find the coefficients for each PID block. PID coefficients have been determined through iterations until reaching the desired tuning target, which is the peak gain coefficient. This has allowed the successful manipulation of motors in the joints while achieving the desired critical damping value. The PID blocks operate in saturation mode, ensuring that the constraints of the circuit are within the specified limits, providing a stable and secure output.



**Figure 4.79 Robot Arms Haptic Feedback Simscape Simulation**

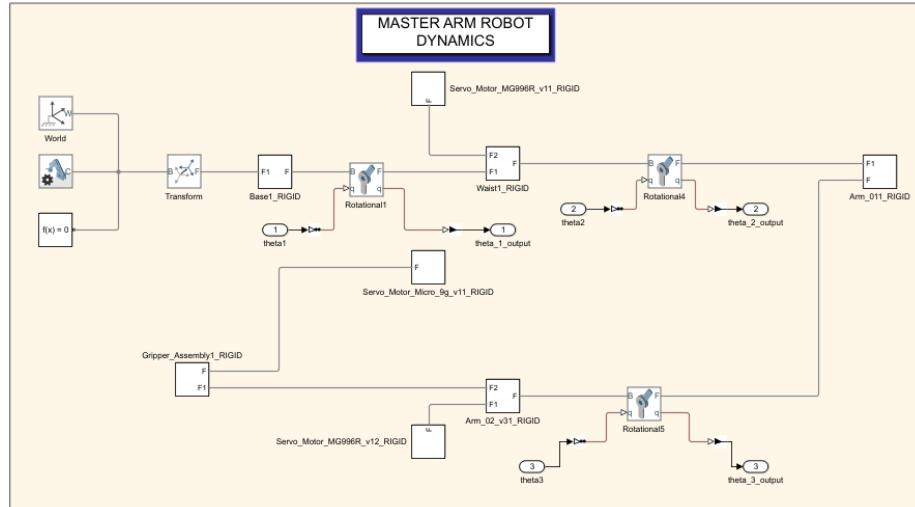


Figure 4.80 Master Arm Multibody Diagram

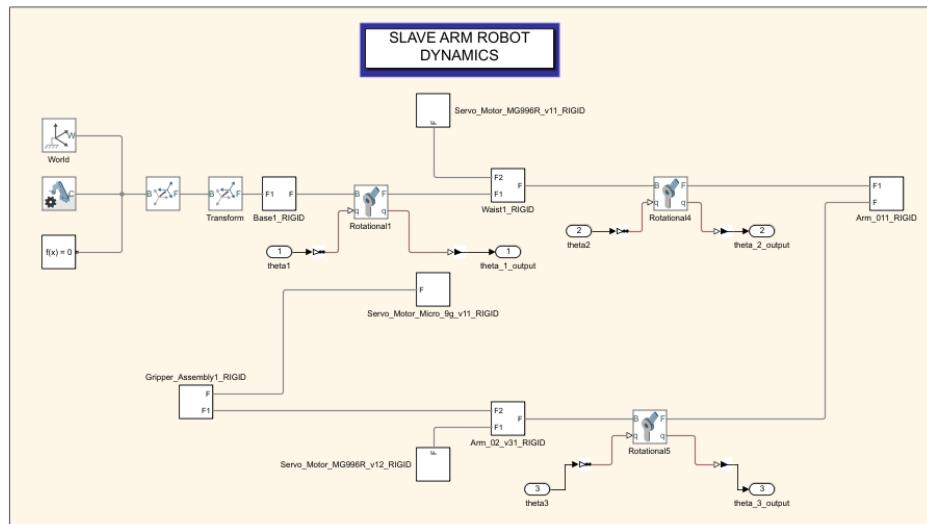
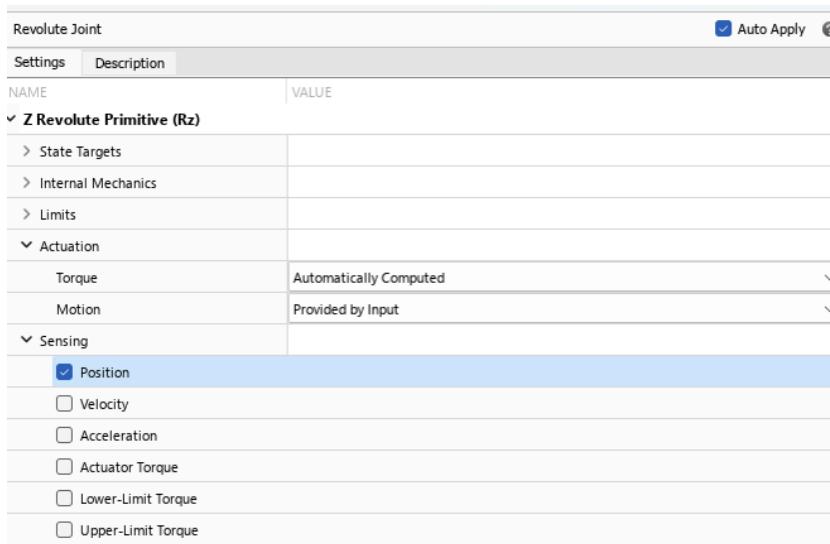


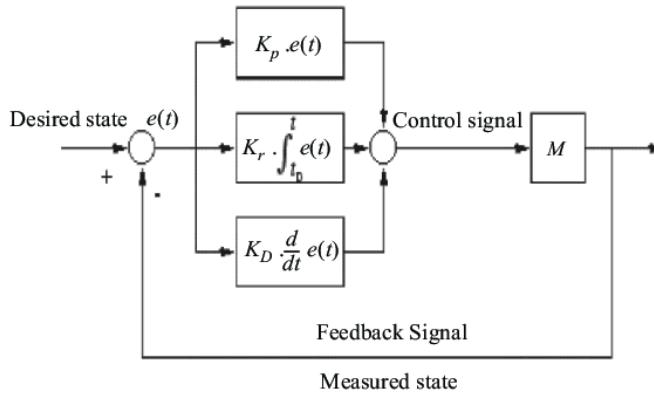
Figure 4.81 Slave Arm Multibody Diagram



**Figure 4.82 Joint Actuation & Sensing Types in Simscape**

#### 4.4.2 Control Method & Optimization

In our bilateral teleoperation robot arm project, the selection of PID control with the loop tuning method was motivated by several key factors. Firstly, PID controllers offer inherent stability and robustness, crucial for ensuring the system's reliability in teleoperation scenarios. The loop tuning method further provides a systematic means to fine-tune PID parameters, allowing adaptability to variations in the robot arm's dynamics. Additionally, PID control simplifies implementation while delivering precise and optimized performance in terms of minimizing overshoot and settling time. The real-time responsiveness of PID controllers aligns with the project's requirements for quick and accurate control. Furthermore, the use of PID with loop tuning adheres to industry standards, facilitating integration with existing technologies and ensuring compatibility with established control practices. Overall, this choice was driven by the need for a stable, adaptable, and industry-standard control strategy that meets the specific demands of our teleoperation project.

**Figure 4.83 PID Control Block Scheme**

```

Some closed-loop poles are marginally stable (decay rate near 1e-07)
Final: Peak gain = 1.76, Iterations = 154
Some closed-loop poles are marginally stable (decay rate near 1e-07)
Final: Peak gain = 1.76, Iterations = 93
Some closed-loop poles are marginally stable (decay rate near 1e-07)
Final: Peak gain = 1.76, Iterations = 90
Some closed-loop poles are marginally stable (decay rate near 1e-07)
Final: Peak gain = 1.76, Iterations = 79
Some closed-loop poles are marginally stable (decay rate near 1e-07)
Final: Peak gain = 5.78, Iterations = 101
Some closed-loop poles are marginally stable (decay rate near 1e-07)
Final: Peak gain = 1.76, Iterations = 79
Some closed-loop poles are marginally stable (decay rate near 1e-07)
Final: Peak gain = 1.63, Iterations = 107
Some closed-loop poles are marginally stable (decay rate near 1e-07)

```

**Figure 4.84 Loop Tuning Iteration Process**

#### 4.4.3 Control Method & Optimization Realization

The PID coefficients were optimized through iterations, utilizing trial and error, to achieve the desired response from the system without inducing rapid oscillations. The system was designed to deliver an underdamped response to facilitate a quick reaction. As a result, the following PID coefficients were obtained.

```

int deriv=error-preerror;
preerror=error;
int pwm =8*error+0.1*integral+120*deriv;

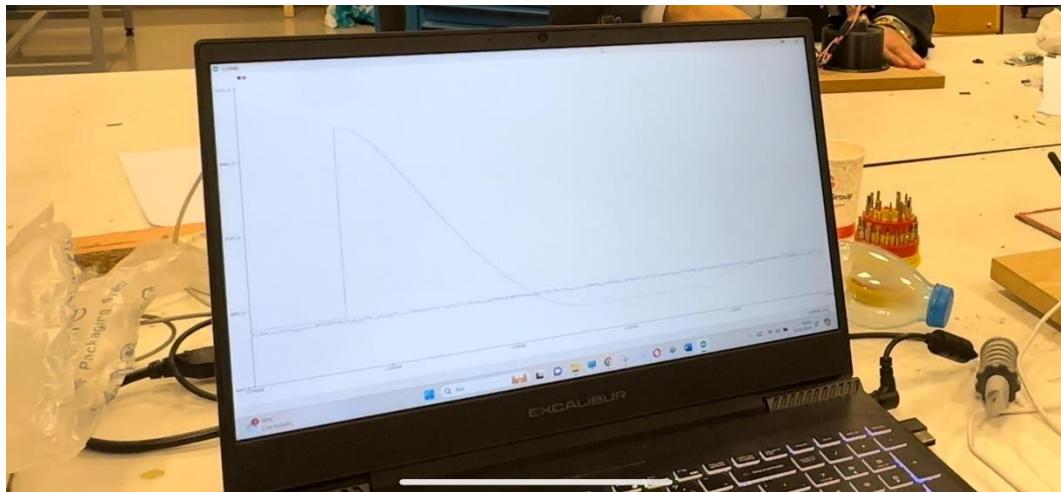
if (error<=1 && error>=-1){integral=0;pwm=0; }
if (pwm>=1023){pwm=1023; }
if (pwm<=-1023){pwm=-1023; }

```

**Figure 4.85 PID Coefficients Found by Optimization**

~~XXXXXXFigure: PID Coefficients Found by OptimizationXXXXXX~~

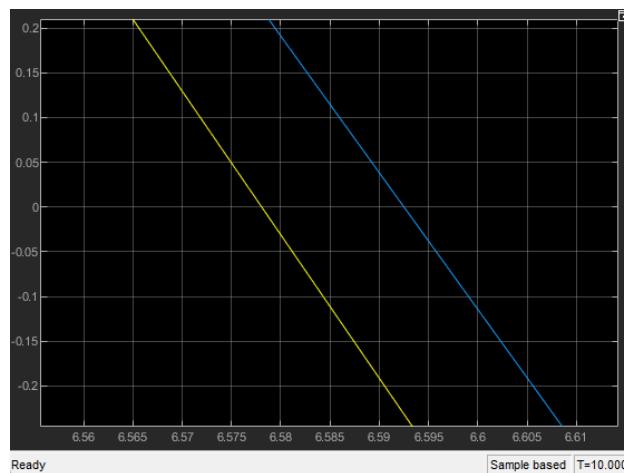
Below is the output graph obtained using the PID coefficients as seen in the code. The System response is underdamped response as it's seen.



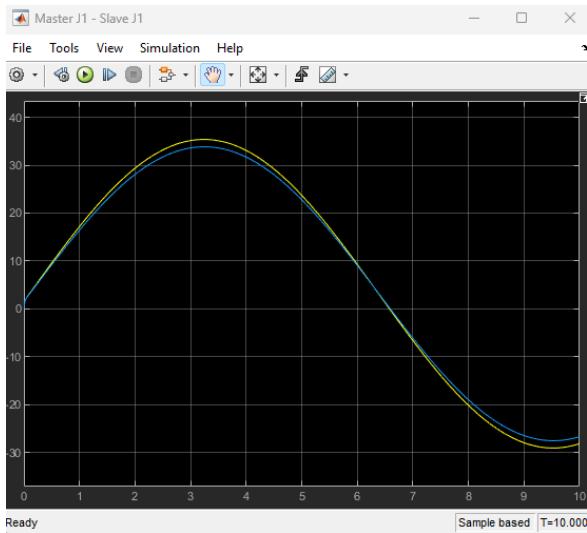
**Figure 4.86 Underdamped System Response**

#### 4.4.4 Simulink Simulation Output Graphs

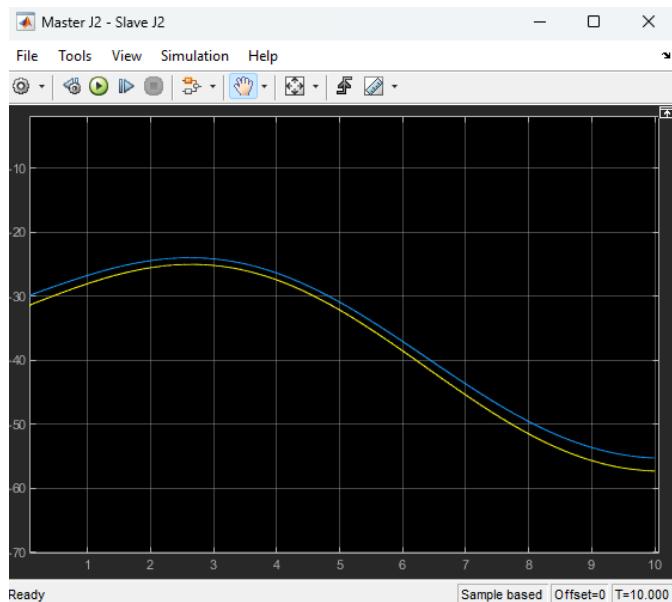
The simulation outputs of the circuit created in Simulink are shown in the graphs below. These graphs compare the positions of joints in the master and slave arms. As a result, the arms have successfully tracked each other to the desired extent.



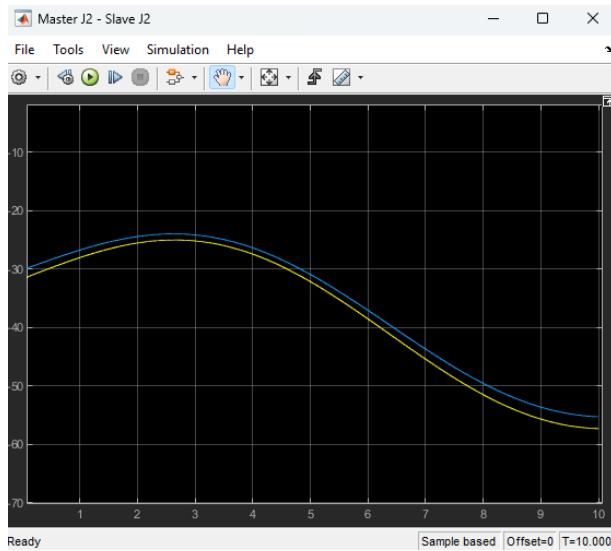
**Figure 4.87 Delay Between the Arm Motions Output**



**Figure 4.88 Master First Joint Angle – Slave First Joint Angle Comparison Output**



**Figure 4.89 Master Second Joint Angle – Slave Second Joint Angle Comparison Output**

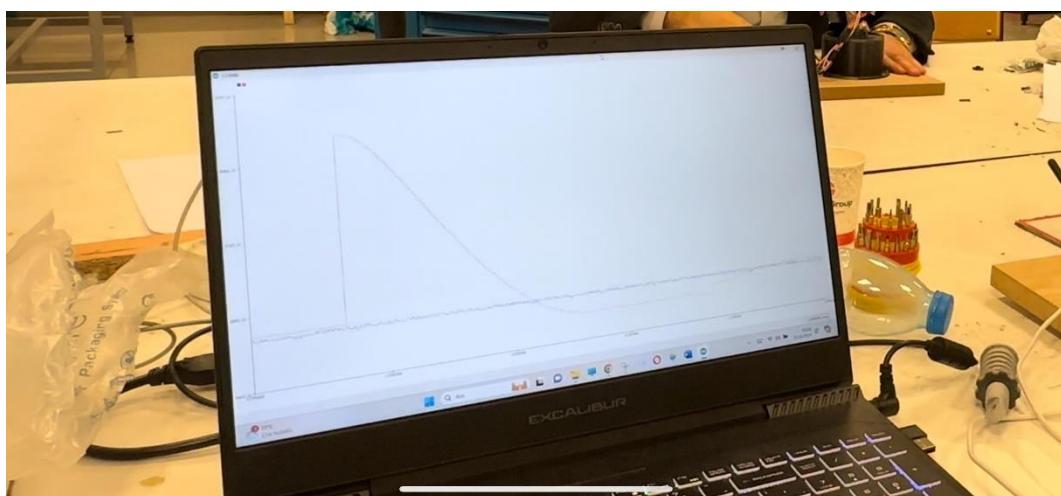


**Figure 4.90 Master Third Joint Angle – Slave Third Joint Angle Comparison Output**

#### 4.4.5 Control Output Graphs in The Real Prototype

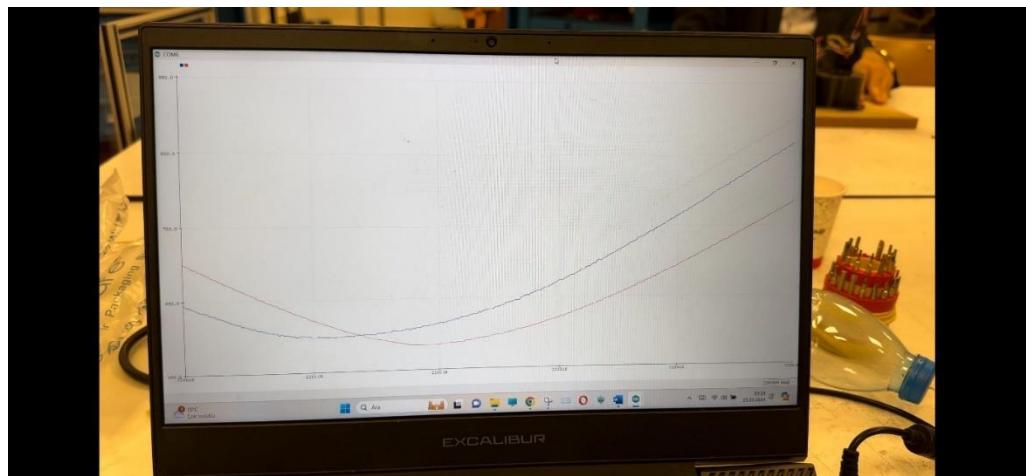
These graphs represent the system response while the robot is operating in real-time. These outputs depict the system response for a single joint, with similar results observed for other joints. The horizontal axis indicates the baud rate (2000000), while the vertical axis reads the magnetic encoder output ranging from 0 to 1023.

Below graph illustrates the output as the Slave Arm attempts to move instantaneously from its current position at 90 degrees to the 0 position of the Master Arm. It can be likened to providing a step response to the Slave Arm joint.4.91



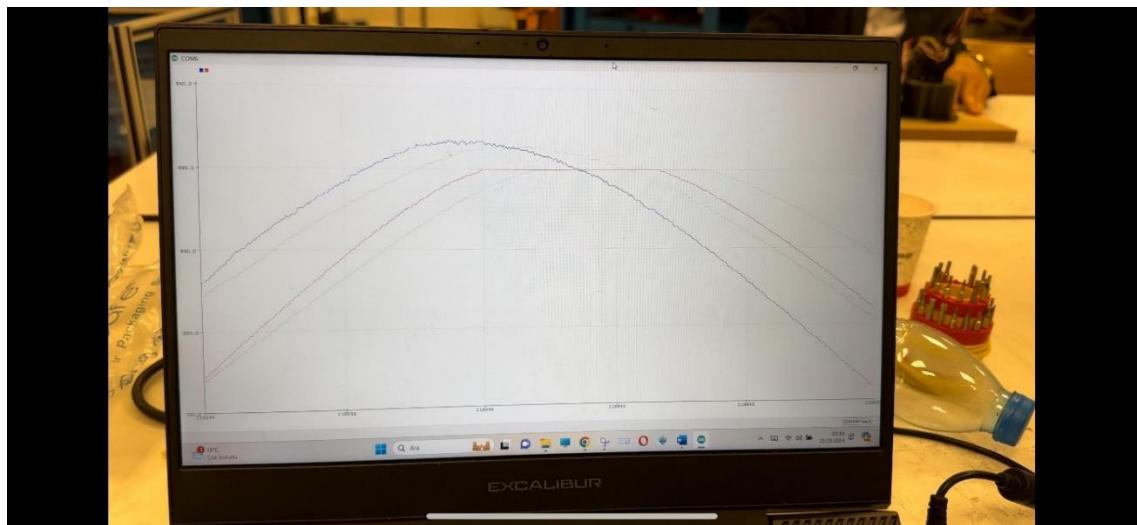
**Figure 4.91 Step Response Output of the Slave Arm of One of the Joints**

Below graph, we observe the output graph of the slave arm, which tracks the periodic motion of the master arm.



**Figure 4.92 Sine Wave Response of the Slave Arm of One of the Joints**

Below graph, we observe the situation where the slave arm stops following the master arm after reaching a predetermined limit. The blue curve represents the master arm, while the red curve represents the slave arm.



**Figure 4.93 Slave Arm Reaching to the Predetermined Limit Shown in Red Curve**

## 4.5 Overall System View

### 4.5.1 Model of the Overall System

The comprehensive version of all the stages shown above is visible below. The process, starting with the operator's physical manipulations in the Physical Stage, is illustrated to progress towards the Electronical Stage, then the Control Stage, and finally reaching the Mechanical Stage.

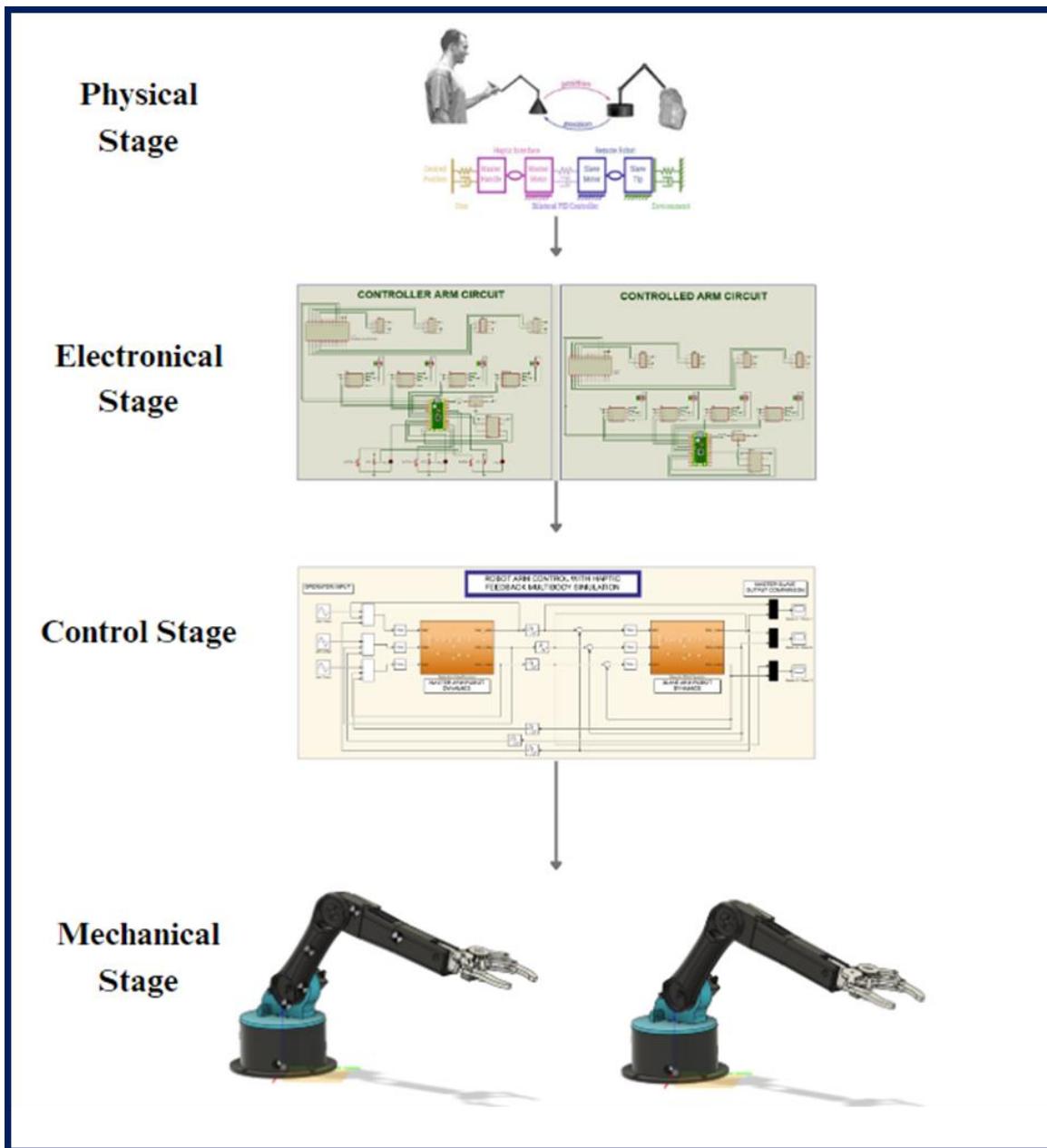
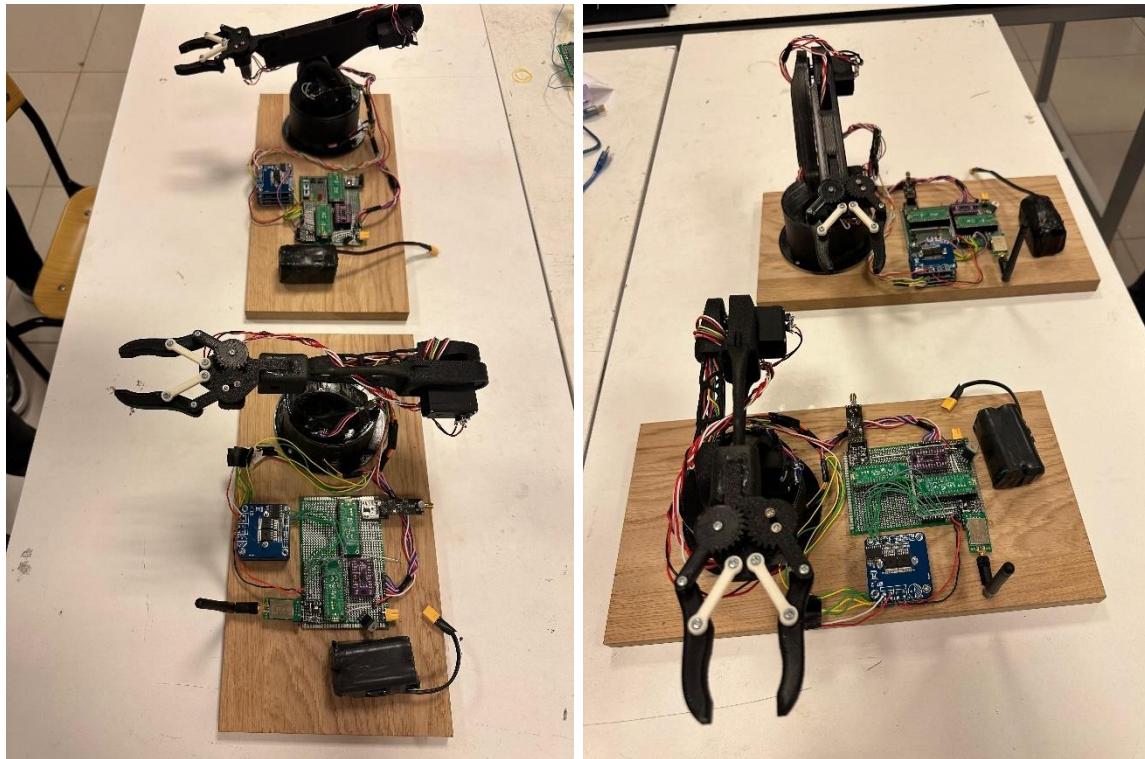


Figure 4.94 Overall System View

#### 4.5.2 Overall System View in Real

The visuals below depict an overall system view encompassing the power, electronic, communication, and mechanical components of the robot arms. The robot arms are mounted on a wooden base.

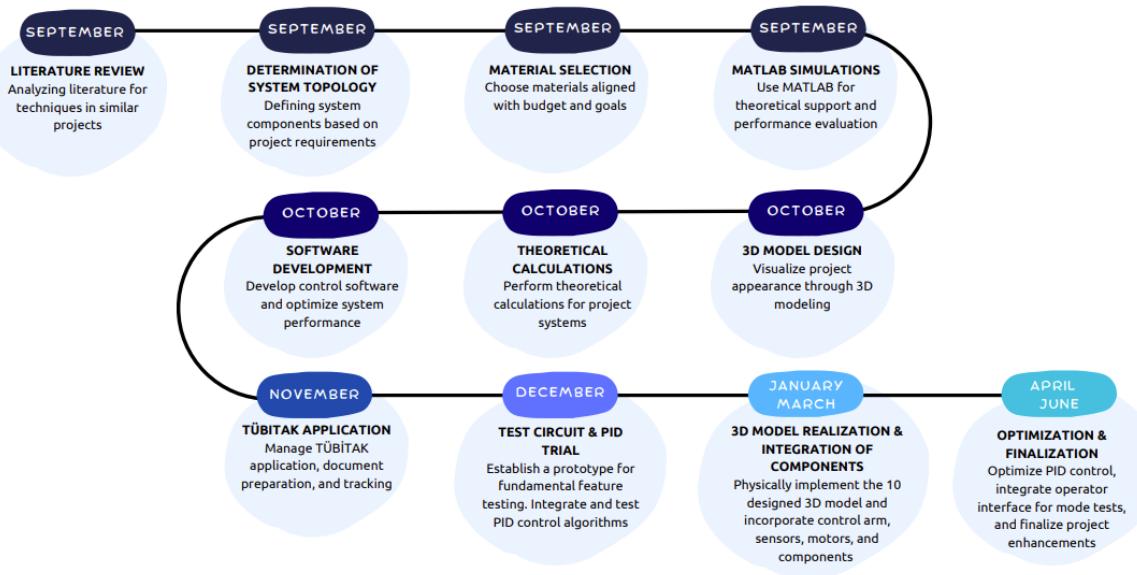


**Figure 4.95 Overall System View in Real**

## 5. WORK SCHEDULE

**Table 5-1 - Gantt Chart of The Project**

	October	November	December	January	February	March	April	May
<b>Literature Review &amp; Source Investigation</b>								
<b>Determination of Scope and Requirements</b>								
<b>Determination of System Topology</b>								
<b>MATLAB Simulations</b>								
<b>Theoretical Calculations</b>								
<b>3D Model Design</b>								
<b>Software Development</b>								
<b>Test Circuit &amp; PID Trial</b>								
<b>3D Model Realization &amp; Integration of Components</b>								
<b>Optimization &amp; Finalization</b>								

**Figure 5.1 Road Map**

## 6. BUDGET

### 6.1 Material List

Budget Type	Requested Budget Amount (TRY)	Justification for Request
<b>Consumable Materials</b>	650	Connection cables and accessories, Soldering materials, Adhesive, 3 pieces of Double-Sided Perforated Pertinax, 2 Wooden Boards
<b>Machinery/Equipment (Fixed Assets)</b>	9150	8 pieces of Orion 18650 3.7V 3200mah 3c Rechargeable Li-ion Batteries, 2 pieces of 2S 20A 8.4V BMS 18650 Lithium Battery Charging Protection Modules, 6 pieces of MG995R Servo Motors, 2 pieces of SG90 9G Mini Servo Motors, 8 pieces of AS5600 Magnetic Encoders, 8 pieces of BTS7960 Motor Drivers, 2 pieces of LM7805 Voltage Regulators, 4 pieces of NRF24L01+PA+LNA SMA Antenna 2.4GHz 5km Wireless Communication Modules, 2 pieces of Raspberry Pi Pico, 2 pieces of Raspberry Pico W, Mechanical Parts for Arm (3D printer filament), 3 pieces of LED, 6 pieces of button
<b>Service Procurement</b>	-	-
<b>Transportation</b>	-	-
<b>TOTAL</b>	9800	-

### 6.2 Service Procurements

No service procurement has been made.

### 6.3 External Support Application

An application has been submitted for the TÜBİTAK-2209-A University Students Research Projects Support Program, and we are awaiting the results.

## CONCLUSION

In summary, this thesis has outlined the initial design considerations and objectives for a haptic feedback-controlled robotic arm tailored specifically for laboratory applications. The primary goal of this design phase is to lay the foundation for the development of a system that aims to minimize direct human-object interaction, enhance automation, and ultimately improve the user experience through the integration of effective haptic feedback technology. The proposed system holds the promise of offering a realistic sense of touch, thereby facilitating precise and accurate control over laboratory procedures. While the project is still in its nascent stages, the envisioned outcomes include increased laboratory process automation, improved operational efficiency, and enhanced safety measures for human-machine interaction. The low latency and high precision exhibited in the initial design phase suggests potential for real-time responsiveness, contributing to overall user satisfaction. Looking ahead, this project sets the stage for further refinement and development, with future work focusing on the implementation, testing, and optimization phases. The findings and insights gained during this design phase will guide subsequent steps toward the realization of a fully functional haptic feedback-controlled robotic arm for laboratory applications.

## REFERENCES

- [1] Srinivasan, M. A., & Basdogan, C., “Haptic feedback: A brief history from telepresence to virtual reality,” *Haptics-e: The Electronic Journal of Haptics Research*, 1997.
- [2] Okamura, A. M., “Haptic feedback in robot-assisted minimally invasive surgery,” *Current Opinion in Urology*, 19(1), 102-107, 2009.
- [3] Okamura, A. M., Richard, A., & Cutkosky, M. R., “Feeling is Believing: A Laboratory Module on Haptic Perception Using Haptic Feedback Devices,” *Haptics-e: The Electronic Journal of Haptics Research*, 2016.
- [4] Saha, A., Saha, S., & Saha, M., “Automation in laboratory medicine: A revolutionary step in the improvement of quality and efficiency,” *Journal of Laboratory Physicians*, 8(2), 55-61, 2016.
- [5] Romano, J. M., Hsiao, K., Niemeyer, G., Chitta, S., & Kuchenbecker, K. J., “Human-inspired robotic grasp control with tactile sensing,” *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, 2011.
- [6] “Haptic feedback for improved robotic arm control during simple grasp, slippage, and contact detection tasks”, *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 16-21 May 2016.
- [7] Rodríguez-Sedano, F.J., Conde, M.Á., Rodríguez-Lera, F.J. et al., “Measuring the impact of haptic feedback in collaborative robotic scenarios”, *Universal Access Inf Soc*, 2023.
- [8] “Robotic arm kinematics and bilateral haptic feedback over an ethernet communications link”, *2013 IEEE International Conference on Automation Science and Engineering (CASE)*, 17-20 August 2013.
- [9] R. J. Stone, “Haptic feedback: A potted history, from telepresence to virtual reality,” in *Workshop on Haptic Human-Computer Interaction*, ser. LNCS, S. Brewster and R. Murray-Smith, Eds., vol. 2058. Glas-gow, UK: Springer-Verlag Berlin Heidelberg, 2000, pp. 1–7.
- [10] G. C. Burdea and N. A. Langrana, “Virtual force feedback: Lessons, challenges, future applications,” *IEEE/ASME Trans. Mechatron.*, vol. 5,no. 2, pp. 178–182, 1993.

## APPENDIX

Below is the MATLAB codes to obtain the dynamic equations and calculations.

```
% symbolic joint angles
syms theta_1 theta_2 theta_3
syms lc1 lc2 lc3 l2
syms m1 m2 m3
% links inertia moments
syms Ic1 Ic2 Ic3 Ics1
% joint angular velocities
syms theta_dot_1 theta_dot_2 theta_dot_3
% joints angular accelerations
syms theta_ddot_1 theta_ddot_2 theta_ddot_3
% gravitational acceleration
syms g

adot1=theta_dot_1
adot2=theta_dot_2
adot3=theta_dot_3

a_ddot1=theta_ddot_1
a_ddot2=theta_ddot_2
a_ddot3=theta_ddot_3

a1=theta_1 , a2=theta_2 , a3=theta_3;

% -----velocity jacobians for each link-----

% combined center vector of first link and fist servo wrt base 0

0_c1=[ 0; 0; 0 ; ]
```

```
% velocity jacobian of 0_c1

Jv1=[ diff(0_c1(1),a1),diff(0_c1(1),a2),diff(0_c1(1),a3);

       diff(0_c1(2),a1),diff(0_c1(2),a2),diff(0_c1(2),a3);

       diff(0_c1(3),a1),diff(0_c1(3),a2),diff(0_c1(3),a3);
]

Jv1t= transpose(Jv1)

d1=m1*Jv1t*Jv1

% combined center vector of link 2 wrt base 0

0_c2=[ lc2*cos(a1)*cos(a2);
        -lc2*sin(a1)*cos(a2);
        lc2*sin(a2)      ;
]

% velocity jacobian of 0_c2

Jv2=[ diff(0_c2(1),a1),diff(0_c2(1),a2),diff(0_c2(1),a3);

       diff(0_c2(2),a1),diff(0_c2(2),a2),diff(0_c2(2),a3);

       diff(0_c2(3),a1),diff(0_c2(3),a2),diff(0_c2(3),a3);]

Jv2t= transpose(Jv2)
```

$d2=m2*Jv2t*Jv2$

% combined center vector of link3,servo3,load and gripper\_servo wrt base 0

```
0_c3=[ cos(a1)*(l2*cos(a2)+lc3*cos(a2+a3));
        -sin(a1)*(l2*cos(a2)+lc3*cos(a2+a3));
        l2*sin(a2)+lc3*sin(a2+a3)      ;
        ]
```

% velocity jacobian of 0\_c3

```
Jv3=[ diff(0_c3(1),a1),diff(0_c3(1),a2),diff(0_c3(1),a3);
       diff(0_c3(2),a1),diff(0_c3(2),a2),diff(0_c3(2),a3);
       diff(0_c3(3),a1),diff(0_c3(3),a2),diff(0_c3(3),a3);]
```

$Jv3t= transpose(Jv3)$

$d3= m3*Jv3t*Jv3$

% -----angular velocity jacobians for each link-----

% angular velocity jacobian of 0\_c1

```
Jw1=[  
      0,0,0;  
      0,0,0;  
      1,0,0;  
      ]
```

$Jw1t=transpose(Jw1)$

```
d4=Jw1t*Ic1*Jw1
```

```
% angular velocity jacobian of 0_c2
```

```
Jw2=[  
    0,0,0;  
    0,1,0;  
    1,0,0;    ]
```

```
Jw2t=transpose(Jw2)
```

```
d5=Jw2t*Ic2*Jw2
```

```
% angular velocity jacobian of 0_c3
```

```
Jw3=[  
    0,0,0;  
    0,1,1;  
    1,0,0;  
]
```

```
Jw3t=transpose(Jw3)
```

```
d6=Jw3t*Ic3*Jw3
```

```
% lagrange inertial matrix D(q)
```

```
D=d1+d2+d3+d4+d5+d6
```

```
%----- Christoffel coefficient for C(q,q_dot) matrix---  
  
c111=0.5* ( diff(D(1,1),a1) + diff(D(1,1),a1) - diff(D(1,1),a1) )*adot1*adot1  
  
c112=0.5* ( diff(D(1,2),a1) + diff(D(1,1),a2) - diff(D(1,2),a1) )*adot1*adot2  
  
c113=0.5* ( diff(D(1,3),a1) + diff(D(1,1),a3) - diff(D(1,3),a1) )*adot1*adot3  
  
c121=0.5* ( diff(D(1,1),a2) + diff(D(1,2),a1) - diff(D(2,1),a1) )*adot2*adot1  
  
c122=0.5* ( diff(D(1,2),a2) + diff(D(1,2),a2) - diff(D(2,2),a1) )*adot2*adot2  
  
c123=0.5* ( diff(D(1,3),a2) + diff(D(1,2),a3) - diff(D(2,3),a1) )*adot2*adot3  
  
c131=0.5* ( diff(D(1,1),a3) + diff(D(1,3),a1) - diff(D(3,1),a1) )*adot3*adot1  
  
c132=0.5* ( diff(D(1,2),a3) + diff(D(1,3),a2) - diff(D(3,2),a1) )*adot3*adot2  
  
c133=0.5* ( diff(D(1,3),a3) + diff(D(1,3),a3) - diff(D(3,3),a1) )*adot3*adot3  
  
C1=c111+c112+c113+c121+c122+c123+c131+c132+c133
```

%-----

$$c211=0.5 * (\text{diff}(D(2,1),a1) + \text{diff}(D(2,1),a1) - \text{diff}(D(1,1),a2)) * \text{adot1} * \text{adot1}$$

$$c212=0.5 * (\text{diff}(D(2,2),a1) + \text{diff}(D(2,1),a2) - \text{diff}(D(1,2),a2)) * \text{adot1} * \text{adot2}$$

$$c213=0.5 * (\text{diff}(D(2,3),a1) + \text{diff}(D(2,1),a3) - \text{diff}(D(1,3),a2)) * \text{adot1} * \text{adot3}$$

$$c221=0.5 * (\text{diff}(D(2,1),a2) + \text{diff}(D(2,2),a1) - \text{diff}(D(2,1),a2)) * \text{adot2} * \text{adot1}$$

$$c222=0.5 * (\text{diff}(D(2,2),a2) + \text{diff}(D(2,2),a2) - \text{diff}(D(2,2),a2)) * \text{adot2} * \text{adot2}$$

$$c223=0.5 * (\text{diff}(D(2,3),a2) + \text{diff}(D(2,2),a3) - \text{diff}(D(2,3),a2)) * \text{adot2} * \text{adot3}$$

$$c231=0.5 * (\text{diff}(D(2,1),a3) + \text{diff}(D(2,3),a1) - \text{diff}(D(3,1),a2)) * \text{adot3} * \text{adot1}$$

$$c232=0.5 * (\text{diff}(D(2,2),a3) + \text{diff}(D(2,3),a2) - \text{diff}(D(3,2),a2)) * \text{adot3} * \text{adot2}$$

$$c233=0.5 * (\text{diff}(D(2,3),a3) + \text{diff}(D(2,3),a3) - \text{diff}(D(3,3),a2)) * \text{adot3} * \text{adot3}$$

$$C2=c211+c212+c213+c221+c222+c223+c231+c232+c233$$

%-----

$$c311=0.5 * (\text{diff}(D(3,1),a1) + \text{diff}(D(3,1),a1) - \text{diff}(D(1,1),a3)) * \text{adot1} * \text{adot1}$$

$$c312=0.5 * (\text{diff}(D(3,2),a1) + \text{diff}(D(3,1),a2) - \text{diff}(D(1,2),a3)) * \text{adot1} * \text{adot2}$$

$$c313=0.5 * (\text{diff}(D(3,3),a1) + \text{diff}(D(3,1),a3) - \text{diff}(D(1,3),a3)) * \text{adot1} * \text{adot3}$$

$$c321=0.5 * (\text{diff}(D(3,1),a2) + \text{diff}(D(3,2),a1) - \text{diff}(D(2,1),a3)) * \text{adot2} * \text{adot1}$$

$$c322=0.5 * (\text{diff}(D(3,2),a2) + \text{diff}(D(3,2),a2) - \text{diff}(D(2,2),a3)) * \text{adot2} * \text{adot2}$$

$$c323=0.5 * (\text{diff}(D(3,3),a2) + \text{diff}(D(3,2),a3) - \text{diff}(D(2,3),a3)) * \text{adot2} * \text{adot3}$$

$$c331=0.5 * (\text{diff}(D(3,1),a3) + \text{diff}(D(3,3),a1) - \text{diff}(D(3,1),a3)) * \text{adot3} * \text{adot1}$$

$$c332=0.5 * (\text{diff}(D(3,2),a3) + \text{diff}(D(3,3),a2) - \text{diff}(D(3,2),a3)) * \text{adot3} * \text{adot2}$$

$$c333=0.5 * (\text{diff}(D(3,3),a3) + \text{diff}(D(3,3),a3) - \text{diff}(D(3,3),a3)) * \text{adot3} * \text{adot3}$$

$$C3=c311+c312+c313+c321+c322+c323+c331+c332+c333$$

%----- Centrifugal and coriolis matrix C(q,q\_dot)

$$C=[C1;C2;C3]$$

```
%----gravity matrix  
  
g_0=[0;0;-g]  
  
G=-(m1*Jv1t + m2*Jv2t + m3*Jv3t)*g_0  
  
%----- Lagrange Dynamic equations of RRR robot arm -----  
  
a_ddot=[ a_ddot1;  
         a_ddot2;  
         a_ddot3;  
         ]  
  
T=D*a_ddot + C + G
```

## ÖZGEÇMİŞ

Kronolojik sıraya göre eğitim ve öğretimine ve varsa çalıştığı yerlere ilişkin bilgileri içermelidir.