



الجمهورية الجزائرية الديمقراطية الشعبية
People's Democratic Republic of Algeria
وزارة التعليم العالي والبحث العلمي
Ministry of Higher Education and Science Research
جامعة غرداية
University of ghardaia
كلية العلوم والتكنولوجيا
Faculty of Science and Technology
قسم الآلية والكهروميكانيك
Department of Automatics and Electromechanics



Registration N
/.../.../.../.../...

Master's Thesis in Automation and Systems Engineering, in accordance with the amended
Ministerial Decree No. 008 Supplementing Ministerial Decision No. 12/75.

Master & start-up

Faculty: Science and technology

Field: Automatic

Major: Automatic and systems

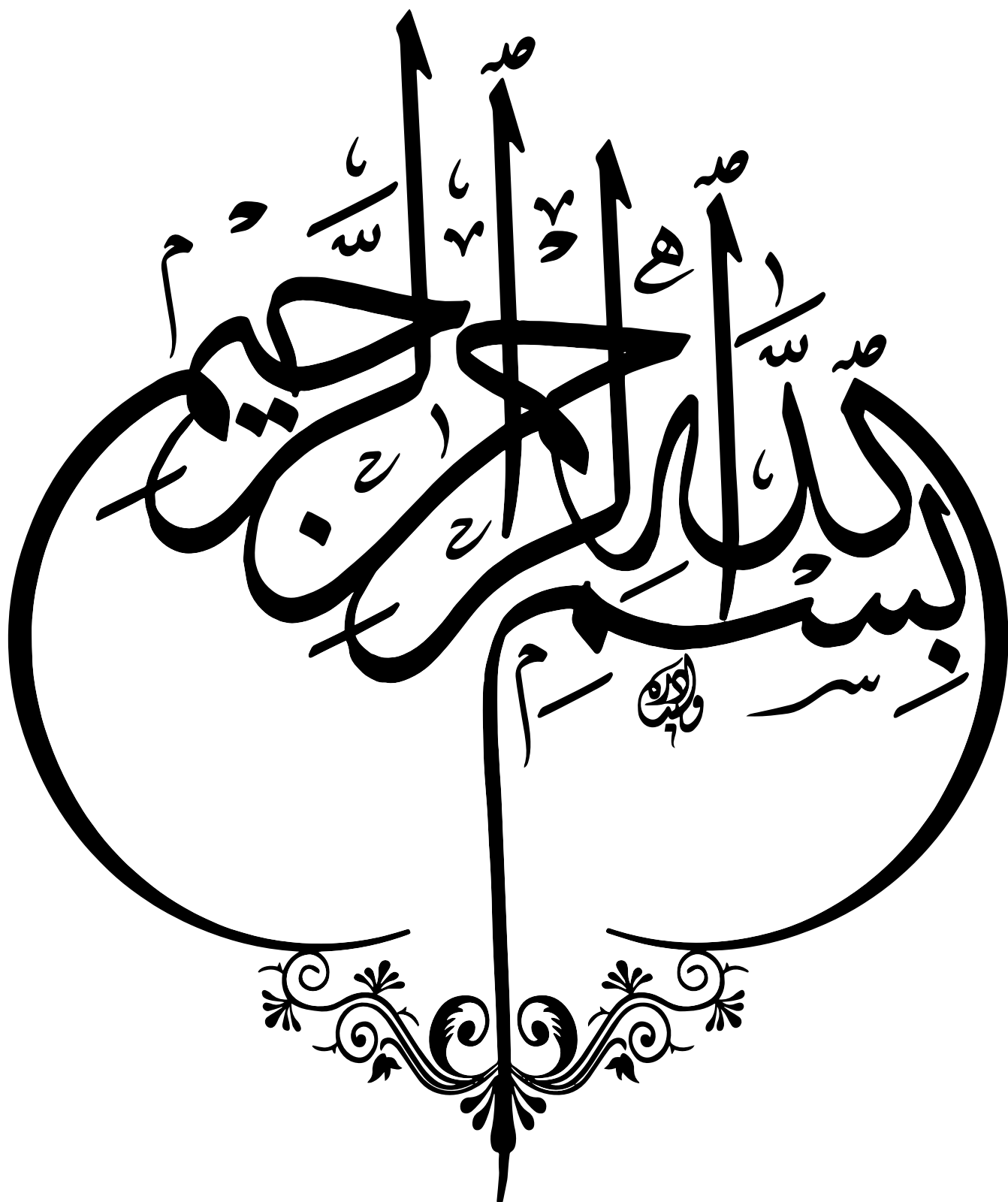
Modeling, Design and Control of a 4-arm Delta Parallel Robot

Presented by: Hamel Soundous, Moulay Omar Afrah

Publicly defended on 21/06/2026

Dr.	Boukhari Hamed	MCA	Univ of Ghardaia	President
Dr.	Mosbah Charaf Abdelkarim	MCA	Univ of Ghardaia	Examiner
Dr.	Rebai Aissa	MCB	Univ of Ghardaia	Examiner
Dr.	Euldji Rafik	MCB	Univ of Ghardaia	Supervisor

Academic Year 2025/2026



Dedication

First, I thank God, the Almighty, for granting me the strength, patience, and ability to complete this work. I am deeply grateful for all His blessings.

I dedicate this work to my beloved mother, whose love, support, and sacrifices have been my greatest source of strength. To my late father, who I dearly wish were here to witness this moment, I know in my heart that he would be proud.

My sincere thanks go to my brothers and sisters, especially my elder brother, for his constant support and help throughout this journey. Your effort, even to the point of personal sacrifice, means more to me than words can express.

I am also deeply grateful to my best friend, who stood by my side and supported me through every step of this experience.

I would like to express my sincere gratitude to my supervisor, Dr. Euldji Rafik, for his guidance, support, and valuable advice throughout the realization of this work.

I would like to express my appreciation to our professors, Dr. Boukhari, Dr. Mosbah charaf abdelkarim, Dr. Hassan Nacer, and Dr. Boughelga, Dr. BaKAR for their guidance, support, and encouragement when it was most needed.

Special thanks to the laboratory engineer, Reciou M, who was like a sister to us. Her kindness, support, and continuous help, especially in providing materials, played an essential role in completing this work.

Finally, I dedicate this work to myself — for the perseverance, strength, and determination to keep going despite all challenges.

This achievement is the result of all your support, and I am forever grateful.

Soundous

Dedication

To the one who taught me that the mind is a trust and knowledge is a mission...

To myself, who endured the long nights of research, believed in the goal when the path
grew narrow,
and refused to stop short of the finish line.

To the one who carried me before I ever held a pen...

To my beloved mother, an endless spring of tenderness,
whose prayers were my secret strength in every step.

To the one who taught me the meaning of perseverance and giving...

To my dear father, who paved my path with his hard work,
and was my first support and my last refuge.

To those who shared my joy of success before it was even achieved...

To my cherished family, each one of them a candle that lit my way.

To all of you, I dedicate the fruit of my efforts,
this humble thesis that carries between its lines a piece of your souls.

Afrāh

Acknowledgements

First and foremost, we express our deepest gratitude to Allah, the Almighty, for granting us the strength, patience, and perseverance required to accomplish this work.

We would like to extend our sincere appreciation to our supervisor, Dr. Euldji Rafik, for his invaluable guidance, continuous support, insightful advice, and encouragement throughout the realization of this thesis.

We are also grateful to , Dr.Boukhari, Dr.Mosbah charaf abdelkarim, Dr.Hassan Nacer,and Dr.Boughelga, for their time, valuable comments, and constructive suggestions, which have significantly contributed to the improvement of this work.

Our sincere thanks are extended to the laboratory engineer,Reciouï M, for her assistance and support in providing the necessary resources and materials required for the successful completion of this project.

Finally, we would like to thank all those who contributed, directly or indirectly, to the completion of this work. Their support and encouragement are deeply appreciated.

المخلص

يهدف هذا المشروع إلى دراسة، تصميم، والتحكم في روبوت متوازي من نوع "دلتا" ذو أربعة أذرع لتطبيقات الأتمتة الذكية، حيث استهلكت الدراسة بالنمذجة الكينماتيكية والديناميكية وتحليل حيز الحركة في بيئة MATLAB، متبوعةً بالتصميم الميكانيكي الرقمي واختبار تداخل الأجزاء للهيكل باستخدام برنامج SolidWorks، بالإضافة إلى هندسة وتوجيه مسارات بطاقة التحكم الإلكترونية ومحاكاة دوائرها عبر برنامج Proteus. وبعد الانتقال من المحاكاة إلى الإنجاز الواقعي والتصنيع الفعلي لأجزاء الروبوت، تم بناء نظام هجين للقيادة والتحكم يربط بين قدرات المعالجة الرياضية لبرنامج MATLAB والتحكم اللحظي لوحدة الإدخال والإخراج في لوحة Arduino عبر الاتصال التسلسلي؛ لتُظهر النتائج التجريبية النهائية كفاءة استثنائية، واستقراراً ديناميكياً، ودقة عالية في تتبع المسارات تم إثباتها عملياً عبر أربعة تطبيقات متكاملة تشمل: الرسم ثنائي الأبعاد بدقة حركية عالية، والتحكم اللاسلكي عن بُعد باستخدام تقنية البلوتوث، ونظام الرؤية الحاسوبية لتتبع وفرز الأجسام ذات اللون الأزرق آنياً عبر الكاميرا، وأخيراً محاكاة خطوط الإنتاج الصناعية عبر تطبيق دورة الالتقاط والوضع (Pick-and-Place) السريعة.

الكلمات المفتاحية: الروبوت المتوازي، روبوت دلتا، SolidWorks، MATLAB، Proteus، Arduino، الرؤية الحاسوبية، تتبع الألوان، الالتقاط والوضع، التحكم اللاسلكي.

Abstract

This project aims to study, design, and control a four-arm delta parallel robot for intelligent automation applications. The study began with kinematic and dynamic modeling and motion space analysis in MATLAB, followed by digital mechanical design and component interlocking testing of the structure using SolidWorks. The electronic control board's paths were then engineered and routed, and its circuitry simulated using Proteus. After transitioning from simulation to real-world implementation and actual manufacturing of the robot's components, a hybrid drive and control system was built. This system combines the mathematical processing capabilities of MATLAB with real-time control of input/output modules on an Arduino board via serial communication. The final experimental results demonstrated exceptional efficiency, dynamic stability, and high path-tracking accuracy, which were practically proven through four integrated applications:

high-resolution 2D motion drawing, wireless remote control using Bluetooth, a computer vision system for real-time tracking and sorting of blue objects via a camera, and finally, simulation of industrial production lines using a rapid pick-and-place cycle.

Keywords: Parallel robot, Delta robot, SolidWorks, MATLAB, Proteus, Arduino, computer vision, color tracking, pick-and-place, wireless control

Résumé

Ce projet vise à étudier, concevoir et commander un robot parallèle « delta » à quatre bras destiné à des applications d'automatisation intelligente. L'étude a débuté par la modélisation cinématique et dynamique, ainsi que par l'analyse de l'espace de travail sous MATLAB. Elle s'est poursuivie par la conception mécanique CAO et les tests d'assemblage sous SolidWorks, parallèlement au routage et à la simulation des circuits de la carte de commande électronique via Proteus.

Après la phase de fabrication et d'assemblage physique du prototype, un système de commande hybride a été développé. Ce système associe la puissance de calcul de MATLAB au contrôle en temps réel des entrées/sorties d'une carte Arduino via une communication série. Les résultats expérimentaux finaux démontrent une excellente efficacité, une grande stabilité dynamique et une haute précision dans le suivi de trajectoire. Ces performances ont été validées à travers quatre applications intégrées : le tracé de trajectoires 2D haute résolution, le contrôle sans fil via Bluetooth, un système de vision par ordinateur pour le suivi et le tri d'objets en temps réel, et enfin, la simulation d'une ligne de production industrielle via un cycle de transfert rapide (*pick-and-place*).

Mots-clés : Robot parallèle, Robot Delta, SolidWorks, MATLAB, Proteus, Arduino, Vision par ordinateur, Tri couleur, *Pick-and-place*, Contrôle sans fil

Contents

Dedications	iii
Acknowledgements	v
Abstract	vi
List of Figures	xi
List of Tables	xiv
General Introduction	1
1 Introduction to Industrial Robots and Delta Manipulators	3
1.1 Introduction	3
1.2 Industrial Robots	3
1.3 Different Types of Robots	4
1.3.1 fixed robots	4
1.3.2 Serial Manipulator Robots	5
1.3.3 Parallel Manipulator Robots	6
1.3.4 Two Generations of Parallel Robots	8
1.3.5 Hybrid Manipulator Robots (Serial–Parallel)	9
1.4 mobile robot	9
1.4.1 Definition	9
1.4.2 Key Categories (classifications)	10
1.5 Actuators used in robotics	11
1.5.1 Pneumatic	11
1.5.2 Hydraulic actuators	12
1.5.3 Electrical actuators	12
1.5.4 Mechanical actuators	13
1.6 Characteristics of Robots	13
1.6.1 Precision	13
1.6.2 Repeatability	13
1.6.3 Absolute Accuracy	13
1.6.4 Resolution	14
1.6.5 Relationship Between Precision, Repeatability, and Resolution	14

1.7	The Delta Robots	15
1.7.1	The Kinematic Diagram	16
1.7.2	The layout graph	17
1.7.3	Singular Configurations of Delta Parallel Robots	18
1.8	Examples of Realizations and Performances of Delta Robots	19
1.9	ROMI Linear Delta - Ultra-Precision Platform	20
1.10	3D-Printed Open-Source Delta - Educational Platform	21
1.11	DOF UAV-Mounted System	21
1.12	Five-DOF Delta-Like Robot - Industrial Packaging System	22
1.13	DeltaFlex - Additively Manufactured Compliant-Joint System	23
1.14	Motivation	24
1.15	Conclusion	24
2	Mathematical model of the parallel robot	25
2.1	Introduction	25
2.2	System Description	25
2.3	Delta robot design	26
2.4	Mathematical models of the Delta robot	27
2.4.1	Geometric model (MGD)	28
2.4.2	Inverse kinematics (IK)	30
2.4.3	Forward kinematics (FK)	32
2.4.4	Singularity Analysis	34
2.5	Dynamic	36
2.5.1	Introduction	36
2.5.2	Dynamic Modeling	36
2.5.3	Numerical Results	36
2.5.4	Discussion	37
2.5.5	Workspace Analysis	37
2.5.6	Detailed Analysis and Results	38
2.6	Conclusion	38
3	Hardware realisation	39
3.1	Introduction	39
3.2	Mechanical Parts	39
3.2.1	Robot Geometric Characterization	39
3.3	Mechanical Construction	40
3.3.1	Moving Platform	41
3.3.2	Ball Joints	41
3.3.3	Lower Arms AB	42
3.3.4	Driving Arms PA	42

3.3.5	Servomotors	42
3.3.6	Supports	43
3.4	Internal Components of The servo	44
3.4.1	Fixed Base	45
3.4.2	Fixing the base to a support	46
3.5	The electrical part	46
3.5.1	Power Supply Circuit	46
3.5.2	The electronic Circuit	47
3.5.3	components	48
3.6	Conclusion	52
4	Software Development and Control using Image Processing	53
4.1	Introduction	53
4.2	Image Acquisition and Vision System	53
4.2.1	Image Processing and Color Detection	54
4.2.2	Calibration and Coordinate Transformation	58
4.3	MATLAB to Arduino Communication	60
4.4	Control System Implementation	63
4.4.1	Control Logic Overview	63
4.4.2	Servo Control and Offset Calibration	63
4.5	Servomotor Control	63
4.6	conclusion	64
5	Applications and Performance Evaluation of the Delta Robot	65
5.1	Introduction	65
5.2	Delta Robot Performance	65
5.2.1	Workspace (Working Volume)	66
5.2.2	Speed of the Delta Robot	68
5.2.3	Accuracy	69
5.2.4	The maximum load	70
5.3	Practical applications of the robot	71
5.3.1	pick and place	71
5.3.2	Blue Color Tracking Using Vision-Based Control	73
5.3.3	the delta robot for 2D shape creation	73
5.3.4	Bluetooth-Based Wireless Control	74
5.4	Conclusion	75
	General Conclusion	76
	Bibliography	78

List of Figures

1.1	General Block Diagram of a Mechatronic System [5]	4
1.2	IRB 4400 - Fast, Compact and Versatile Industrial Robot	5
1.3	The Hexapod-System robot, photo and layout diagram	6
1.4	The FlexPicker robot (ABB), photo and layout diagram	8
1.5	YuMi .Abb’s Dual -arm robot	9
1.6	IMI Norgren ISOLine tie rod double acting cylinders	11
1.7	hydraulic actuator valve	12
1.8	Electrical actuators	12
1.9	mechanical actuators	13
1.10	Accuracy, resolution, and repeatability (according to Niaritsiry [Niaritsiry, 2006])	14
1.11	The DELTA robot and its main dimensions	16
1.12	Standardized Joint Representations	17
1.13	Enter Caption	17
1.14	DNC ISO 6431 Pneumatic Standard cylinder	18
1.15	Harvard’s milliDelta Robot	20
1.16	ROMI Linear Delta	20
1.17	Neutron - Open Source 3D Printer	21
1.18	UAV frame with mounted drivers, companion computer and payload	22
1.19	Industrial Packaging System	23
1.20	A parametric model for tetra joints applied	24
2.1	UAV frame with mounted drivers, companion computer and payload	25
2.2	3D Model illustrating the mechanical design of the parallel Delta robot . .	26
2.3	Side view of the robot. Coordinate system assignment.	27
2.4	Side view of an arm	28
2.5	Frontal view of an arm.	29
2.6	Frontal view of an arm.	31
2.7	lateral view of an arm	33
2.8	Singularity map based on the Jacobian condition number.	35
2.9	Multi-view analysis of the workspace including Top (XY), Side (XZ), and Front (YZ) views, alongside a Z-coordinate frequency distribution	37
3.1	Schematic and actual photo of the components of the completed delta robot.	40

3.2	Actual photo of the Moving Platform.	41
3.3	Actual photo of a ball joint with it's dimensions.	41
3.4	Actual photo of the lower arm AB with its dimensions	42
3.5	Actual photo of The Driving Arm PA with it's dimensions.	42
3.6	the HS-5765 MH servo motor and its accessories	43
3.7	CAD technical drafting and dimensional specifications of the servomotor support bracket.	43
3.8	The components of a servomotor.	44
3.9	operating principle of a servomotor.	44
3.10	Actual photo showing how the servos are mounted on the base.	45
3.11	Auto CAD drawing showing the positioning and placement of the servomotor on the base.	45
3.12	real image of the support created	46
3.13	Design of the Fixed base and Servomotors for the 4-Arm Delta Robot . . .	46
3.14	Power-on of the ATX-450W power supply	47
3.15	Complete Electronic Circuit Schematic	47
3.16	Custom Control PCB Layout	47
3.17	3D Render of the Custom Control and Power Distribution Board	48
3.18	Arduino uno	48
3.19	Electromagnet	50
3.20	optocoupler	50
3.21	Transistor IRFZ44N	51
3.22	HC-05 Bluetooth Module Views	51
3.23	UART Connection between Arduino UNO and HC-05 Bluetooth Module .	52
4.1	Vision-Based Control Architecture	53
4.2	RGB image representation as three stacked color components (Red, Green, Blue).	54
4.3	Decomposition of the RGB image into its three color planes (Red, Green, Blue).	55
4.4	Detection of the target object using color-based segmentation in MATLAB. .	55
4.5	Extraction of object position in pixel coordinates from MATLAB processing output.	57
4.6	Pixel-to-coordinate transformation for robot workspace mapping.	57
4.7	Alignment between image and workspace coordinate systems.	58
4.8	Defining the origin at the center of the workspace.	59
4.9	Conversion from pixel values to real-world units.	60
4.10	Transmission of computed servo angles from MATLAB and corresponding response in simulation.	61

4.11	Conceptual representation of serial communication between MATLAB and Arduino.	61
4.12	Arduino program receiving and parsing serial data from MATLAB.	62
4.13	Servo-specific pulse width calibration parameters used to compensate for actuator offsets.	63
4.14	PWM Timing for different shaft positions.	64
5.1	Real prototype of the four-arm Delta robot.	65
5.2	Characteristic angles defining the workspace of the Delta robot.	66
5.3	Minimum and Maximum reach height of the Delta robot	67
5.4	Experimental workspace of the robot in the X–Y plane	68
5.5	Determination of the robot speed using video analysis.	69
5.6	Calculating the load for the upper arm.	70
5.7	Measurement of the test load.	71
5.8	Load attached to the end-effector.	71
5.9	pick and place application	72
5.10	Vision-based object detection and sorting system	73
5.11	2D shapes generated by the robot (four- and three-leaf clover and a square). factory system for object handling.	74
5.12	Transmission of joint angles via Serial Bluetooth Terminal application . . .	75

List of Tables

2.1	Delta robot design parameters.	27
2.2	Delta robot parameters.	30
3.1	Mechanical components and dimensions of the Delta robot.	40

General Introduction

Robotics has become a key technology in modern industry, enabling automation, precision, and efficiency in tasks ranging from manufacturing and packaging to assembly and material handling. Among the many types of robotic systems, parallel robots have shown particular promise for high-speed, high-precision applications, thanks to their unique multi-arm architecture. Despite the success of conventional 3-arm delta robots in industrial environments, they face limitations in workspace, payload capacity, and stability during high-speed operation [1][2].

Robotics emerged in industry with the introduction of the first industrial robot, Unimate, in 1959. Early robots were mainly serial manipulators. In the 1980s, parallel robot structures were introduced to improve speed and accuracy, leading to the development of the Delta robot by Reymond Clavel in 1985.

To address these challenges, this thesis focuses on the design, modeling, and control of a 4-arm Delta parallel manipulator. By adding an additional kinematic arm, the proposed design improves load distribution, enhances structural rigidity, and expands the operational workspace. These enhancements aim to increase the system's performance and stability, making it better suited for demanding industrial tasks that require speed, accuracy, and Stability[3][4]

The primary goal of this research is to develop a complete robotic system capable of performing high-speed pick-and-place and precision manipulation tasks. The robot integrates mechanical, electrical, and control components. By incorporating a computer vision system, the focus expands from stable basic motion to intelligent, vision-guided positioning, ensuring robust performance through careful modeling, hardware implementation, and advanced image processing control.

Chapter 1: provides a general introduction to robotics, presenting serial, parallel, and hybrid robot architectures, their main advantages and limitations, and the background of delta robots. It also discusses the motivation for improving the delta robot structure and the benefits of a 4-arm design.

Chapter 2: introduces the mathematical modeling of parallel robots, including forward and inverse kinematics, dynamics, workspace and singularity analysis, highlighting their effects on robot performance and motion limitations.

Chapter 3: focuses on the realization of the 4-arm delta manipulator. This chapter covers the mechanical design, electrical components, and hardware implementation, including system assembly and circuit integration.

Chapter 4: centers on software and control using image processing. It discusses the integration of computer vision systems, with particular emphasis on camera calibration, ensuring alignment between image coordinates and the robot workspace for accurate positioning.

Chapter 5: presents applications and performance evaluation of the Delta manipulator. It showcases experimental results, including pick-and-place operations, vision-based object detection and sorting, and 2D shape generation, while evaluating the system in terms of speed, precision, and workspace, and the The maximum load and highlighting the improvements achieved with the 4-arm configuration compared to conventional designs

Chapter 1 | Introduction to Industrial Robots and Delta Manipulators

1.1 Introduction

In this chapter, different types of industrial robots are first presented, namely serial robots, parallel robots, and hybrid robots. The advantages and disadvantages of serial and parallel robots are discussed.

After that, the focus is placed on parallel robots, where their different types are introduced. Finally, the Delta robot is presented, along with its main applications and some of its existing versions.

1.2 Industrial Robots

A robotic manipulator, regardless of the task assigned to it (such as object transfer, welding, or assembly), is a mechanism capable of moving and positioning an object called the end effector or tool point within a region of space known as the workspace.

To distinguish the robots studied in this work from simple manipulators whose motions are limited by rigid mechanical stops, the term industrial robot is used. A general definition of an industrial robot is illustrated in Figure. 1.1 [5] .

An industrial robot is composed of a mechanical structure driven by actuators based on commands generated by a control unit. These commands depend on the information provided by the sensors.

The use of external sensors, known as exteroceptive sensors, to evaluate and measure the interaction between the robot and its environment directly at the end-effector is becoming increasingly common in high-precision robotic applications.

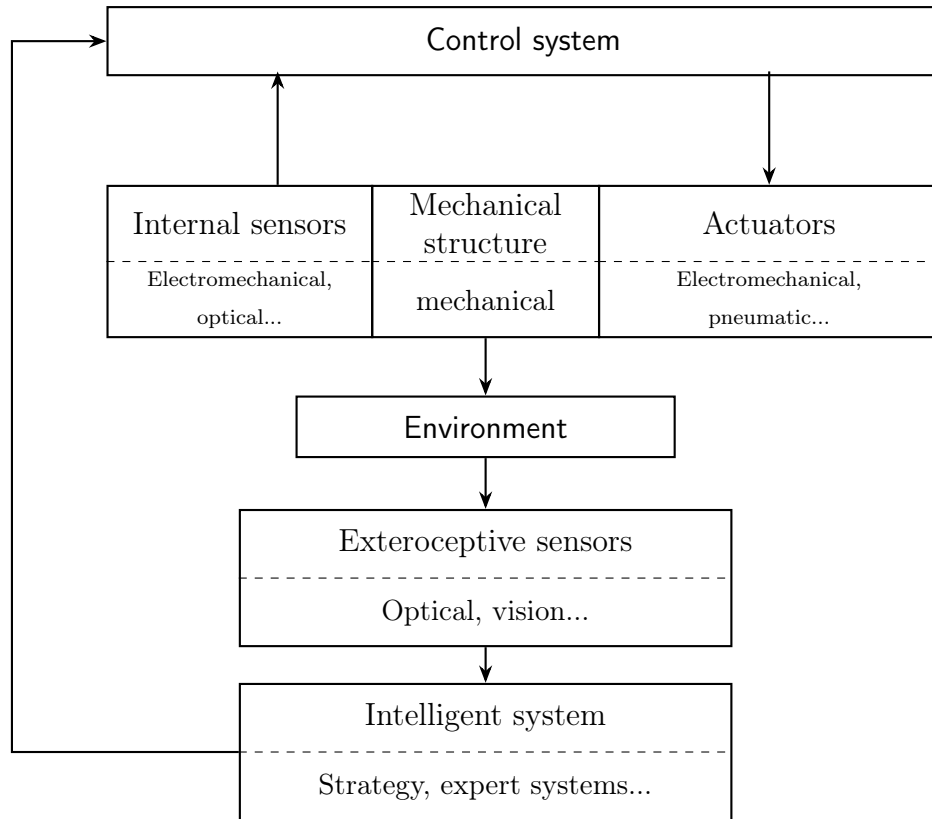


Figure 1.1: General Block Diagram of a Mechatronic System [5]

in Figure. 1.1 , two types of coordinate system commonly used in robotics can be identified. The first type corresponds to joint coordinates, also called generalized coordinates, which describe the configuration of the robot through the positions of its actuated joints. The second type corresponds to operational coordinates, also known as task-space coordinates, which define the position and orientation of the end-effector in the task reference frame. The direct and inverse geometric models allow for the transformation from one coordinate system to another.

The mechanical architecture represents the structure that connects the robot base to its end-effector. It is made up of rigid links connected by passive or active joints, depending on whether they are actuated or not [5].

1.3 Different Types of Robots

1.3.1 fixed robots

Fixed Robots are robotic systems permanently mounted to a stationary base (floor, ceiling, or workbench) that operate within a limited, defined workspace. They excel at precise and repetitive tasks such as welding, assembly, and pick-and-place operations

in manufacturing environments. Common examples include articulated robot arms and SCARA robots.

1.3.2 Serial Manipulator Robots

1.3.2.1 Definition

Most industrial robots developed to date are of the serial type, meaning that their moving structure is an open kinematic chain composed of a succession of rigid links connected by single-degree-of-freedom joints. Each joint is driven by an actuator located either in the joint itself or in one of the preceding links, as illustrated in Figure. 1.2.



Figure 1.2: IRB 4400 - Fast, Compact and Versatile Industrial Robot
[6]

Serial robots have the advantage of offering a large workspace and being relatively simple in terms of control-related computations. Their main disadvantages are as follows:

- High inertia, due to the distribution of masses along the entire kinematic chain (actuators and transmission components).
- Lack of rigidity, caused by the serial arrangement of elastic elements, as well as fatigue and wear of power transmission links supplying the actuators (such as cables and flexible hoses).
- Fatigue and wear of communication links that ensure information transfer between sensors and the control system. This aspect is particularly important in terms of operational safety, since transmission errors may have harmful consequences on the robot motion.

1.3.3 Parallel Manipulator Robots

For certain industrial applications such as high-speed machining and rapid handling, serial kinematic manipulators are no longer the most suitable solution. Indeed, in this type of architecture, each actuated axis must support the following one. As a result, the moving masses are relatively high, which negatively affects the dynamic performance of the system [5].

For this reason, parallel kinematic manipulators are increasingly used today. Their high dynamic performance and high load capacity are efficiently exploited in industrial environments, making parallel robots particularly well suited for demanding applications.

1.3.3.1 Definition

A parallel manipulator is a closed kinematic chain mechanism in which the end-effector is connected to the base through several independent kinematic chains. In 1947, a new structure known as the Gough platform was introduced. This revolutionary structure, initially used for tire testing, belongs to the family of hexapods and is now commonly known as the Stewart platform. Hexapods are widely used for the precise positioning and orientation of heavy loads. In such applications, the dynamic performance of the structure is not the primary criterion; rather, the objective is to achieve high positioning accuracy for massive objects. In the example shown in Figure. 1.3, the presented hexapod is capable of positioning a load of 200 kg.

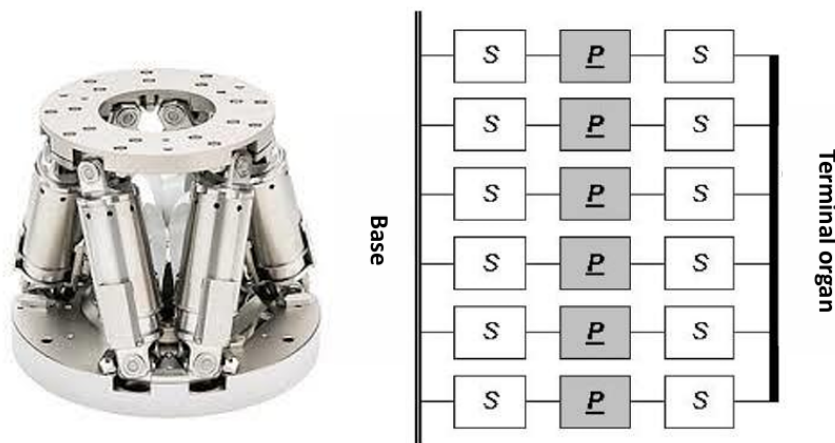


Figure 1.3: The Hexapod-System robot, photo and layout diagram [7]

Parallel robots are therefore considered as a solution to the limitations of serial robots. In this context, their advantages as well as their limitations and drawbacks are presented below [5].

1.3.3.2 Advantages

The parallel arrangement of several kinematic chains, each driven by an actuator, generally leads to the following advantages:

- High load capacity.
- Ability to achieve high dynamic motions (high accelerations).
- High mechanical rigidity.
- Low moving masses.
- High natural frequencies, resulting in reduced positioning errors caused by uncontrolled oscillations of the moving structure.
- Possibility of placing actuators directly on or near the fixed base, which leads to:
 - a wide choice of motors and gear reducers, since their mass has little influence on the system inertia.
 - simplified and more reliable connections between motors, sensors, and the controller.
 - easier cooling of actuators, reducing precision loss due to thermal expansion and allowing higher power levels.
 - easier isolation of motors from the workspace for clean-room environments, explosive atmospheres, or applications requiring intensive washing.
- ease of sensor integration.
- Modular mechanical design, which facilitates manufacturing and series production due to the presence of identical components.
- Limited influence of manufacturing tolerances on positioning accuracy.

note that this list of advantages is not necessarily verified for all parallel robot prototypes. In particular, early parallel machine tools suffered from precision and rigidity limitations. These limitations have motivated the development of improved parallel architectures and advanced control strategies in more recent designs. In this context, redundant parallel robots, such as Delta four arm manipulators, have been proposed to enhance fault tolerance, allow stiffness optimization, and improve load distribution and positioning accuracy.

1.3.3.3 Disadvantages

Compared to serial robots, the disadvantages of parallel robots can be summarized as follows:

- A limited workspace relative to the total volume of the mechanism. One solution consists in using linear actuators oriented in the same direction; increasing their stroke leads to an increased workspace. Another approach involves the use of rotary actuators arranged to cover a large common spatial range.

- Direct geometric models, which are sometimes difficult to determine. Iterative methods such as the Newton method and interval-based approaches (which guaranty the existence of solutions) are commonly used to address this difficulty.
- Strong coupling between the motions of the different kinematic chains, requiring highly coordinated actuator control. Although early electronic and computational limitations made such control difficult, modern control systems handle this issue without major computational problems. However, challenges remain, particularly in trajectory generation for parallel architecture machine tools.
- Strongly variable coupling between kinematic chains, which often complicates controller tuning. Over sizing actuators is one possible solution to mitigate this issue.
- The presence of singular configurations, which may lead to loss of control of the moving structure or even mechanical damage. This is one of the most critical issues in the design of parallel robots.

1.3.4 Two Generations of Parallel Robots

Numerous historical studies on parallel robots, such as those of Bonev and Merlet, describe their origins in detail [5]. Here, we simply emphasize the existence of two main generations of parallel robots:

- The first generation, represented by the Gough–Stewart platforms, also known as hexapods, in which six actuators connect the base to a mobile platform Figure. 1.4, enabling complex motions through the parallel arrangement of kinematic chains.
- The second generation, represented by the Delta structure proposed by Clavel. the robots in this family achieve exceptional performance, with velocities up to 10 m/s and accelerations reaching 20 g. These robots are characterized by:
 - actuators fixed on the base.
 - lightweight moving components Figure. 1.4.



Figure 1.4: The FlexPicker robot (ABB), photo and layout diagram

[8][9]

1.3.5 Hybrid Manipulator Robots (Serial–Parallel)

Hybrid manipulators Figure. 1.5 are robotic structures that combine serial and parallel connections within the same system. This type of architecture leads to a wide variety of manipulator designs. These robots are generally modular and can be extended by adding additional modules over large distances. Each module typically provides a hemispherical workspace, and the combination of several modules results in a larger and more symmetric overall workspace. Hybrid manipulators are known to exhibit a high force-to-weight ratio, exceeding that of conventional industrial and research manipulators. This performance is mainly due to their lightweight structure combined with high rigidity, which results from the predominantly axial distribution of forces within the mechanism. Actuation in hybrid manipulators is commonly achieved using prismatic actuators, such as ball-screw mechanisms with anti-backlash nuts or linear induction motors, which allow high positioning precision and precision [10].



Figure 1.5: YuMi .Abb’s Dual -arm robot
[11]

1.4 mobile robot

1.4.1 Definition

A Mobile Robot is an automated machine capable of moving within its environment independently or under remote control, using propulsion systems such as wheels, legs, or tracks. Unlike stationary industrial robots, mobile robots are not fixed to a single location, allowing them .

to perform diverse tasks in dynamic environments like warehouses, hospitals, and homes.

1.4.2 Key Categories (classifications)

Classification of Mobile Robots by Locomotion Mode Mobile robots are primarily categorized by their locomotion systems, which dictate their stability, maneuverability, and suitability for specific operational environments. While traditional taxonomies focus on single-mode systems, recent literature increasingly emphasizes hybrid and multimodal designs for complex terrain adaptation .

1.4.2.1 Wheeled Robots

Wheeled robots are the most prevalent class due to their mechanical simplicity and energy efficiency on prepared surfaces. They are typically classified by their wheel configuration and steering geometry, such as differential drive, Ackermann steering, or omnidirectional systems[12] [13]. While highly efficient, their mobility is limited to relatively smooth and continuous terrains.

1.4.2.2 Legged Robots

Legged platforms (bipedal, quadrupedal, or hexapedal) are designed to navigate irregular and unstructured terrains where wheeled systems fail [14]. These robots utilize discrete footholds, allowing them to step over obstacles. However, they require complex control systems to maintain dynamic stability and manage high-degree-of-freedom joint coordination [14][15].

1.4.2.3 Tracked Robots

Tracked or treaded robots use continuous tracks to distribute weight over a large surface area, providing superior traction and stability on soft or loose ground (e.g., sand, mud, or snow). They are frequently employed in search-and-rescue and military applications due to their robustness in extreme outdoor environments [16].

1.4.2.4 Aerial Robots (UAVs)

Operating in three-dimensional airspace, aerial robots are primarily categorized into Unmanned Aerial Vehicles (UAVs), including fixed-wing and rotary-wing platforms like quadcopters. Unlike terrestrial robots, these systems offer unconstrained high-speed mobility and rapid deployment capabilities. They are essential for surveillance, mapping, and agricultural inspection, representing a major domain of mobile robotics [17].

1.4.2.5 Underwater Robots (AUVs and ROVs)

Operating in GPS-denied and high-pressure environments, underwater mobile robots are categorized into Autonomous Underwater Vehicles (AUVs) and Remotely Operated

Vehicles (ROVs). These platforms use thrusters for propulsion and are essential for deep-sea exploration, infrastructure inspection, and marine biological research [17].

1.4.2.6 Hybrid and Multimodal Robots

A significant trend in recent research is the development of hybrid systems that combine multiple locomotion modes to overcome the limitations of single-mode platforms [16][12].

Wheeled-Legged Hybrids: These robots utilize wheels on the ends of legs to combine the speed/efficiency of wheeled movement with the terrain-clearing capabilities of legged systems [15][12].

Aerial-Terrestrial Hybrids: Systems capable of both flight and ground movement, often through morphological adaptation or specialized transitions [16].

1.5 Actuators used in robotics

An actuator is a device that converts the energy supplied to it into a usable physical effect. This physical effect produces work that modifies the behavior or the state of a machine, also referred to in automation as the operative part. There are four types of actuators in robotics:

1.5.1 Pneumatic

Pneumatic actuators operate by converting energy stored in a compressed gas into mechanical energy. The compressed gas is used as a means of energy transmission and storage, enabling the generation of motion within the system.



Figure 1.6: IMI Norgren ISOline tie rod double acting cylinders [18].

1.5.2 Hydraulic actuators

Hydraulic actuators consist of a cylinder–piston assembly that converts hydraulic energy into mechanical motion, which may be linear, rotary, or oscillatory. They are mainly employed in systems requiring high force and robustness.

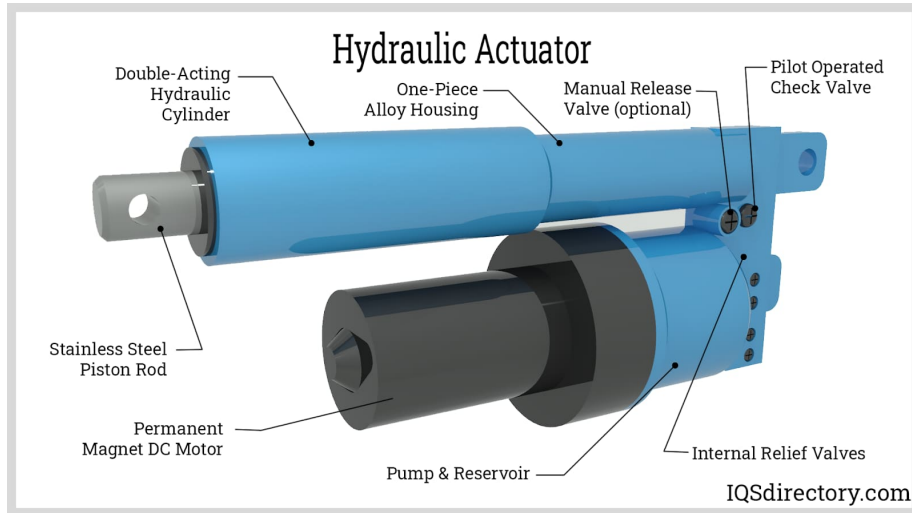


Figure 1.7: hydraulic actuator valve
[19]

1.5.3 Electrical actuators

refers to all energy transferred or stored using electricity. This energy is transferred from one system to another by the movement of charges.

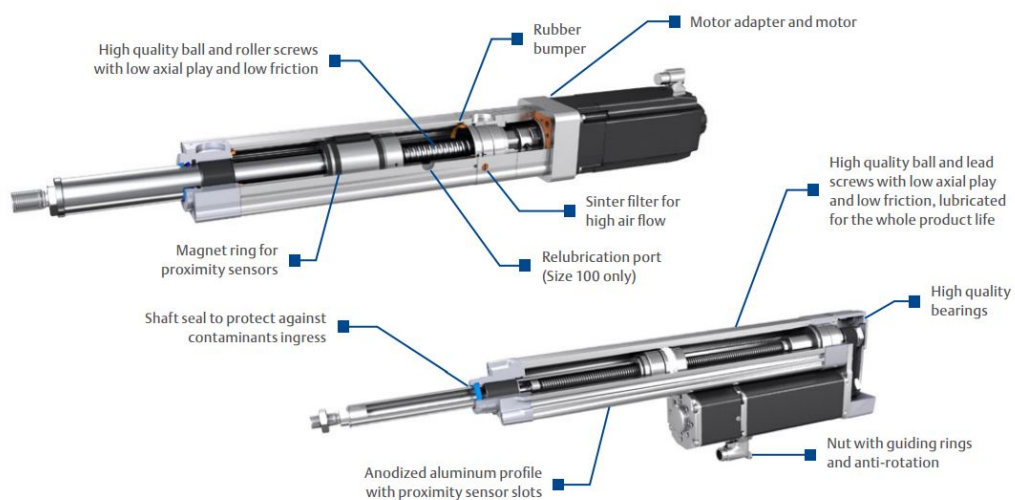


Figure 1.8: Electrical actuators
[20]

1.5.4 Mechanical actuators

providing kinetic or potential mechanical energy, it is a conserved quantity when no external forces or non-conservative forces (such as friction or impact) act on the system and is therefore practical to use.

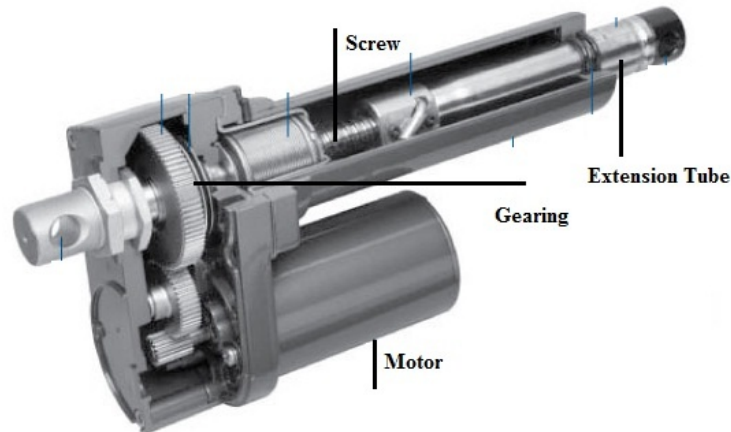


Figure 1.9: mechanical actuators
[\[21\]](#)

1.6 Characteristics of Robots

1.6.1 Precision

The precision of a robot is defined as the deviation between a commanded position and the actual position reached by the end-effector, in terms of both position and orientation. This deviation is expressed with respect to a reference coordinate frame and is generally characterized by absolute accuracy and repeatability[\[5\]](#).

1.6.2 Repeatability

Repeatability, also referred to as relative accuracy, describes the ability of a robot to return to a previously attained position under identical operating conditions. It is quantified by the dispersion of the end-effector position when the same target is reached multiple times. Repeatability is mainly influenced by mechanical clearances, friction, electrical noise, encoder resolution, and other stochastic disturbances within the robotic system [\[5\]](#).

1.6.3 Absolute Accuracy

Absolute accuracy represents the tolerance within which a robot can place its end-effector relative to a desired position in the workspace. It depends not only on mechanical quality but also on the accuracy of the geometric and kinematic models used by the controller.

It is affected by environmental disturbances, modeling errors, joint backlash, friction, structural deflections, control-loop limitations, and uncertainties between reference frames[5].

1.6.4 Resolution

The resolution of a robot refers to the smallest incremental motion that can be achieved by the It is primarily determined by the actuator characteristics and the resolution of the position sensors[5].

However, the effective resolution at the end-effector level may be influenced by friction, mechanical play, elastic deformation, and environmental factors.

1.6.5 Relationship Between Precision, Repeatability, and Resolution

Figure illustrates the relationship between commanded position, achieved position, absolute accuracy, repeatability, and resolution. Although these performance metrics are closely related, they describe distinct aspects of robot behavior and must be evaluated separately. For clarity in the subsequent analysis, the sources of absolute and relative positioning errors are commonly grouped into several categories.

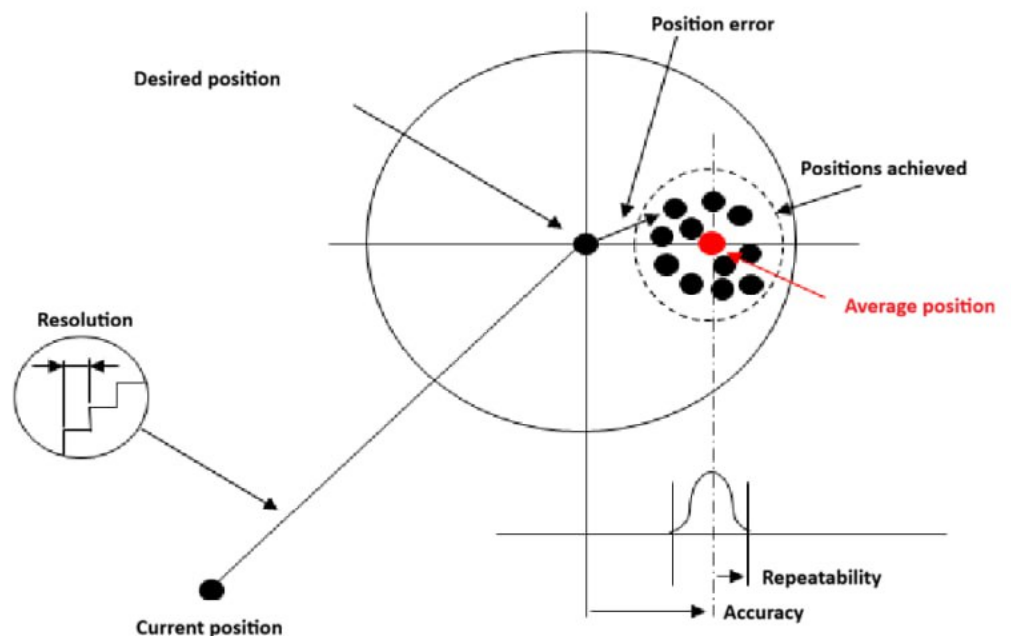


Figure 1.10: Accuracy, resolution, and repeatability (according to Niaritsiry [Niaritsiry, 2006])

[22][8]

- **Geometric Errors**

Geometric errors arise from imperfections in the robot's mechanical structure and inaccuracies in the geometric parameters used in the kinematic model. These include errors in link lengths, joint offsets, joint axis misalignment, and assembly tolerances. Such errors are systematic and repeatable, and they represent the primary source of absolute positioning errors in robotic manipulators.

- **Kinematic and Modeling Errors**

Kinematic and modeling errors are caused by simplifications and assumptions made during the development of the robot's mathematical model. These errors include neglected joint flexibility, idealized joint behavior, and approximations in forward and inverse kinematics. In parallel robots, such as the Delta robot, modeling inaccuracies can significantly affect the estimation of the end-effector position due to the closedloop kinematic structure.

- **Control and Actuation Errors**

Control and actuation errors originate from limitations in the control system and actuator performance. These include servo tracking errors, actuator saturation, encoder quantization, backlash, and friction effects. These errors mainly affect the repeatability of the robot rather than its absolute accuracy.

- **External Disturbances and Structural Deformations**

External disturbances and structural deformations are caused by environmental influences such as temperature variations, vibrations, payload changes, and elastic deformations of the robot structure. In high-speed parallel manipulators, elastic deformations of links and joints can lead to noticeable deviations in end-effector positioning, particularly under dynamic operating conditions.

1.7 The Delta Robots

A Delta robot is a type of parallel robotic manipulator characterized by a fixed base and a mobile platform. The Delta robot is a parallel manipulator composed of a fixed base and a mobile platform connected by multiple identical kinematic chains.

The actuators, typically rotary electric motors, are mounted on the fixed base and transmit motion to the end-effector through lightweight arms arranged in a parallel architecture.

This configuration allows the robot to achieve high speeds, high accelerations, and good positioning accuracy, particularly for translational motions of the end-effector. The Delta robot was first introduced in 1988 by Reymond Clavel at the Swiss Federal Institute of Technology (EPFL).

It was originally developed to meet industrial demands for high-speed pick-and-place applications, where low moving mass and high dynamic performance are essential.

The parallel structure of the Delta robot enables efficient force distribution and is well suited for tasks that require rapid and repetitive motion [23].

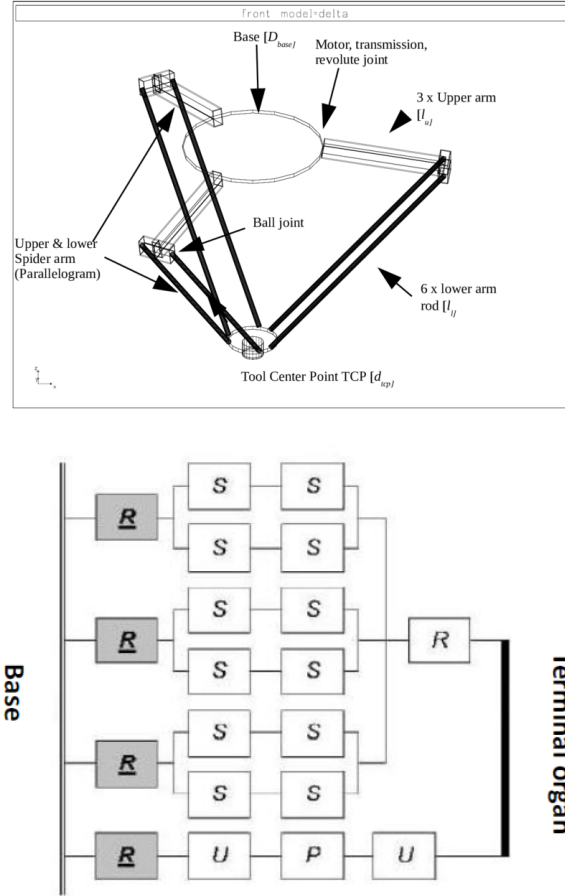


Figure 1.11: The DELTA robot and its main dimensions [22].

1.7.1 The Kinematic Diagram

Kinematic diagrams are commonly used to describe the structure of robotic mechanisms by representing links and joints and the relative motion between them.

These diagrams provide a simplified view of the mechanical architecture, allowing an initial understanding of how motion is transmitted from the actuators to the end-effector.

For simple planar mechanisms, kinematic diagrams are easy to interpret.

However, for spatial mechanisms with multiple closed kinematic chains, such as parallel robots, the diagram may rapidly become complex and difficult to read. For this reason, standardized representations of mechanical joints are often used to describe robot architectures in a clear and systematic manner [5].

Figure 1.12: Standardized Joint Representations

Joint Type	Representation	Relative movements.		2D orthogonal projection (iso)	3D perspective
		R	T		
pivot		Rx	0		
		0	0		
		0	0		
Slide		0	Tx		
		0	0		
		0	0		
Sliding pivot.		0	0		
		0	0		
		Rz	Tx		
spherical joint.		Rx	0		
		Ry	0		
		Rz	0		
Flat support		0	Tx		
		Ry	0		
		0	Tz		

[24].

1.7.2 The layout graph

The conventions for the layout graphs are presented in Tableau. 1.13.

Figure 1.13: Enter Caption

Joint Type	Representation	
	Passive Joint Symbol	Actuated Joint Symbol
Revolute (R)		
Revolute (R)		
Universal (U)		—
Spherical (S)		—

[22][24]

While kinematic diagrams and joint representations provide useful insight into the structure of a mechanism, they quickly lose clarity when applied to complex spatial robots with multiple closed kinematic chains.

In such cases, geometric details obscure the underlying structure of the mechanism, making analysis difficult.

1.7.3 Singular Configurations of Delta Parallel Robots

In robotic manipulators, singular configurations correspond to particular positions where the relationship between actuator motion and end-effector motion becomes ill-conditioned.

For serial manipulators, singularities are usually associated with a loss of degrees of freedom, whereas for parallel manipulators, such as the Delta robot, singularities result from a gain of one or more uncontrollable degrees of freedom. Passing through a singular-configuration may cause loss of control, excessive internal forces, or instability of the end effector.

For the Delta robot, singularities are commonly analyzed by studying the relative arrangement of the parallelogram links connecting the base to the moving platform. Each link is assumed to transmit a force along the line joining its two spherical joints. The platform is fully constrained and stable only if the force lines are properly distributed in space[25].

• Force-Line Interpretation

In this work, each bar is assumed to transmit only a force whose line of action coincides with the straight line connecting its two end joints. Under this assumption, the pose of the mobile platform is fully defined and statically constrained if the following conditions are satisfied [5].

- The lines of action of the forces are distributed in non-parallel and non-coplanar planes.
- No more than two force lines are parallel.

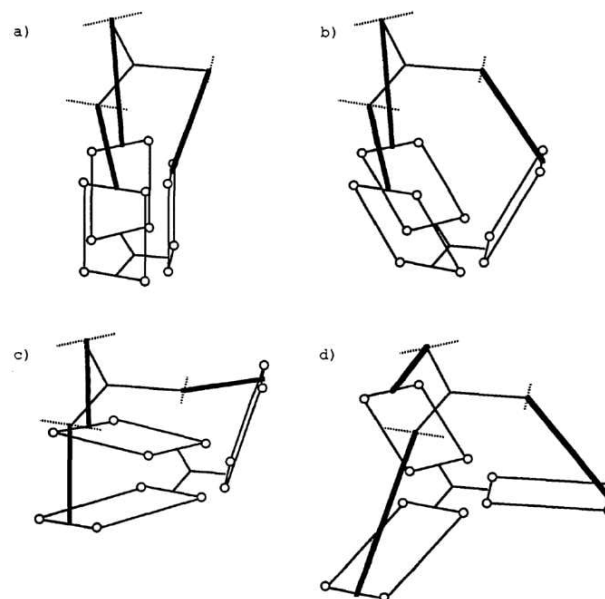


Figure 1.14: DNC ISO 6431 Pneumatic Standard cylinder [25].

• Representative Singular Configurations

The figure illustrates representative singular configurations of a Delta-type parallel robot, obtained from specific relative alignments of the kinematic chains.

- a) Fully parallel force lines: When all force transmission lines become parallel to a common direction, the mobile platform gains additional degrees of freedom. In this configuration, the platform may undergo combined translational and rotational motion with reduced constraint, resulting in a severe loss of stiffness.
- b) Partial parallelism of chains: When a subset of the force lines becomes parallel, the platform retains one uncontrollable degree of freedom, typically corresponding to motion along a curved trajectory.
- c) Coplanar force transmission lines: If several force lines lie in the same plane or in parallel planes, the platform may rotate around an axis defined by the remaining constrained joints, leading to a singular rotational mode.
- d) Parallel arrangement of kinematic planes: When the kinematic chains are arranged such that their motion planes become parallel or coincident, the platform may experience multiple uncontrolled motions, including rotations within the platform plane and small translational displacements normal to it.

1.8 Examples of Realizations and Performances of Delta Robots

Delta robots have been widely studied and implemented in both research laboratories and industrial environments due to their high speed, high precision, and favorable dynamic characteristics. Several industrial prototypes and systems illustrate the evolution and performance limits of this parallel architecture.

• milliDelta Robot Micro-Scale High-Bandwidth System

The milliDelta, developed by McClintock et al, at Harvard University, represents the state-of-the-art in miniaturized parallel manipulators, measuring only $15\text{ mm} \times 15\text{ mm} \times 20\text{ mm}$ with a total mass of 430 mg.

Using PCB-MEMS (Printed Circuit Board Micro-Electro-Mechanical Systems) construction combined with three piezoelectric bending actuators, the robot achieves exceptional dynamic performance, including velocities up to 0.45 m/s, accelerations reaching 215 m/s^2 , and positioning precision down to $5\text{ }\mu\text{m}$ within a 7.01 mm^3 workspace.

The system demonstrates a bandwidth of 75 Hz, enabling it to follow high-frequency periodic trajectories suitable for applications requiring rapid response and fine motion control.

The milliDelta’s payload capacity of 1.31 g makes it ideal for vibration cancelation in microsurgical procedures, precision microassembly tasks, and high-bandwidth micromanipulation operations where both compactness and micrometer-level accuracy are critical requirements.

This pioneering work demonstrates that the fundamental advantages of delta architecture—high speed, high precision, and parallel actuation—can be preserved even at the microscale, opening new possibilities for minimally invasive medical interventions and micro-manufacturing processes [26].

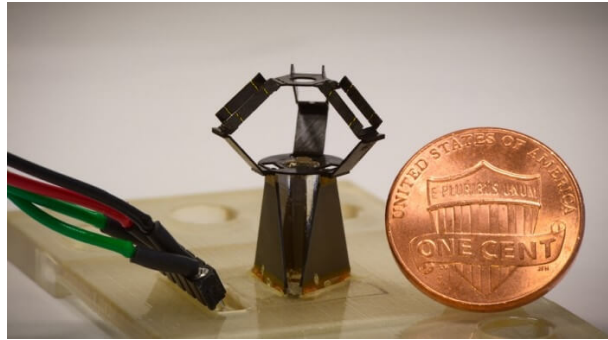


Figure 1.15: Harvard’s milliDelta Robot

1.9 ROMI Linear Delta - Ultra-Precision Platform

ROMI Linear Delta - Ultra-Precision Platform the ROMI Linear delta employs linear actuators in a Delta configuration, eliminating rotary-to-linear conversion errors and achieving exceptional motion precision of $3.3 \pm 0.3 \mu\text{m}$ motion precision.

This linear geometry reduces mechanical complexity, eliminates gear backlash, and provides predictable kinematics ideal for ultra-precision tasks.

The system excels in microsurgery, semiconductor wafer inspection, and optical alignment, where micrometer-level accuracy is essential.

Its demonstrated performance validates linear Delta architecture as superior for applications prioritizing precision over workspace volume or payload capacity[27].



Figure 1.16: ROMI Linear Delta

1.10 3D-Printed Open-Source Delta - Educational Platform

The open-source delta robot, developed by Ayazbay et al, demonstrates the democratization of parallel robotics technology through additive manufacturing and readily available components, including arduino microcontrollers and standard servo motors.

Fully 3D-printed structural components and low-cost electronics enable a complete system cost less than \$500, making sophisticated manipulation capabilities accessible to educational institutions, hobbyists, and small-scale manufacturing operations.

Performance specifications include a payload capacity of up to 200 g, a maximum velocity of 0.4 m/s, a throughput of approximately 30 picks per minute, and positioning accuracy of 2.2 mm—adequate for educational demonstrations, prototype development, and low-precision pick-and-place operations.

The trajectory planning and kinematics have been experimentally validated through systematic testing, confirming that the printed structure maintains sufficient rigidity for repeatable manipulation tasks despite the inherent compliance of thermoplastic materials.

The open-source design enables students and researchers to directly examine and modify parallel robot kinematics, making it a popular platform for robotics education, entrepreneurial ventures, and small-scale production where investment in expensive industrial equipment cannot be justified [28].



Figure 1.17: Neutron - Open Source 3D Printer [29].

1.11 DOF UAV-Mounted System

The H-Delta, developed by Hodgins, extends traditional three- degree-of freedom delta architecture to five degrees of freedom by incorporating two additional rotational capabilities through a decoupled wrist mechanism that enables orientation control independent of translational positioning.

This configuration addresses a fundamental limitation of standard delta robots, which can position their end-effector in 3D space but cannot control its orientation, restricting their utility for manipulation tasks requiring specific approach angles or tool orientations.

The prototype achieves translational positioning accuracy within approximately 3 mm and rotational accuracy within 0.5° , with the decoupled architecture ensuring that rotational commands do not induce translational motion and vice versa, simplifying control algorithms, and improving stability when the base platform is moving or subject to external disturbances.

What distinguishes the H-Delta most significantly is its design optimization for mounting on unmanned aerial vehicles (UAVs), where weight, compactness, and decoupled control are critical system requirements to maintain stability during flight operations.

Prototype demonstrations have validated the capability of the H-Delta for UAV-mounted object retrieval and manipulation missions, showcasing potential applications in search-and-rescue operations, infrastructure inspection, power line maintenance, and aerial construction tasks where traditional ground-based manipulation is impractical or dangerous [30].



Figure 1.18: UAV frame with mounted drivers, companion computer and payload [31].

1.12 Five-DOF Delta-Like Robot - Industrial Packaging System

This delta-like five-degree-of-freedom delta-like robot, documented in the Journal of Mechanisms and Robotics, addresses the critical limitations of traditional robot UPU (Universal-Prismatic-Universal) wrist mechanisms commonly used to add rotational capabilities to standard delta robots, specifically targeting the reduced vertical workspace and joint backlash that compromise accuracy in industrial packaging applications.

The parallelogram-based wrist design overcomes these limitations through a successive-chain architecture that integrates seamlessly with the delta's three primary legs, providing enhanced rotational capabilities while maintaining the delta's characteristic speed and precision advantages.

The design was fully optimized through combined kinematic analysis and dynamic

simulation to balance competing objectives of workspace volume, rotational range, structural stiffness, and actuator loads, resulting in approximately 40% greater vertical workspace compared to equivalent UPU designs and reduction of joint backlash to less than 0.05 mm (a 90% improvement over traditional universal joints).

A physical prototype was constructed and experimentally evaluated, demonstrating performance improvements suitable for demanding packaging operations that require precise box orientation, label alignment, and stacking tasks.

In industrial packaging environments, these improvements translate directly into economic benefits: reduced backlash improves quality control and reduces rejection rates, improved vertical reach allows taller stacks and denser workspace utilization, and maintained speed ensures that throughput requirements are met while achieving superior accuracy compared to traditional 5-DOF delta implementations [30].



Figure 1.19: Industrial Packaging System

1.13 DeltaFlex - Additively Manufactured Compliant-Joint System

DeltaFlex, developed by Parmiggiani et al, represents an innovative approach to delta robot fabrication that leverages additive manufacturing (AM) technology and compliant joint design to create monolithic robotic structures that eliminate traditional multi-component assembly while enabling rapid design iteration and application-specific customization.

The principles of Design for Additive Manufacturing (DfAM) guided the development process, with the Finite Element Method (FEM) simulation used extensively to optimize the joint geometry, predict compliance characteristics, and verify that stress concentrations remained within safe limits, with experimental validation demonstrating that the predictions of FEM matched the actual measured stiffness and repeatability within 95% accuracy.

The primary trade-off in compliant joint design involves balancing joint flexibility (necessary for motion) against structural stiffness (necessary for load bearing and precision).

Increased compliance expands the achievable workspace but reduces payload capacity, and positioning accuracy, while reduced compliance improves stiffness but limits range of motion.

This demonstrated viability validates additive manufacturing as a rapid prototyping and possibly even production-scale fabrication method for parallel robots, particularly valuable for research applications that require quick design iterations, specialized end-effector geometries, or customized compliance characteristics tailored to specific manipulation tasks [32].

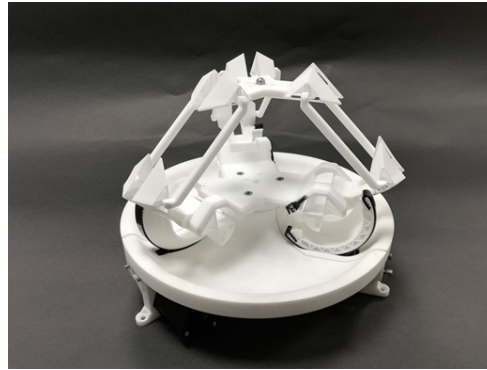


Figure 1.20: A parametric model for tetra joints applied [33].

1.14 Motivation

Delta robots are widely used in high-speed and high-precision applications due to their lightweight structure and parallel kinematic architecture.

However, classical three-arm Delta robots present certain limitations in terms of stiffness, load distribution, and workspace constraints, especially in tasks that require higher stability or additional functionalities.

The introduction of a fourth arm aims to improve structural rigidity, improve positioning accuracy, and increase operational reliability. In addition, integration of a vision system allows real-time perception of the workspace, enabling intelligent control, error correction, and flexible task execution.

This work is therefore motivated by the need to explore an extended Delta robot architecture that combines mechanical enhancement with sensing capabilities, making it suitable for advanced industrial and research applications.

1.15 Conclusion

This chapter introduced the fundamentals of industrial robotic systems, with a focus on parallel manipulators and Delta robots. The structure, main characteristics, and application areas of Delta robots were briefly presented. The limitations of conventional three-arm Delta architectures motivated the study of a four-arm configuration. This provides the basis for the modeling and control work developed in the following chapters.

Chapter 2 | Mathematical model of the parallel robot

2.1 Introduction

Mathematical modeling is a vital step in the design and control of parallel robots. this chapter focuses on analyzing the behavior of the 4-arm Delta robot from two main perspectives : kinematic modeling—including forward kinematics, inverse kinematics, and singularity analysis—and dynamics modeling. Furthermore, the robot’s workspace will be assessed using MATLAB. these models provide the essential foundation for the control strategies and electrical design covered in the following chapters.

2.2 System Description

The proposed system is built around a Delta robot that features a parallel architecture with four identical kinematic chains. This design uses a parallelogram-based structure to ensure that the moving platform maintains a fixed orientation throughout its motion. Each chain includes an actuated revolute joint connected to a rod system through passive spherical joints, which connect the fixed base to the moving platform.

The four actuators are symmetrically positioned at 90° intervals on a square base, allowing three translational degrees of freedom (3-DoF) along the x, y, and z axes.

This configuration provides high stiffness and a lightweight design, as the links experience only tension and compression forces, resulting in precise positioning of the end-effector.

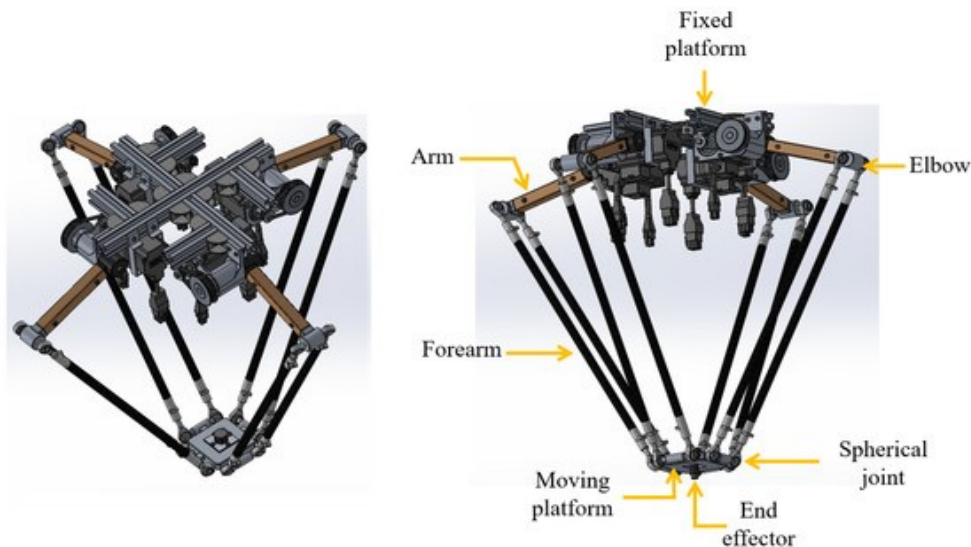


Figure 2.1: UAV frame with mounted drivers, companion computer and payload

2.3 Delta robot design

The mechanical design of the investigated parallel robot is based on a kinematic structure consisting of four active revolute joints. Each joint possesses a single degree of freedom, with its axis of rotation aligned with the Z-axis of the base.

Motion is transmitted from the main arms to the moving platform via two-bar mechanisms employing spherical joints with three rotational degrees of freedom, enabling a highly flexible kinematic response.

To analyze the kinematic and dynamic behavior of the system, modeling was performed within the Simscape Multibody environment.

This model incorporates the physical and geometric properties of each component—including masses, centers of mass, and moments of inertia—based on an assumed material density of 1000 kg/m^3 . As shown in Figure. 2.2.

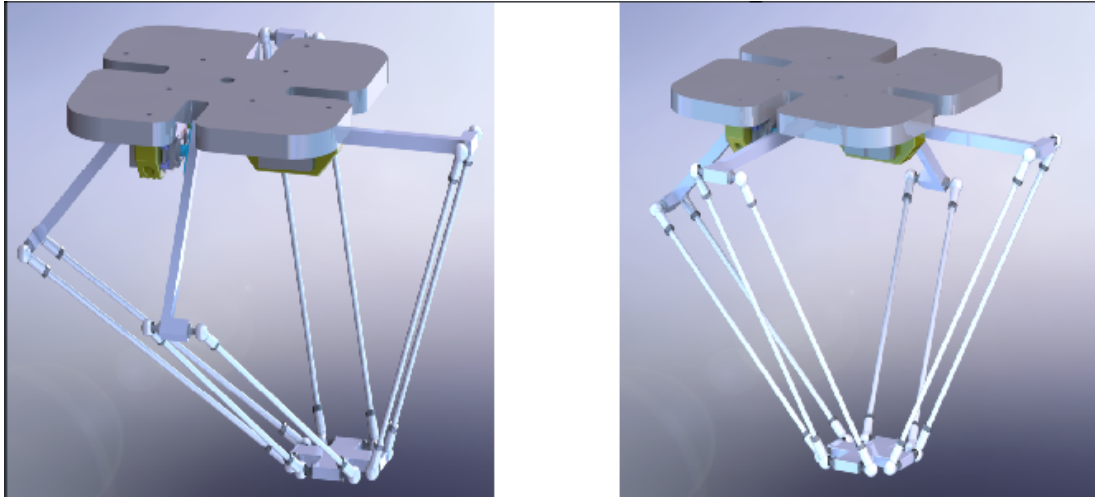


Figure 2.2: 3D Model illustrating the mechanical design of the parallel Delta robot

Furthermore, to ensure operational stability and prevent mechanical collisions or the robot falling into singularities, joint limits were imposed on the actuated joints, restricting their motion to an angular range of -25° to 100° . The dimensions of each component are listed in Tableau. 2.1

These characteristics, combined with the 50 W motor power considered for this application, enable an effective payload capacity of up to 1 kg.

This load capacity is sufficient to handle the weight of a fast food order with its toppings, which is approximately 480 g.

The robot's calibration position is defined by setting the arms at 0° relative to the axis of their respective local coordinate X systems, as shown in Figure. 2.3.

Table 2.1: Delta robot design parameters.

Parameter	Value (mm)
Fixed upper base	
Length	300
Width	300
Height	20
Joints	
Radius	20
Arms	
Length	160
Width	15
Height	15
Elbow	
Radius	10
Length	50
Forearms	
Radius	5
Length	326
Moving lower base	
Length	45.96
Width	45.96
Height	11.86

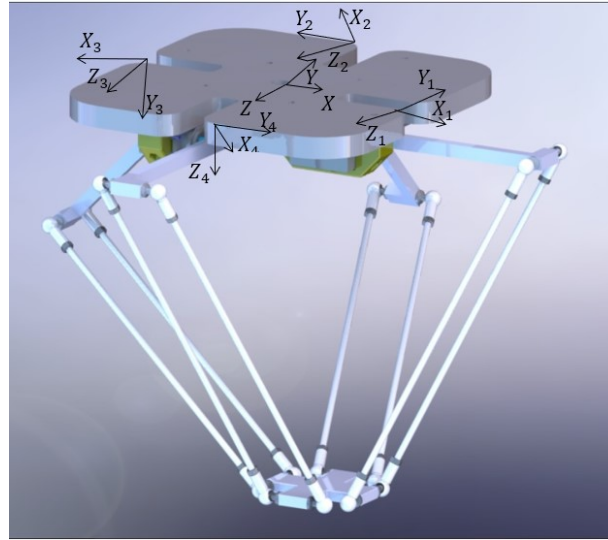


Figure 2.3: Side view of the robot. Coordinate system assignment.

2.4 Mathematical models of the Delta robot

To carry out specific tasks with this robot, the position and orientation of its end effector relative to the base must be defined, which requires a kinematic model for accurate representation.

In Addition, to develop control strategies that ensure optimal performance of the manipulator, a dynamic model is necessary to capture and predict its behavior over time. The mathematical modeling framework presented in this chapter is based on the foundations established by Urrea et al. [34]. However, during the implementation and validation of these models for this study, several analytical refinements and corrections were applied to

the original derivations to ensure consistency in the 4-arm Delta robot's kinematic and dynamic representations.

2.4.1 Geometric model (MGD)

To obtain the geometric model, a homogeneous coordinate system was employed to represent positions and orientations in space. The origin of this coordinate system coincides with the geometric center of the fixed base of the robot.

It is assumed that arm lengths remain constant throughout motion and that the robot does not enter any singular configurations during normal operation. The Z coordinate is defined as positive downward, aligned with the direction of gravitational force, which has a value of 9.81 m/s^2 .

In addition, joints are considered rigid, with negligible bending or deformation during movement. Based on these assumptions, the assignment of coordinate axes for each arm and the fixed base is illustrated in Tableau. 2.1.

The length of the i th forearm corresponds to the vector norm vector connecting the two anchor points A_i and B_i , as shown in Tableau. 2.4.

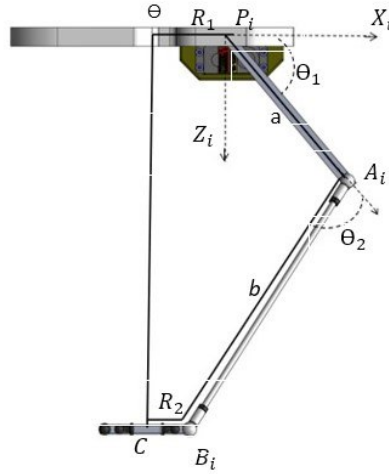


Figure 2.4: Side view of an arm

These kinematic relationships are expressed in Equation (2.1).

$$\|\overrightarrow{A_i B_i}\|^2 = b_i^2 \quad (2.1)$$

where:

i : 1,...,4

A_i : Center of ball joints in the forearm i .

B_i ,: Center of the ball joint relative to forearm i on the moving platform.

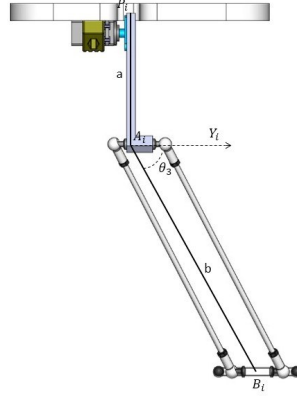


Figure 2.5: Frontal view of an arm.

Figure. 2.4 shows the front view of one of the arms and its relationship with the motion On Y the coordinate axis. The geometric model of the Delta robot is shown below, where the position of point C represents the location of the end-effector.

The relationship between the origin O and the point C is expressed geometrically using Equation (2.2).

$$\overrightarrow{OC} = \overrightarrow{OP_i} + \overrightarrow{P_iA_i} + \overrightarrow{A_iB_i} - \overrightarrow{B_iC} \quad (2.2)$$

Each of the terms in(2.2) can be expressed in a matrix form, as shown in(2.2) in Equation(2.3)–(2.7).To simplify the nomenclature,it is defined:

$$c\theta_i = \cos \theta_i \text{ and } s\theta_i = \sin \theta_i.$$

$$\overrightarrow{OP_i} = \begin{bmatrix} R_1 \\ 0 \\ 0 \end{bmatrix} \quad (2.3)$$

$$\overrightarrow{P_iA_i} = \begin{bmatrix} ac\theta_1 \\ 0 \\ as\theta_1 \end{bmatrix} \quad (2.4)$$

$$\overrightarrow{A_iB_i} = \begin{bmatrix} bc(\theta_1 + \theta_2)s\theta_3 \\ bc\theta_3 \\ bs(\theta_1 + \theta_2)s\theta_3 \end{bmatrix} \quad (2.5)$$

$$\overrightarrow{B_iC} = \begin{bmatrix} -R_2 \\ 0 \\ 0 \end{bmatrix} \quad (2.6)$$

$$\overrightarrow{CO} = \begin{bmatrix} c\phi_i & s\phi_i & 0 \\ -s\phi_i & c\phi_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = \begin{bmatrix} p_x c\phi_i + p_y s\phi_i \\ -p_x s\phi_i + p_y c\phi_i \\ P_z \end{bmatrix} \quad (2.7)$$

The parameter values of these equations correspond to those presented in Tableau. 2.2.

Table 2.2: Delta robot parameters.

Parameter	Value
R_1	100 mm
a	160 mm
b	326 mm
R_2	45.96 mm
ϕ_1	0°
ϕ_2	90°
ϕ_3	180°
ϕ_4	270°
$\theta_{1_i} \ (i = 1, \dots, 4)$	$-25^\circ \leq \theta_{1_i} \leq 100^\circ$

2.4.2 Inverse kinematics (IK)

To obtain the inverse kinematics of the robot, an angle ϕ was generated between the X axis of the fixed base and the X axis of each arm, which being $\phi_1 = 0^\circ$, $\phi_2 = 90^\circ$, $\phi_3 = 180^\circ$, $\phi_4 = 270^\circ$.

Because the four arms of the robot are designed to be exactly the same, finding the value of each actuated joint ($\theta_{1_i}, i = 1 - 4$) it is sufficient to repeat the procedure described below but modify the value of ϕ_i according to the angle of each arm with respect to the X axis of the fixed base. From the matrix terms, Equation (2.2) can be rewritten as a system of three equations, as shown in (2.8)–(2.10):

$$R_1 + ac\theta_1 + bc(\theta_1 + \theta_2)s\theta_3 = R_2 + C_x c\phi_i + C_y s\phi_i$$

yields:

$$R_1 + ac\theta_1 + bc(\theta_1 + \theta_2)s\theta_3 = R_2 + C_x c\phi_i + C_y s\phi_i \quad (2.8)$$

$$bc\theta_3 = -C_x s\phi_i + C_y c\phi_i \quad (2.9)$$

$$as\theta_1 + bs(\theta_1 + \theta_2)s\theta_3 = C_z \quad (2.10)$$

Solving Equation(2.9) it is obtained θ_3 :

$$\theta_3 = \cos^{-1} \left(\frac{-C_x s\phi_i + C_y c\phi_i}{b} \right) \quad (2.11)$$

According to Figure(2.6) and considering the triangle PAB , $\overrightarrow{PB}$ can be defined using the cosine theorem, as presented in (2.12).

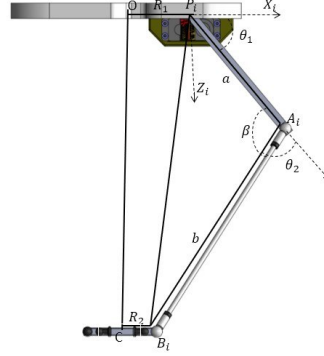


Figure 2.6: Frontal view of an arm.

$$\overrightarrow{PB}^2 = a^2 + b^2 - 2ab \cos(\beta) \quad (2.12)$$

employing trigonometric identities $\cos(\beta) = -\cos(\theta_2)\sin(\theta_3)$, therefore, (2.12) can be rewritten as shown in (2.13).

$$\overrightarrow{PB}^2 = a^2 + b^2 + 2abc\theta_2 s\theta_3 \quad (2.13)$$

According to the geometric description in Figure (2.6), $\overrightarrow{PB}$ can also be expressed as (2.14).

$$\overrightarrow{PB} = \begin{bmatrix} R_2 + C_x c\phi_i + C_y s\phi_i - R_1 \\ -C_x s\phi_i + C_y c\phi_i \\ C_z \end{bmatrix} \quad (2.14)$$

Furthermore, the magnitude of this vector can be expressed as follows :

$$\overrightarrow{PB}^2 = H^2 + I^2 + J^2 \quad (2.15)$$

where:

$$H = R_2 + C_x c\phi_i + C_y s\phi_i - R_1$$

$$I = -C_x s\phi_i + C_y c\phi_i$$

$$J = C_z$$

Equating (2.13) with (2.15) and solving θ_2 , we obtain the following.

$$\theta_2 = \cos^{-1} \left(\frac{H^2 + I^2 + J^2 - a^2 - b^2}{2abs\theta_3} \right) \quad (2.16)$$

To find the angle θ_1 Equation(2.10) is used but rewritten as presented in (2.17).

$$as\theta_1 + bs\theta_1c\theta_2s\theta_3 + bc\theta_1s\theta_2s\theta_3 = C_z \quad (2.17)$$

Therefore,we know that :

$$s\theta_1 = \frac{2 \tan(\frac{\theta_1}{2})}{1 + \tan^2(\frac{\theta_1}{2})} \quad (2.18)$$

$$c\theta_1 = \frac{1 - \tan^2(\frac{\theta_1}{2})}{1 + \tan^2(\frac{\theta_1}{2})} \quad (2.19)$$

Substituting (2.18) and (2.19) into (2.17), the following expression is obtained:

$$(-N - L) \tan^2\left(\frac{\theta_1}{2}\right) + 2M \tan\left(\frac{\theta_1}{2}\right) + (N - L) = 0 \quad (2.20)$$

where:

$$M = a + bc\theta_2s\theta_3$$

$$N = bs\theta_2s\theta_3$$

$$L = C_z$$

Finally, by solving (2.20) it is possible to find θ_1 from (2.21).

$$\theta_1 = 2 \tan^{-1} \left(\frac{-M \pm \sqrt{M^2 + N^2 - L^2}}{(L - N)} \right) \quad (2.21)$$

2.4.3 Forward kinematics (FK)

Obtaining the forward kinematics of this type of robot is typically much more complex than obtaining its inverse kinematics.

However, to calculate forward kinematics, it can be considered that the values of the angles θ_i ($i = 1 - 4$) are known,because of which the position of the end effector can be defined with respect to the coordinate system XYZ ; e.g., the point C composed of C_x, C_y , and C_z . If a point D_i ($i = 1 - 4$) is defined on each arm of the robot such that $A_iD_i = B_iC = R_2$, as shown in Figure. 2.7

four virtual spheres can be traced whose intersection is the point C . Each sphere has its center at D_i and a radius equal to b . Therefore, the point D_i can be defined as follows:

$$D_{x_i} = (R_1 + bc\theta_i - R_2)c\phi_i \quad (2.22)$$

$$D_{y_i} = (R_1 + bc\theta_i - R_2)s\phi_i \quad (2.23)$$

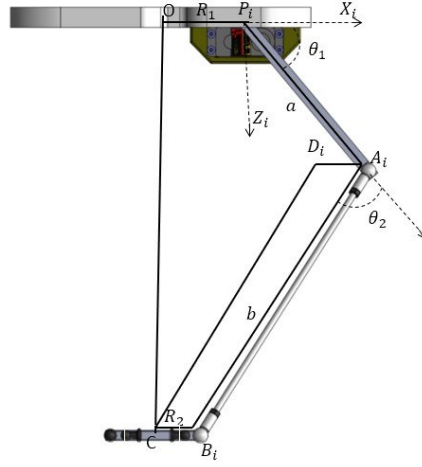


Figure 2.7: lateral view of an arm

$$D_{z_i} = -bs\theta_i \quad (2.24)$$

On the other hand, the equations of the spheres can be obtained as presented in (2.25):

$$(C_x - D_{x_i})^2 + (C_y - D_{y_i})^2 + (C_z - D_{z_i})^2 = b^2 \quad (i = 1 - 4) \quad (2.25)$$

As θ_i ($i = 1 - 4$) is known, the values of D_{x_i} , D_{y_i} and D_{z_i} are constant. Therefore, by defining the variable change in (2.26), it is possible to obtain a new simplified system of equations for the spheres, as presented in (2.27) and (2.28).

$$s_i = D_{x_i}^2 + D_{y_i}^2 + D_{z_i}^2 \quad (2.26)$$

$$C_x(D_{x_1} - D_{x_3}) + C_y(D_{y_1} - D_{y_3}) + C_z(D_{z_1} - D_{z_3}) = \frac{s_1 - s_3}{2} \quad (2.27)$$

$$C_x(D_{x_2} - D_{x_3}) + C_y(D_{y_2} - D_{y_3}) + C_z(D_{z_2} - D_{z_3}) = \frac{s_2 - s_3}{2} \quad (2.28)$$

By solving for C_z in the previous two equations and subtracting to find C_x as a function of C_y , is obtained:

$$C_x = C_y t + u \quad (2.29)$$

where:

$$\begin{aligned}
 t &= \frac{l-r}{q-n} & m &= \frac{\frac{s_1-s_3}{2}}{D_{z_1}-D_{z_3}} & p &= \frac{\frac{s_2-s_3}{2}}{D_{z_2}-D_{z_3}} \\
 u &= \frac{p-m}{q-n} & n &= \frac{D_{x_1}-D_{x_3}}{D_{z_1}-D_{z_3}} & q &= \frac{D_{x_2}-D_{x_3}}{D_{z_2}-D_{z_3}} \\
 & & l &= \frac{D_{y_1}-D_{y_3}}{D_{z_1}-D_{z_3}} & r &= \frac{D_{y_2}-D_{y_3}}{D_{z_2}-D_{z_3}}
 \end{aligned}$$

From (2.29) it is possible to express C_z as a function of C_y :

$$C_z = C_y t_a + u_a \quad (2.30)$$

where:

$$\begin{aligned}
 t_a &= \frac{(r-l)n-(q-n)l}{q-n} \\
 u_a &= \frac{(p-m)n-(q-n)m}{q-n}
 \end{aligned}$$

From the values of C_x and C_z it can be obtained C_y , as follows:

$$C_y = \frac{-e \pm \sqrt{e^2 - 4df}}{2d} \quad (2.31)$$

where:

$$\begin{aligned}
 d &= t^2 + t_a^2 + 1 \\
 e &= 2tu - 2tD_{x_4} - 2D_{y_4} + 2t_a u_a - 2D_{z_4} t_a \\
 f &= s_4 - b^2 - 2u_a D_{z_4} + u_a^2 + u^2 - 2D_{x_4} u
 \end{aligned}$$

2.4.4 Singularity Analysis

To evaluate the kinematic performance and identify potential singularity zones, a singularity analysis was performed using MATLAB. The Jacobian matrix (J), which relates the joint velocities to the end-effector velocities, was computed at every point within the reachable workspace. The Jacobian matrix (J), which relates the joint velocities to the end-effector velocities, was computed at every point within the reachable workspace. It is defined as:

$$\dot{\mathbf{p}} = J \dot{\boldsymbol{\theta}} \quad (2.32)$$

where J is derived from the kinematic chain geometry as:

$$J = \begin{bmatrix} b_1^T \\ b_2^T \\ b_3^T \\ b_4^T \end{bmatrix} \begin{bmatrix} (S_1 b_1)^{-1} & 0 & 0 & 0 \\ 0 & (S_2 b_2)^{-1} & 0 & 0 \\ 0 & 0 & (S_3 b_3)^{-1} & 0 \\ 0 & 0 & 0 & (S_4 b_4)^{-1} \end{bmatrix} \quad (2.33)$$

with $S_i = s_i \times a_i$, where s_i is the unit vector along forearm i and a_i is the unit vector along upper arm i , for $i = 1, \dots, 4$.

The condition number of the Jacobian, denoted as $\kappa(J) = |J| \cdot |J^{-1}|$, was used as the primary metric for dexterity.

A condition number close to 1 indicates optimal dexterity (isotropic configuration), while a high condition number signifies proximity to a singularity.

The results were visualized using a 3D point cloud, where each point is color-coded according to the base-10 logarithm of its condition number.

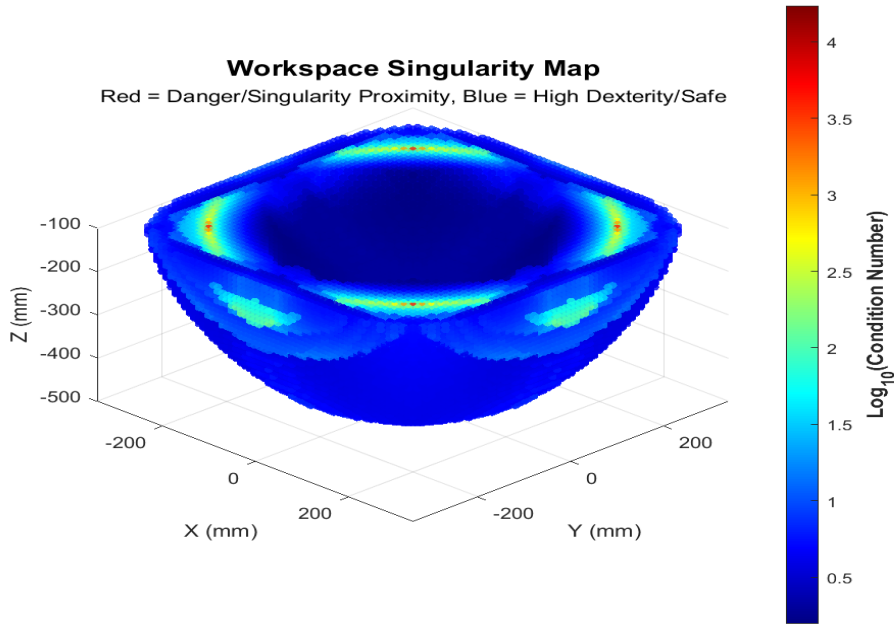


Figure 2.8: Singularity map based on the Jacobian condition number.

2.4.4.1 Discussion of Results

As shown in Figure. 2.8, the singularity map reveals a clear distribution of dexterity across the robot's workspace. The central region of the workspace, depicted in deep blue, exhibits a low condition number (typically $10^{0.5}$ to 10^1), indicating high dexterity and stable kinematic behavior. This "safe zone" occupies the majority of the working volume, which is crucial for precision tasks.

Conversely, the red "danger zones" are concentrated primarily at the boundaries of the workspace. These correspond to configurations where the arms are either fully extended

(reaching the workspace limit) or folded inward, leading to a loss of rank in the Jacobian matrix. Additionally, specific singular "pockets" are visible near the upper edges of the dome-like workspace. To ensure the reliability of the Delta robot, the operational trajectory should be constrained within the blue regions, avoiding the outer edges where the condition number rises sharply, indicating a rapid loss of controllability and mechanical stiffness.

2.5 Dynamic

2.5.1 Introduction

Dynamic analysis is essential for evaluating actuator requirements and ensuring safe operation of parallel robots. In this work, the dynamics of the 4-arm Delta robot are studied to estimate the torque required at each motor.

2.5.2 Dynamic Modeling

The dynamic model is derived using the principle of virtual work, which relates actuator torques to external forces acting on the robot.

Given the Average-speed pick-and-place application, the model is simplified by considering gravity as the dominant effect, while inertial and friction forces are neglected for static analysis. The model includes the masses of the arms, forearms, and moving platform, while the payload has a negligible contribution.

Under these conditions, the actuator torque can be approximated by:

$$\tau \approx \tau_{\text{arm}} + \tau_{\text{platform}} \quad (2.34)$$

with:

$$\tau_{\text{arm}} = m_{\text{arm}} g \frac{L_{\text{arm}}}{2} \cos(\theta) \quad (2.35)$$

$$\tau_{\text{platform}} = \frac{(m_{\text{platform}} + m_{\text{payload}} + 4 \cdot \frac{m_{\text{forearm}}}{2}) g}{4} L_{\text{arm}} \cos(\theta) \quad (2.36)$$

2.5.3 Numerical Results

The torque required to maintain the platform in the central position $[0, 0, -350]$ mm is:

Motor	Torque (Nm)
1	0.347
2	0.347
3	0.347
4	0.347

2.5.4 Discussion

The equal torque values confirm a symmetric load distribution at the center of the workspace. The results show that the required torque is relatively low and mainly due to the robot structure, while the payload effect is negligible.

The obtained torque remains well below the actuator limits, indicating safe operation. For higher speeds or heavier loads, additional dynamic effects should be considered.

2.5.5 Workspace Analysis

The workspace of a Delta parallel robot is the three-dimensional region that can be reached by the end-effector. Defining this region is essential to ensure that the robot operates within its mechanical limits.

In this section, the workspace is analyzed to understand its shape and the distribution of reachable positions.

The workspace was obtained using MATLAB by sampling the joint variables within their allowable ranges and applying the forward kinematics model.

This process generated a point cloud of approximately 99,723 reachable positions, allowing a clear visualization of the workspace boundaries and the distribution of reachables.

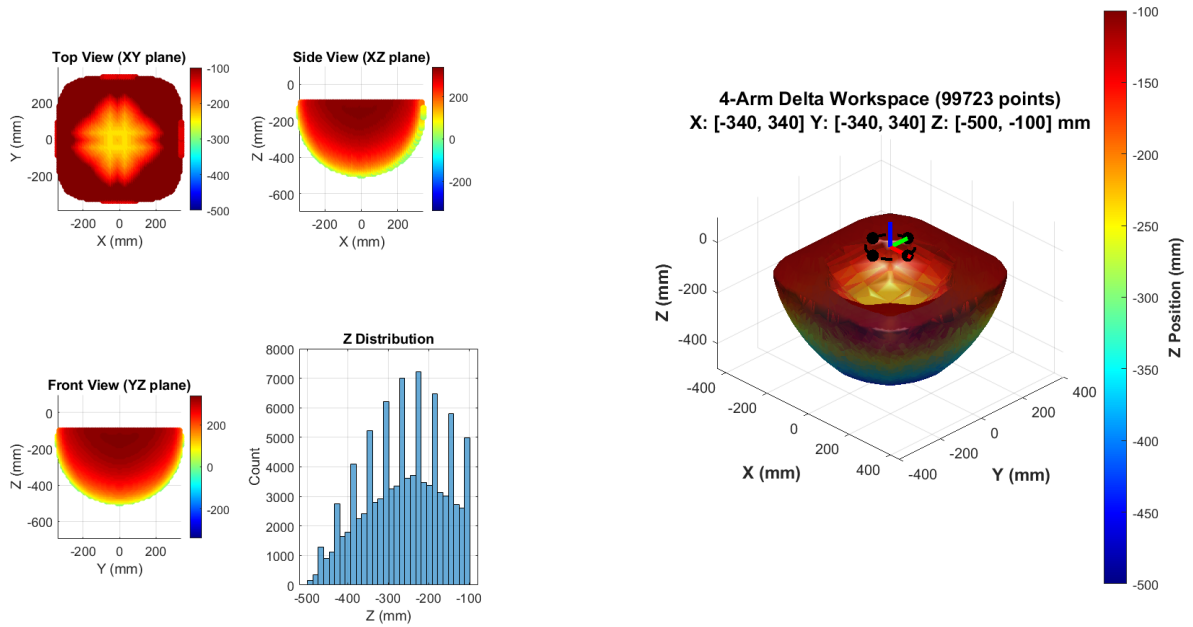


Figure 2.9: Multi-view analysis of the workspace including Top (XY), Side (XZ), and Front (YZ) views, alongside a Z-coordinate frequency distribution

2.5.6 Detailed Analysis and Results

As illustrated in Figure. 2.9, the workspace has a dome-shaped shape, distributed symmetrically around the Z-axis.

The top view (XY plane) shows a square-shaped rounded shape with a maximum reach of approximately ± 340 mm in both X and Y directions. The slight truncation at the corners is due to the kinematic constraints of the four-arm structure.

The robot operates within a vertical range from $Z = -100$ mm to $Z = -500$ mm. The side and front views (XZ and YZ planes) reveal a bowl-shaped profile, where the reachable area gradually decreases as the end-effector moves downward.

The distribution of points along the Z-axis indicates that most reachable positions are concentrated between -200 mm and -350 mm. This region corresponds to the highest flexibility and lateral reach, making it the most suitable zone for efficient operation.

2.6 Conclusion

This chapter presented the kinematic and dynamic analysis of the 4-arm Delta robot, including workspace evaluation, singularity analysis, and torque estimation. The results confirm that the robot operates within safe limits and maintains a stable and symmetric behavior in its workspace. These findings provide a solid foundation for the next chapter, which focuses on the hardware implementation, including the mechanical design and electronic components of the system.

Chapter 3 | Hardware realisation

3.1 Introduction

Following the theoretical study and mathematical modeling of the Delta robot presented in the previous chapter, this third chapter focuses on the practical implementation and hardware realization of the project. The main goal of this phase is to transform the virtual model into a tangible, functional physical system.

To accomplish this, we begin with a detailed overview of the mechanical design and material selection, followed by a description of the electronic architecture, including the selection of actuators and power supply systems. This chapter establishes the physical foundation necessary for the subsequent development and implementation of control algorithms and software programming.

3.2 Mechanical Parts

The robot's structure is built using lightweight and durable materials such as wood, steel, and aluminum to minimize the overall weight while ensuring that the parts remain securely assembled. This design approach considers that the development and control of robots necessitate the calculation of specific mathematical models.

3.2.1 Robot Geometric Characterization

In robotics, obtaining the mathematical model generally requires knowing the lengths and parametric angles that define the robot's structure.

3.2.1.1 Parametric Lengths

We consider the ternary symmetric configuration consisting of four identical kinematic chains arranged at 90° intervals. The parametric lengths and characteristic angles of this structure are represented in Figure. [3.1](#)

RA: Distance between the center of the fixed base and the rotation axis of the arm.

RB: Distance between the center of the platform and the side of the parallelogram attached to the platform.

LA: Length of the arm.

LB: Length of a parallel bar.

R: Difference between the lengths RA and RB; this distance defines the position of point A_i .

The latter is obtained by translating the amplitude R_B of the kinematic chain i , which moves: the point B_i to the center of the platform (nacelle), the point O_i to A_i , and the point C_i to C'_i . The rotation axis of the translated arm is thus denoted as a_i .

3.3 Mechanical Construction

The mechanical structure consists of various components shown in Figure. 3.1 and described in Tableau. 3.1

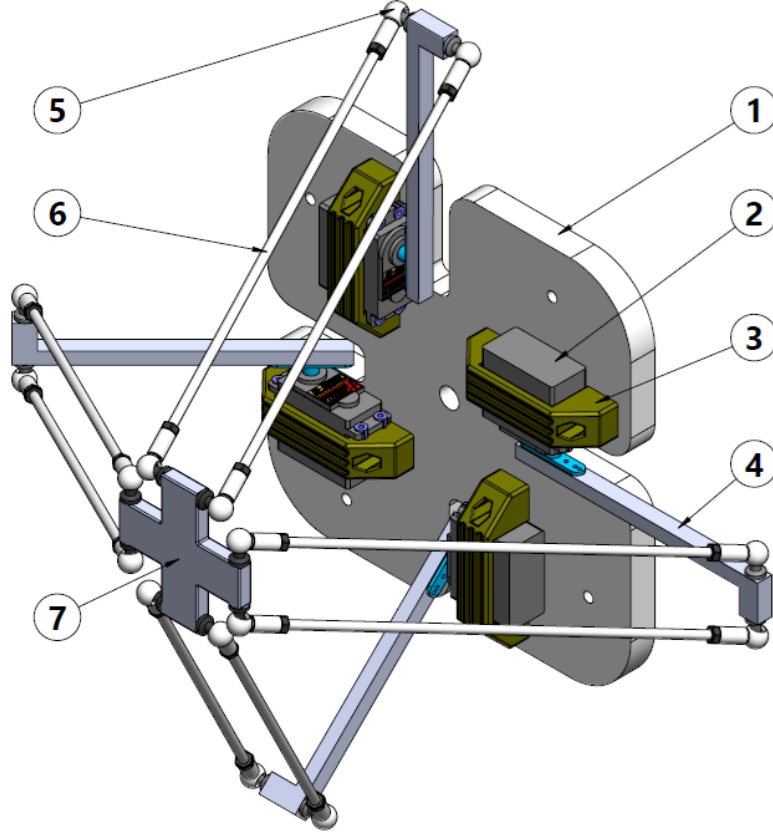


Figure 3.1: Schematic and actual photo of the components of the completed delta robot.

Table 3.1: Mechanical components and dimensions of the Delta robot.

N°	Part Name	Dimensions
1	Fixed base	$OP = 100$ mm
2	Motors	Rotation 0 to 180°
3	support	$110 * 25$ mm
4	Driving Arms PA	$PA = 197.62$ mm
5	Ball Joints	Rotation $\pm 30^\circ$ 0 to 180°
6	Lower Arms AB	$AB = 326.6$ mm
7	Moving platform	$CB = 45.96$ mm

3.3.1 Moving Platform

The moving platform is considered the end-effector of the Delta robot, connecting the parallel arms together. It is made of aluminum to ensure low weight (159 g), allowing the actuators to easily support its motion. The platform has a cross (+) shape with a radius of 45.96 mm.

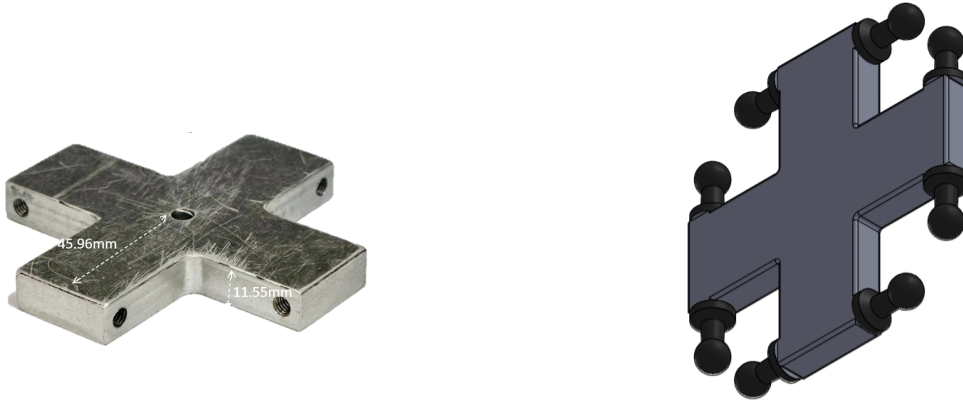


Figure 3.2: Actual photo of the Moving Platform.

3.3.2 Ball Joints

The joints used in this project are ball joints, which allow multi-directional movement between the robot links. They are made of steel and each one weighs approximately 20 g. The dimensions of the ball joints are shown in the Figure, and they were purchased from a local mechanical parts store.

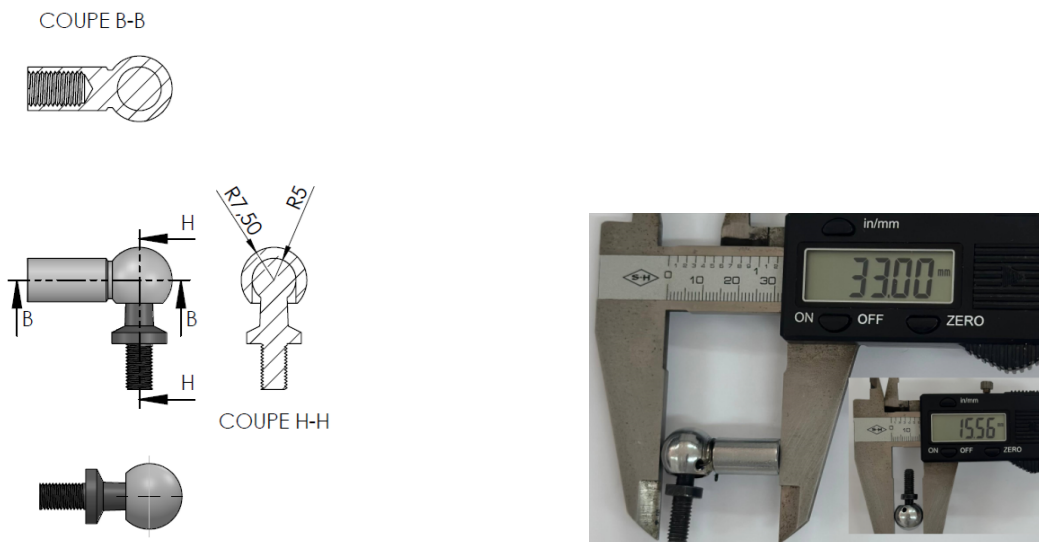


Figure 3.3: Actual photo of a ball joint with its dimensions.

3.3.3 Lower Arms AB

The lower arms connect the driving arms to the moving platform, forming the parallel structure of the robot. Each arm has a length of 341 mm and a weight of approximately 177.60 g. The shape and dimensions of the lower arms are shown in.

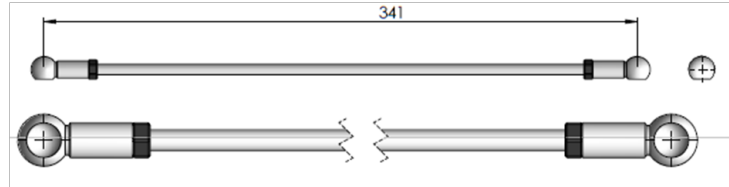


Figure 3.4: Actual photo of the lower arm AB with its dimensions

3.3.4 Driving Arms PA

Figure. 3.5 represents the shape and the corresponding length of the motor arms. The measured weight of the PA arm is 65.66 g. The PA arms are custom-made in an L shape from aluminum, while the AB arms are cylindrical threaded arms purchased from a hardware store.



Figure 3.5: Actual photo of The Driving Arm PA with it's dimensions.

3.3.5 Servomotors

Today, in the robotics field, many applications are implemented using electric actuators such as servomotors due to their ease of control, high precision, and ability to deliver considerable torque. For these reasons, the HS-5765MH servomotor was selected Figure. 3.6.

The HS-5765MH is equipped with metal gears and a 10 mm output shaft, and can deliver a torque of up to 25 kg·cm, making it suitable for handling significant loads. In this project, the total structure weighs approximately 600 g with arm lengths of 13 cm. A simple calculation shows that the motor can lift up to 1.92 kg (Figure. 3.3) , which satisfies the requirements of the system.



Figure 3.6: the HS-5765 MH servo motor and its accessories

3.3.6 Supports

The supports are used to hold and fix the servomotors onto the fixed base, ensuring the stability of the overall structure. They are custom-made parts designed to match the dimensions of the motor and assembly requirements. The shape and dimensions of the supports are shown in Figure. 3.7.

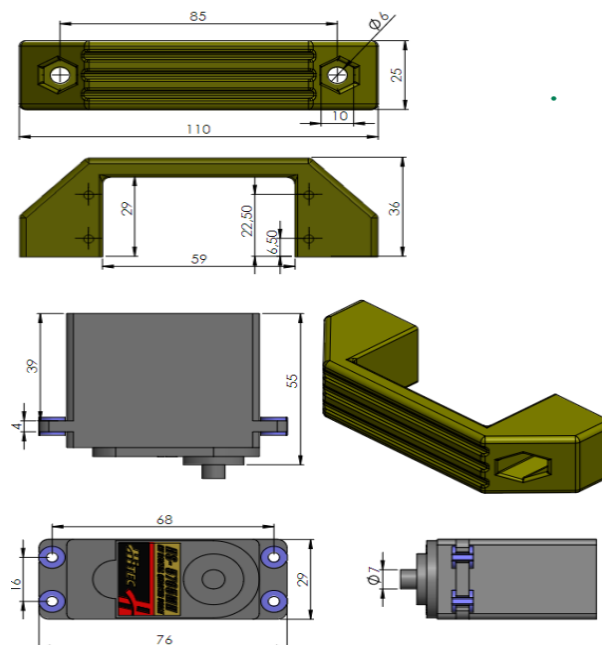


Figure 3.7: CAD technical drafting and dimensional specifications of the servomotor support bracket.

3.4 Internal Components of The servo

A servomotor is a special DC motor equipped with a gearbox and an electronic control circuit, allowing control of both the rotation direction and the position of the output shaft Figure. 3.8. Servomotors provide precise angular positioning and are widely used in robotics and in applications such as remote-controlled cars, aircrafts, and boats.

To operate properly, the servomotor uses a feedback system that continuously returns information about the current position of the output shaft to the control circuit through a transducer, typically a potentiometer integrated into the system.

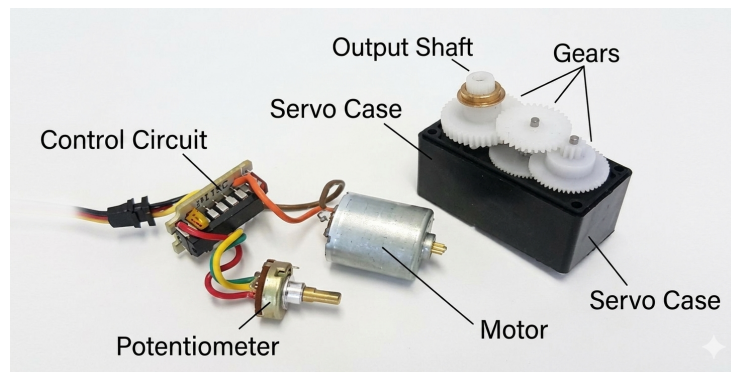


Figure 3.8: The components of a servomotor.

The control circuit compares the feedback signal (actual position) with the input command signal (desired position). The difference between these two signals, known as the error signal, is amplified and used to drive the motor in the appropriate direction until the error is minimized or eliminated.

When the output shaft reaches the desired position, the error becomes zero, and the motor stops. The general operating principle of the servomotor is illustrated in Figure. 3.9.

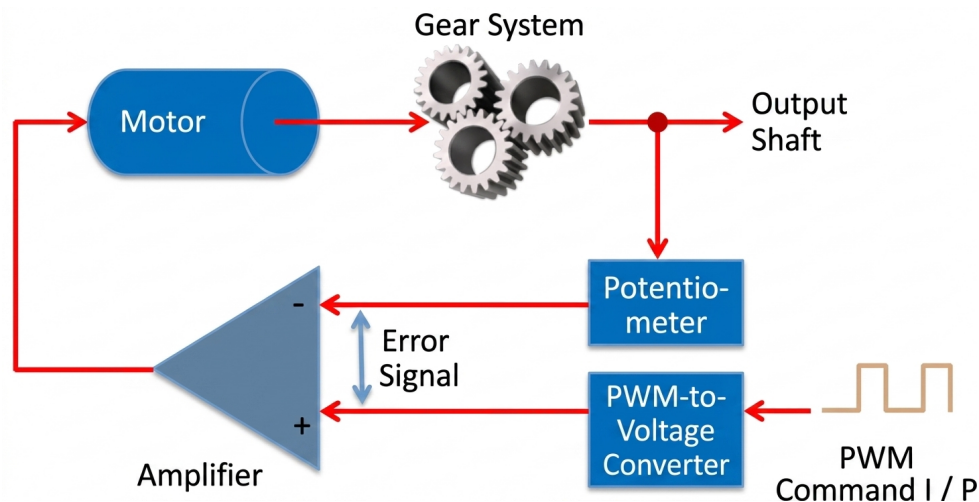


Figure 3.9: operating principle of a servomotor.

3.4.1 Fixed Base

To construct the fixed base, a square configuration is adopted to ensure a symmetrical arrangement of the four servomotors, with 90° between each motor. This configuration allows for proper distribution of the actuators and improves the stability of the robot.

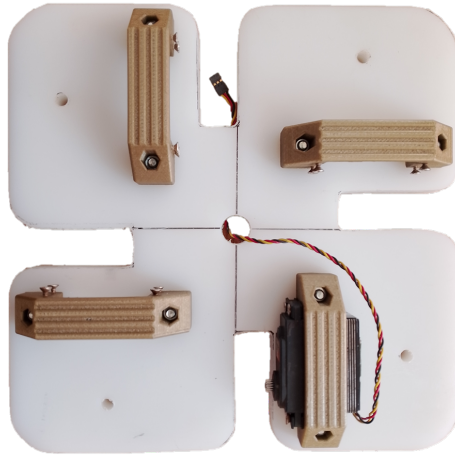


Figure 3.10: Actual photo showing how the servos are mounted on the base.

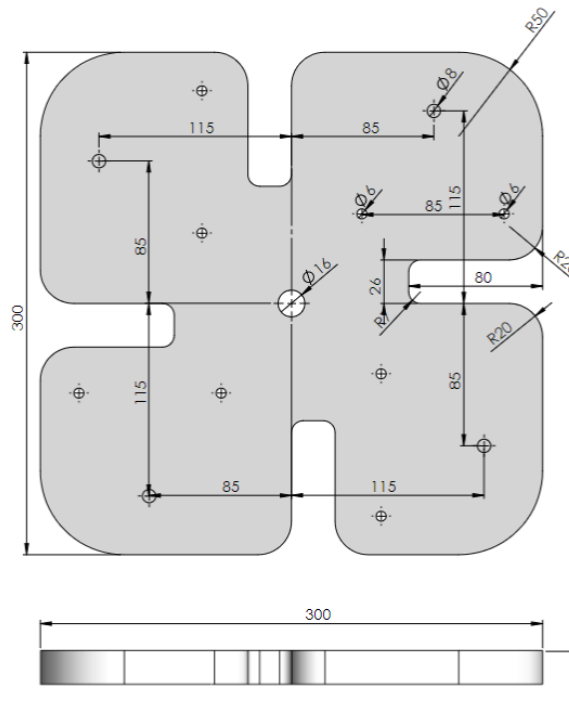


Figure 3.11: Auto CAD drawing showing the positioning and placement of the servomotor on the base.

The servomotors are fixed at specific positions on the base according to the geometric dimensions, ensuring accurate alignment and proper operation of the mechanism. The shape and dimensions of the fixed base are shown in Figure. [3.11](#).

3.4.2 Fixing the base to a support

A support has been constructed to secure the base at a specific height



Figure 3.12: real image of the support created

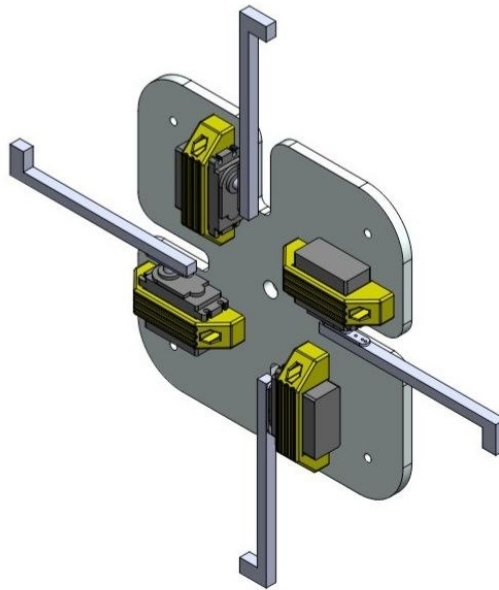


Figure 3.13: Design of the Fixed base and Servomotors for the 4-Arm Delta Robot

3.5 The electrical part

3.5.1 Power Supply Circuit

The power supply stage provides a stabilized 5 V supply required for the servomotors and control system. In this work, an ATX-450W computer power supply is used, as it offers a high current capacity and integrated protection features, ensuring safe and stable operation under varying load conditions.

This type of power supply is well suited for robotic applications where multiple actuators are used.

- **Remark:**

The ATX power supply is activated by connecting the green wire (PS_ON) to a black wire (ground), as shown in Figure. 3.14



Figure 3.14: Power-on of the ATX-450W power supply

3.5.2 The electronic Circuit

In this section of the project, we will discuss the various components of the control circuit, with detailed explanations of each component Figure. 3.15.

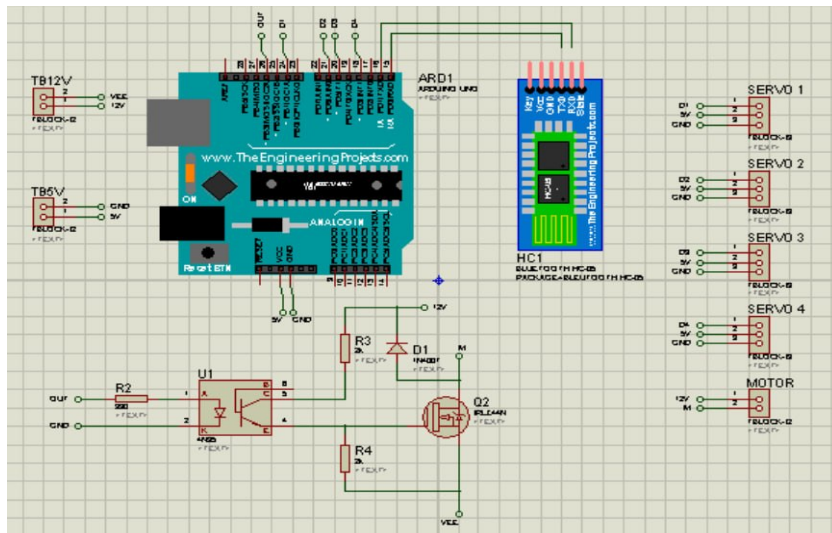


Figure 3.15: Complete Electronic Circuit Schematic

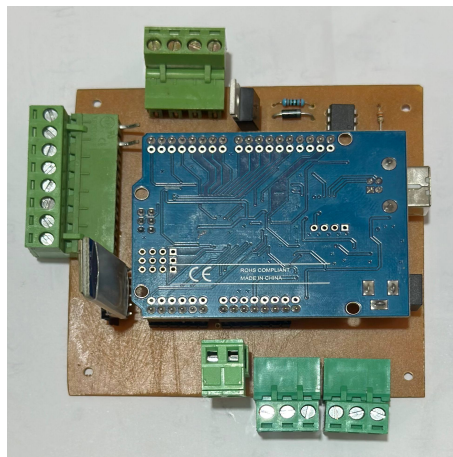


Figure 3.16: Custom Control PCB Layout

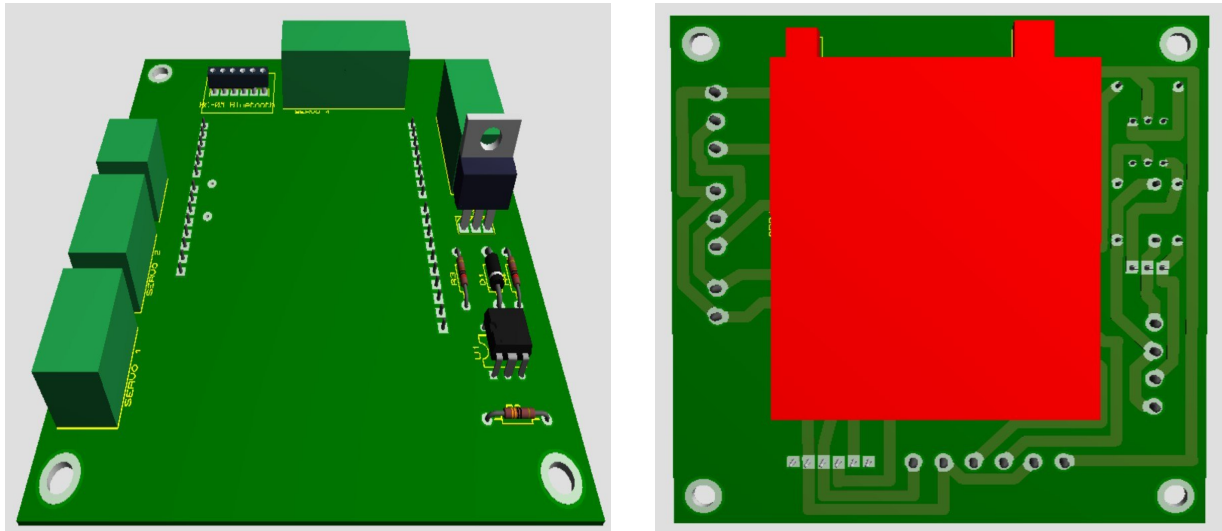


Figure 3.17: 3D Render of the Custom Control and Power Distribution Board

3.5.3 components

3.5.3.1 Arduino Uno

We choose Arduino UNO as a controller. Arduino UNO is a microcontroller board built on the ATmega328P. It contains 14 digital input/output pins (6 of which may be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB port, a power connector, an ICSP header, and a reset button. It includes everything required to support the microcontroller; simply attach it to a computer via a USB cable.

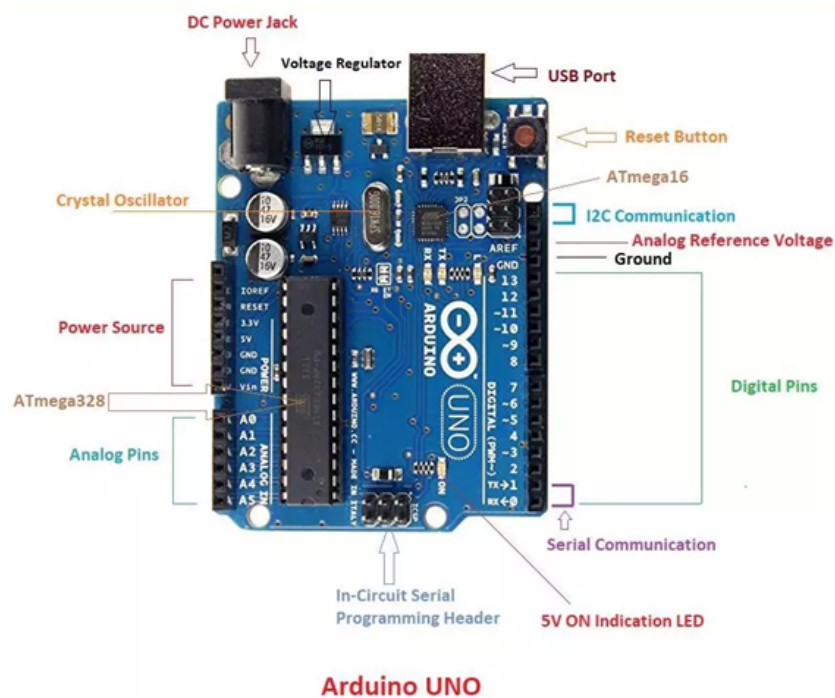


Figure 3.18: Arduino uno

3.5.3.2 Features of the Arduino Uno (ATmega328P)

- **Clock speed:** 16 MHz via an external crystal oscillator, ensuring precise synchronization of PWM signals for the four arms.
- **Flash memory :** 32 KB, dedicated to storing the control program and kinematic algorithms.
- **SRAM :** 2 KB, used for real-time calculation of $[x, y, z]$ coordinates and management of dynamic variables.
- **Inputs/Outputs (I/O):** Includes 14 digital pins (including 6 PWM outputs for servo control) and 6 analog inputs.
- **Communication :** Integrated UART for serial communication via the DB9 connector, allowing interfacing with a PC.

3.5.3.3 Pin Configuration and Connections

- **Power Supply:** The Arduino Uno is powered by a regulated 5V supply, while the actuators are powered independently to ensure system stability.
- **Servomotors (PWM Control):** Digital pins 3, 5, 6, and 9 are assigned to control the four Hitec HS-5765MH servo motors.
- **Electromagnet Control:** Pin 10 is used to drive the electromagnet via an IRLZ44N MOSFET acting as a high-speed switch.
- **Gate Resistor:** A 10 $k\Omega$ resistor is connected between the Arduino pin and the MOSFET gate to protect the microcontroller and ensure reliable switching.

3.5.3.4 Electromagnet

An electromagnet is an actuator positioned at a robot's end-effector. It produces a magnetic field when an electric current flows through it, enabling the robot to pick up and release metallic objects smoothly and precisely during automated operations.

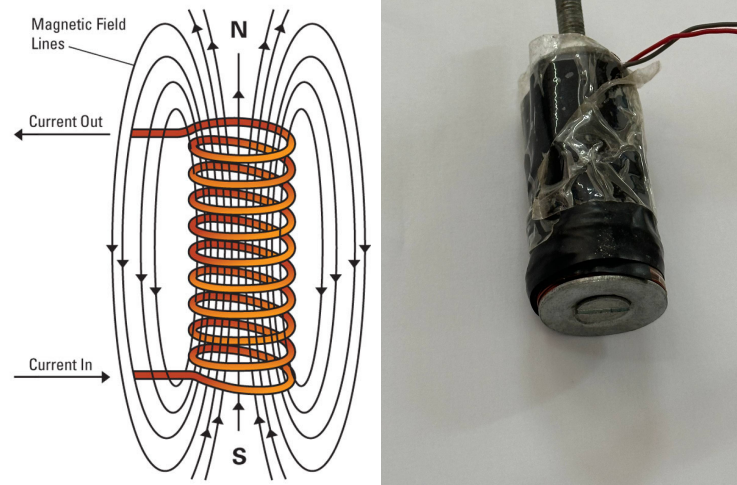


Figure 3.19: Electromagnet

3.5.3.5 optocoupler

An optocoupler is an optoelectronic semiconductor device used to transmit electrical signals between two completely isolated circuits via light. It provides complete electrical isolation between the low-voltage control circuit and the high-voltage power circuit.

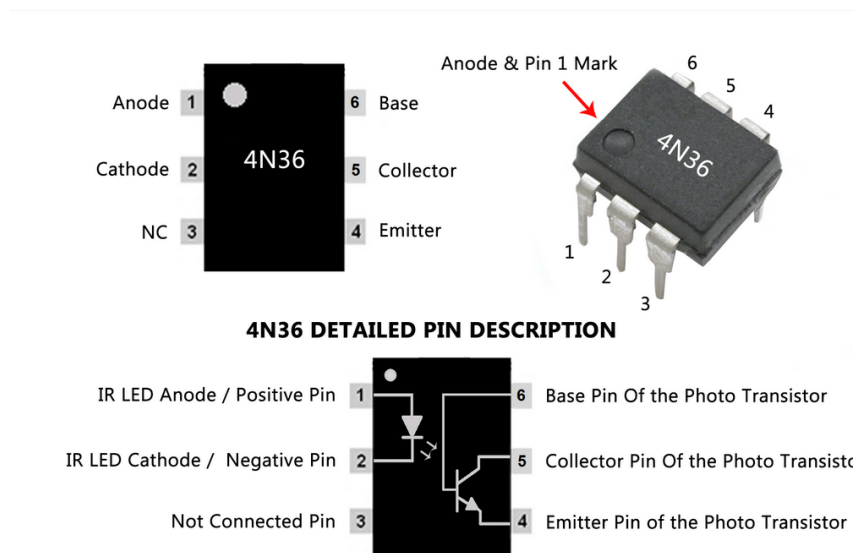


Figure 3.20: optocoupler
[35]

3.5.3.6 transistor

- **Design and Application of IRFZ44N MOSFET for High-Power Load Control Using Arduino**

The IRFZ44N is an N-channel power MOSFET used as an electronic switch for controlling high-power loads. It operates by applying a voltage to the gate terminal, which regulates the current between the drain and source, allowing fast and efficient switching.

In this project, it is used as an interface between the Arduino Uno and high-power devices such as motors and pumps, enabling safe control of high currents using low-power signals.

The device supports up to 55 V drain-source voltage, 49 A continuous current, and has a very low on-resistance of about 0.0175Ω , which improves efficiency and reduces power loss.

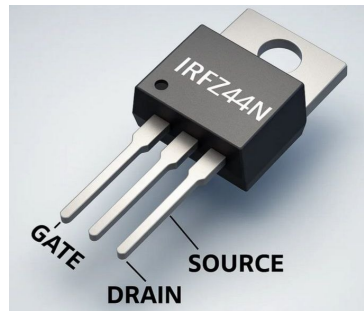


Figure 3.21: Transistor IRFZ44N
[36].

3.5.3.7 The HC-05 Bluetooth Module

The HC-05 module Figure. 3.22 is an easy-to-use Bluetooth SPP (Serial Port Protocol) module, designed for a transparent wireless serial connection setup. This serial port Bluetooth module is fully qualified Bluetooth V2.0 + EDR (Enhanced Data Rate) 3Mbps complete modulation with a 2.4 GHz radio transceiver and baseband [37].



HC-05 Bluetooth



Figure 3.22: HC-05 Bluetooth Module Views

Figure. 3.23 illustrates how to interface the HC-05 module with the Arduino Uno.

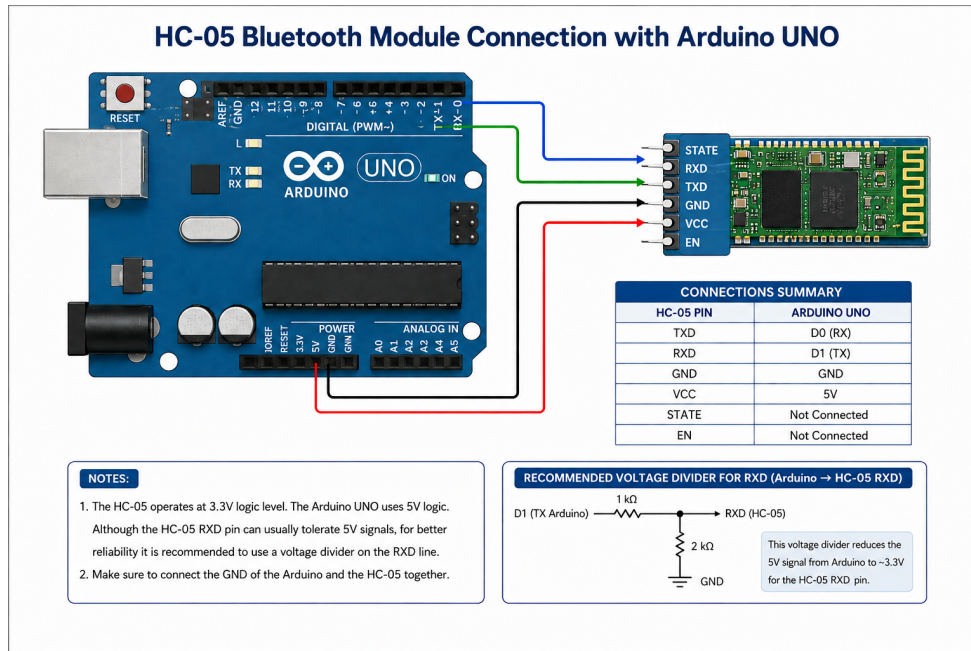


Figure 3.23: UART Connection between Arduino UNO and HC-05 Bluetooth Module

3.6 Conclusion

In summary, this chapter has signified the project's transition from theoretical modeling to physical implementation. We have outlined the mechanical design, with a focus on the use of lightweight and durable materials, and have established the electronic architecture, which is centered around the Arduino Uno and high-torque servomotors.

The Delta robot's structure, power, and control circuits are successfully assembled, establishing the fundamental physical foundation for the subsequent phase of programming and testing control algorithms in a real-world setting.

Chapter 4 | Software Development and Control using Image Processing

4.1 Introduction

Following the hardware and electrical design provided in Chapter 3, this chapter concentrates on the robotic system's software implementation and control. It provides a vision-based technique that employs MATLAB for image acquisition and processing and Arduino for real-time actuation. To enable correct interaction with the robot, the system incorporates image processing, coordinate transformation, and MATLAB/Arduino communication. Finally, several control strategies for servo motor operation and system execution are described.

4.2 Image Acquisition and Vision System

The acquisition chain for the suggested system is shown in Figure. 4.1 The scene with the target object is captured by the camera, and MATLAB is used to process the obtained image. The calibration relation is used to convert the detected object's position from pixel coordinates (x_p, y_p) to Cartesian coordinates (X, Y). The Arduino board receives these coordinates and uses them to create the appropriate control action for the robot.

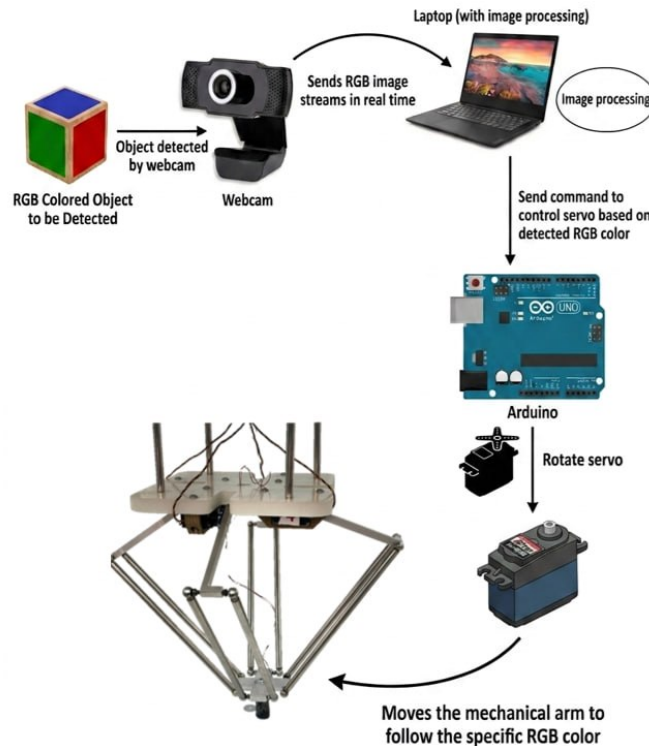


Figure 4.1: Vision-Based Control Architecture

4.2.1 Image Processing and Color Detection

Following image acquisition, the collected frames are analyzed in MATLAB to identify the target item and determine its location. The procedure relies on color analysis and geometric feature extraction.

4.2.1.1 RGB Image Representation

In MATLAB, color images are represented as a three-dimensional matrix of size $M \times N \times 3$, with the third dimension representing the Red, Green, and Blue color channels. Each pixel is represented by an RGB triplet including its intensity values. Each channel in 8-bit pictures runs from 0 to 255, resulting in up to 16.7 million color choices. This approach allows for independent processing of each color component during picture analysis and object detection[38].

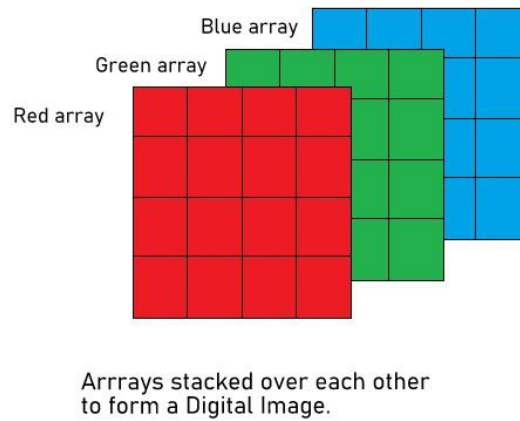


Figure 4.2: RGB image representation as three stacked color components (Red, Green, Blue).

4.2.1.2 Color Plane Separation

An RGB image can be split into three independent matrices representing the red, green, and blue channels. In MATLAB, this is accomplished by matrix indexing: $I(:, :, 1)$, $I(:, :, 2)$, and $I(:, :, 3)$. Each plane contains the intensity distribution of a single color component, allowing for color-based segmentation[38].

$$R = I(:, :, 1);$$

$$G = I(:, :, 2);$$

$$B = I(:, :, 3);$$

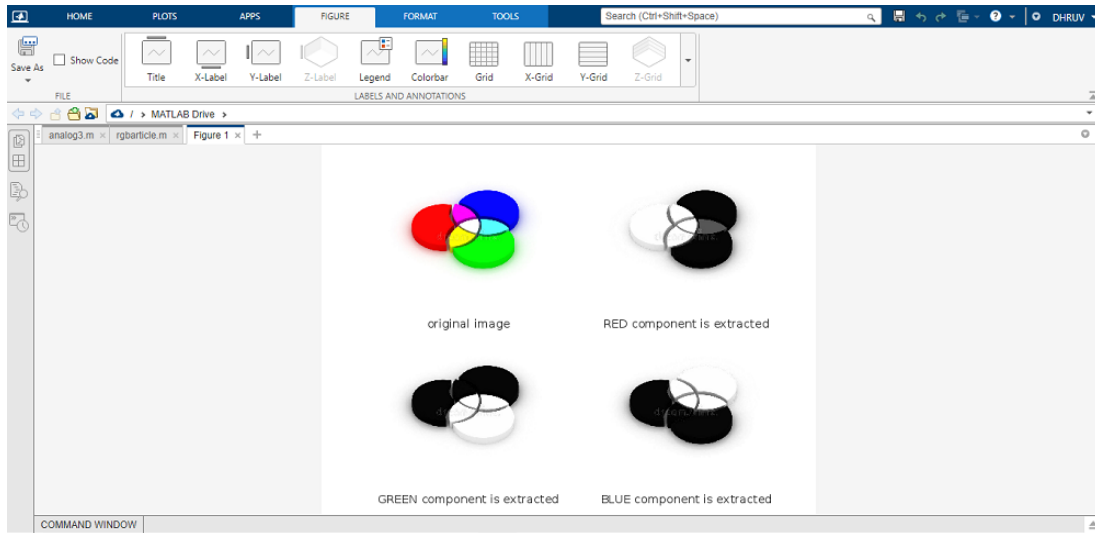


Figure 4.3: Decomposition of the RGB image into its three color planes (Red, Green, Blue).

4.2.1.3 Target Color Detection

Target color detection involves analyzing the image's color information to isolate a particular object on the basis of its color characteristics. In this study, the HSV color space is used to improve resilience and thresholding is used to extract the desired color region. The result is a binary mask of the observed object[38].

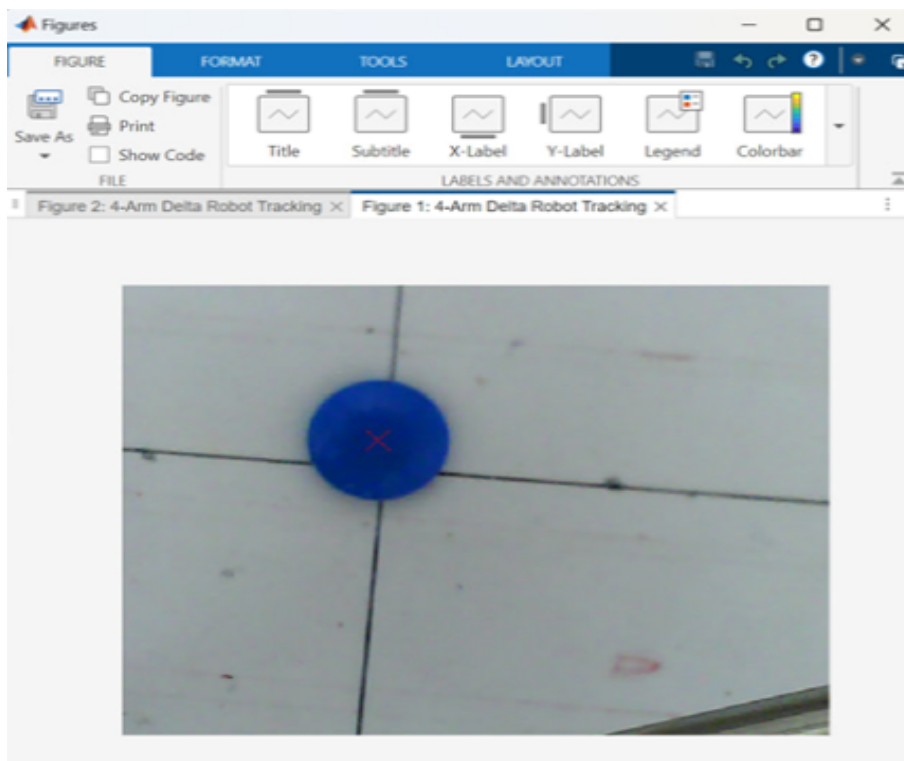


Figure 4.4: Detection of the target object using color-based segmentation in MATLAB.

```
hsvImg = rgb2hsv(img);  
blueMask = (hsvImg(:,:,1) > 0.55) & (hsvImg(:,:,1) < 0.75) & ...  
           (hsvImg(:,:,2) > 0.4) & (hsvImg(:,:,3) > 0.2);
```

The following algorithm summarizes the complete logical steps performed in MATLAB to detect the object and prepare the data for the robot:

Algorithm 4.1 : Object Detection and Coordinate Mapping

Input : RGB Image Frame from Webcam

Output : Target Coordinates (x, y) and Joint Angles

Convert Image from RGB to HSV space;

Apply color thresholding to create **blueMask**;

Extract object Centroid in pixels (X_p, Y_p) ;

Map pixels to Real-World coordinates (x, y) using calibration;

// Note: This data is then used for Inverse Kinematics

Vision Processing Logic: The visual detection system operates through a sequential algorithmic approach consisting of two primary stages.

HSV Segmentation: To isolate the target objects, the captured RGB image is first converted to the HSV (Hue, Saturation, Value) color space. This color model is more robust to lighting variations than RGB. The system achieves accurate segmentation of blue objects by applying a threshold filter on the Hue channel within the specific range:

$$0.55 < H < 0.75$$

Coordinate Mapping: Once the object is detected, a mapping process is performed to translate the pixel coordinates (X_p, Y_p) from the image frame into real-world physical coordinates (x, y) measured in millimeters. This transformation is based on a linear calibration model, which accounts for the camera's height and orientation. The relationship is exemplified by the following linear equation:

$$x = 0.082 \cdot X_p + 14.37$$

This calibrated data is subsequently utilized as the target input for the inverse kinematics block to position the Delta robot's end-effector.

4.2.1.4 Object Position Extraction

After detecting the target object, the centroid of the segmented region is used to compute its position. This centroid represents the object's position in pixel coordinates (xp,yp) within the image frame.

```
stats = regionprops(blueMask, 'Area', 'Centroid');

math_is_safe = false;

if ~isempty(stats)
    [~, idx] = max([stats.Area]);
    centroid = stats(idx).Centroid;
```

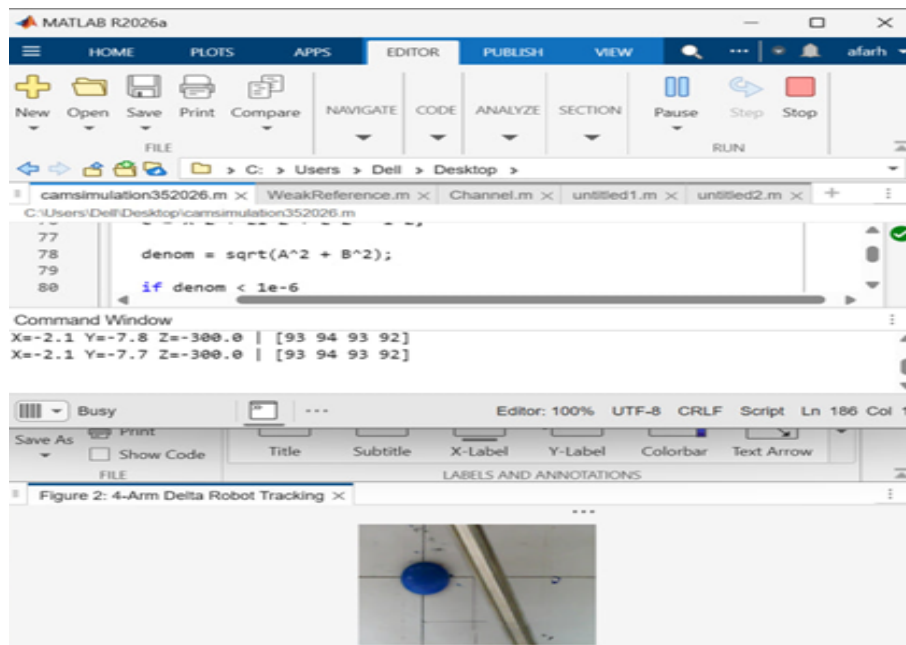


Figure 4.5: Extraction of object position in pixel coordinates from MATLAB processing output.

4.2.1.5 Pixel-to-Coordinate Concept

The pixel coordinates obtained during image processing are not directly applicable to robotic control. As a result, a transformation is necessary to translate pixel values to real-world coordinates using a calibration model. This mapping facilitates the integration of vision data into the robot control system.

```
% --- Pixel to mm Mapping ---
Cx = centroid(1);
Cy = centroid(2);
Cz = -300;
```

Figure 4.6: Pixel-to-coordinate transformation for robot workspace mapping.

4.2.2 Calibration and Coordinate Transformation

Calibration was used to convert pixel coordinates generated from image processing into real-world coordinates that could be used by the delta robot's end effector. This was accomplished experimentally using axis alignment, offset correction, and scaling based on measured positions.

Step 1: Axis Alignment (Coordinate System Matching)

The detected centroid is given by:

$$(x_p, y_p) = (\text{centroid}(1), \text{centroid}(2))$$

observing object movement, it was found that the image axes do not match the workspace. Therefore, the coordinates are swapped:

$$Cx = \text{centroid}(2), \quad Cy = \text{centroid}(1)$$

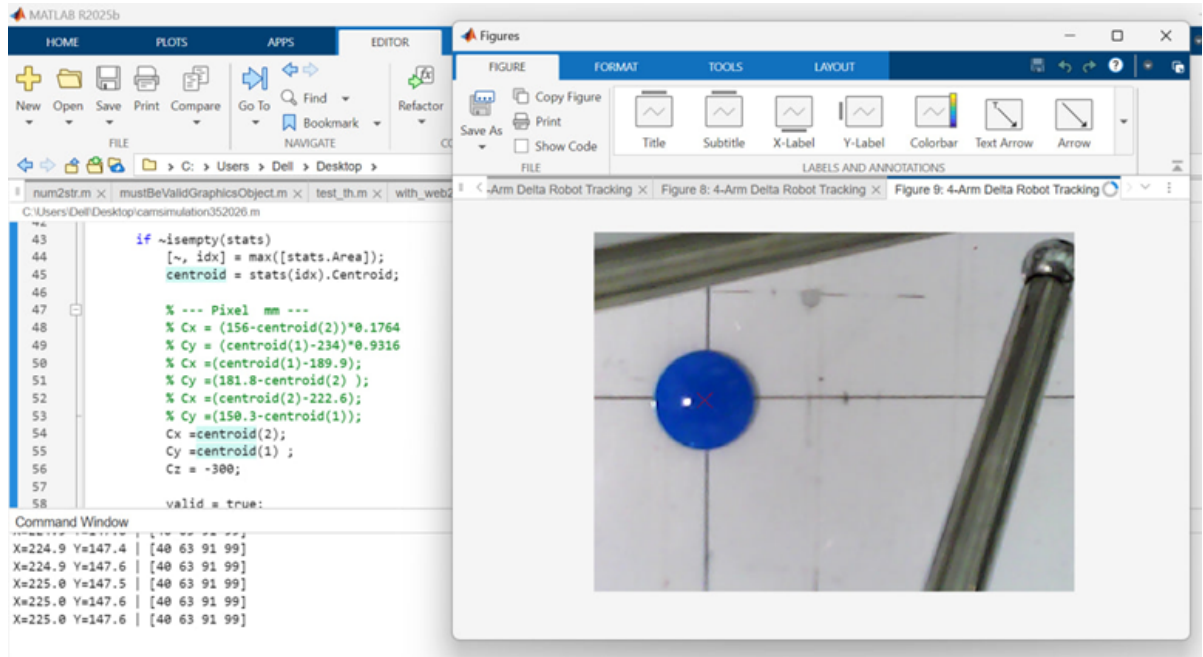


Figure 4.7: Alignment between image and workspace coordinate systems.

Step 2: Origin Adjustment

The object is placed at the center of the workspace (0,0), and its centroid values are recorded:

$$x_0 = 225.2, \quad y_0 = 147.4$$

These offsets are removed:

$$C_x = (\text{centroid}(2) - 225.2)$$

$$C_y = (147.4 - \text{centroid}(1))$$

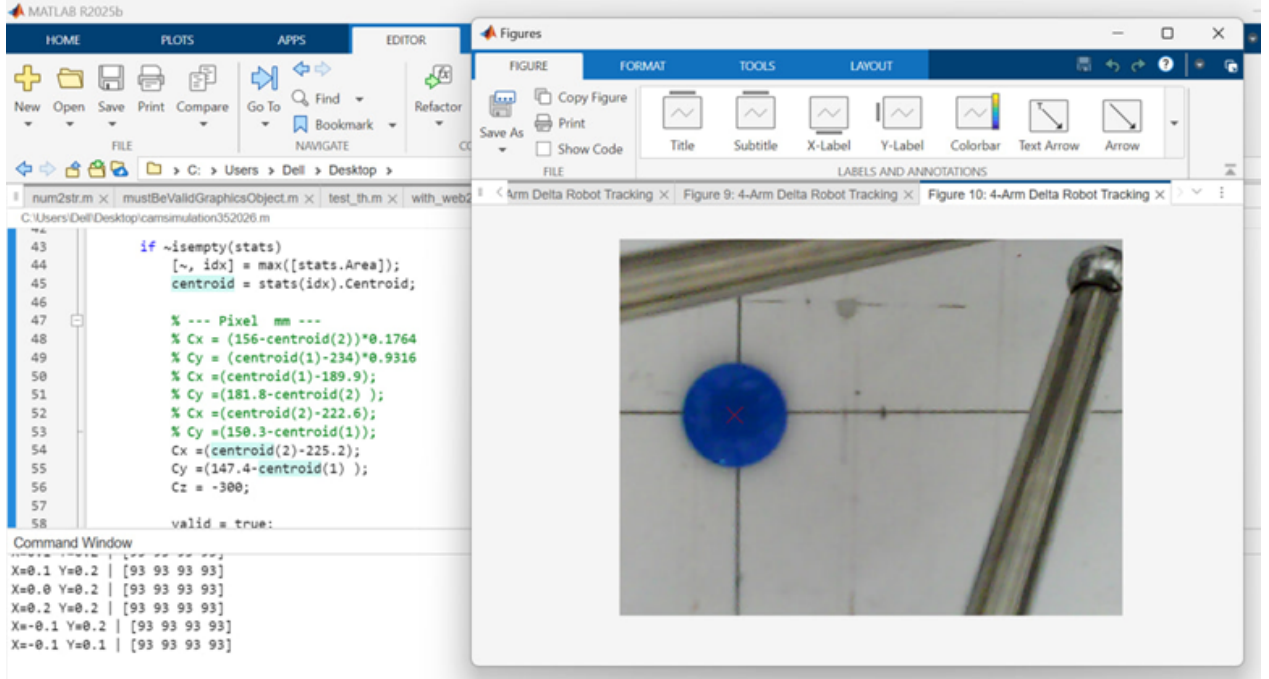


Figure 4.8: Defining the origin at the center of the workspace.

Step 3: Scaling

After axis and origin alignment, scaling factors were determined experimentally. The object was placed at a known position in the workspace ($x = -4, y = 3$) and the corresponding centroid values were recorded.

By comparing real-world measurements with pixel differences, scaling coefficients were calculated for each axis. Final equations:

$$C_x = ((\text{centroid}(2) - 225.2) * 0.020833) * 10;$$

$$C_y = ((147.4 - \text{centroid}(1)) * 0.023942) * 10;$$

$$C_z = -220;$$

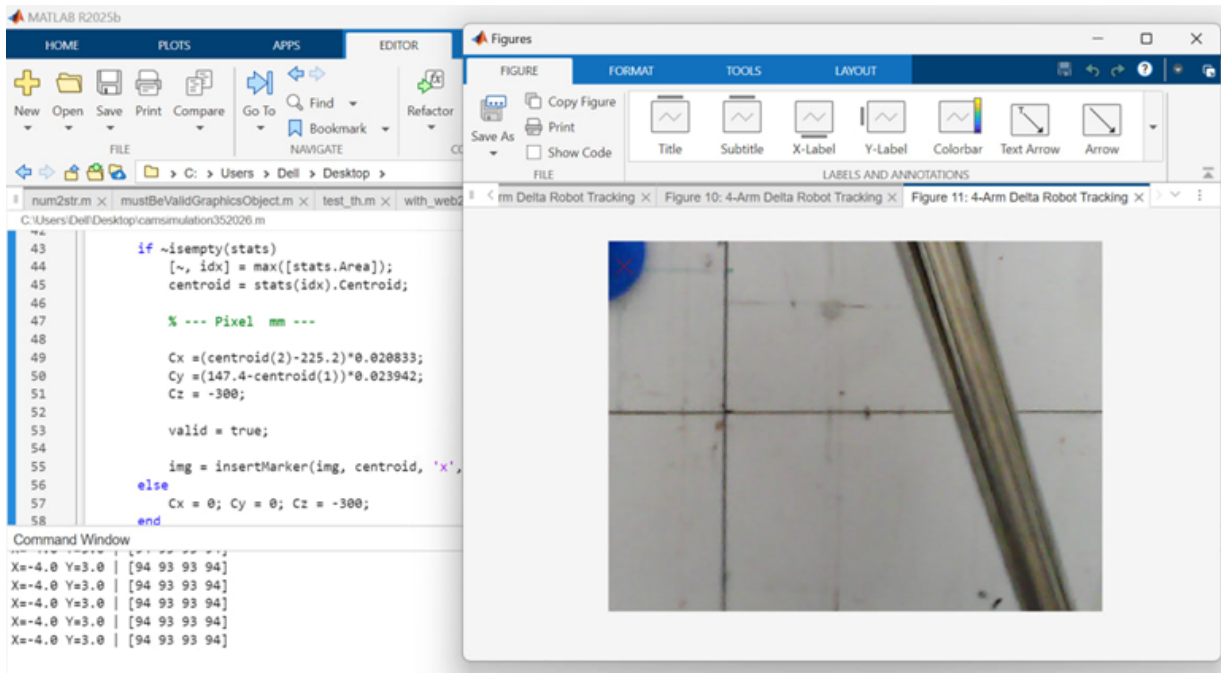


Figure 4.9: Conversion from pixel values to real-world units.

4.3 MATLAB to Arduino Communication

Serial transmission is used to establish communication between MATLAB and the Arduino, which sends control commands from the vision system to the delta robot. The computed angles are sent in real time and executed by the servos.

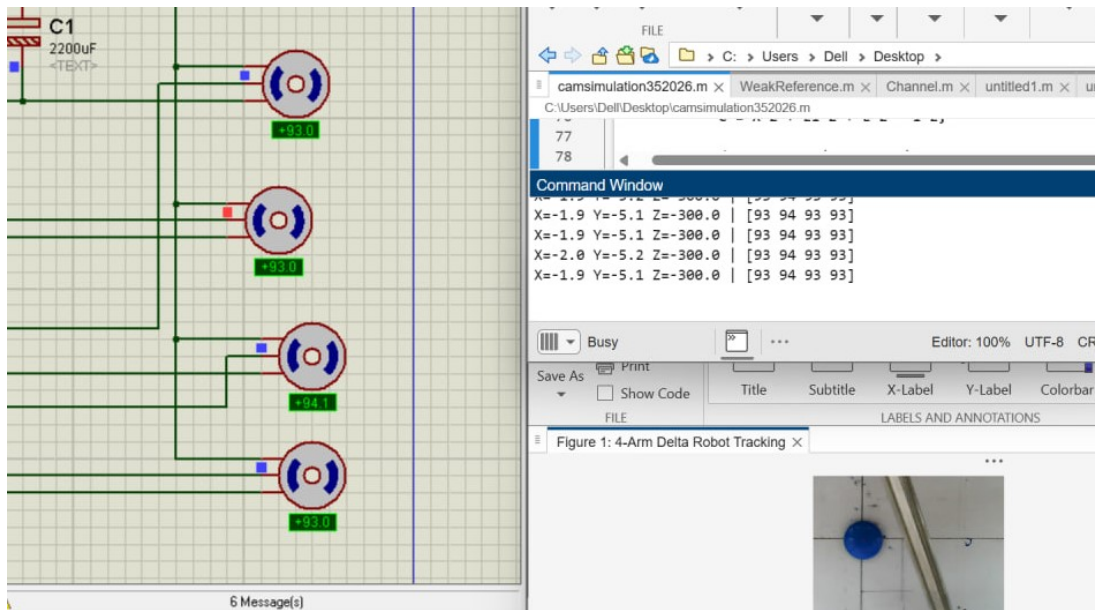
Step 1: transmitting computed servo angles from MATLAB

The computed servo angles are transmitted from MATLAB to the Arduino through serial communication. These values are directly applied to the actuators, resulting in synchronized motion of the end-effector. The simulation confirms that the detected object position, computed coordinates, and resulting actuator angles are consistent.

Step 2: Serial Communication Setup

A serial connection is created between MATLAB and Arduino with a specific COM port and baud rate (9600 bps). This ensures that data is transmitted synchronously between the two systems.

MATLAB transmits the computed angles as ASCII-encoded, comma-separated values. On the Arduino side, these values are decoded using serial parsing functions and converted into integer commands for servo control.



```
msg = sprintf('%d,%d,%d,%d\n', th(1), th(2), th(3), th(4));
write(s, msg, "char");
```

Figure 4.10: Transmission of computed servo angles from MATLAB and corresponding response in simulation.

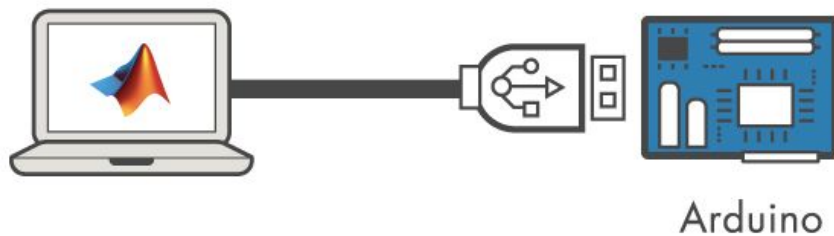


Figure 4.11: Conceptual representation of serial communication between MATLAB and Arduino.

A serial connection is established between MATLAB and Arduino using a predefined COM port and a baud rate of 9600 bps.

Step 3: Receiving Data in Arduino The Arduino reads incoming data using serial functions and extracts the transmitted angles. These values are then assigned to the corresponding servo motors.

```
void setup() {
  Serial.begin(9600);
  Serial.setTimeout(10);
}
```

(a) Serial setup

```
void loop() {
  if (Serial.available() > 0) {
    // Read the comma-separated values sent by MATLAB
    int in1 = Serial.parseInt();
    int in2 = Serial.parseInt();
    int in3 = Serial.parseInt();
    int in4 = Serial.parseInt();

    if (Serial.read() == '\n') {
      myservo1.write(in1);
      myservo2.write(in2);
      myservo3.write(in3);
      myservo4.write(in4);
    }
  }
}
```

(b) Main loop

Figure 4.12: Arduino program receiving and parsing serial data from MATLAB.

The Arduino reads the coordinates sent by MATLAB and converts them into numerical values for motor control. The following algorithm summarizes the complete reception and servo control logic on the Arduino side:

Algorithm 4.2 : Data Reception and Servo Control Logic (Arduino Side)

Input : Serial data string from MATLAB

Output : Servo motor positions $(\theta_1, \theta_2, \theta_3, \theta_4)$

Initialize:

Attach 4 Servos to PWM-enabled Digital Pins;

Set Serial Baud Rate to 9600;

Move Servos to Initial Home Positions;

while true do

if *Serial Data is Available in Buffer* **then**

 Read string until Line Break ($\backslash n$);

 Parse string into numerical variables: $(\theta_1, \theta_2, \theta_3, \theta_4)$;

 // Actuation

 Move Servos to target degrees using `servo.write()`;

 Delay (ms) to allow mechanical stabilization;

String Parsing: The controller carries out string parsing, where the incoming ASCII string is tokenized and converted into numerical integers using parsing functions. This process effectively isolates each of the four joint angles $(\theta_1, \theta_2, \theta_3, \theta_4)$ from the received serial buffer.

Servo Actuation: Following the parsing stage, servo actuation occurs; the processed angles are sent to their respective motors via the `servo.write()` command. A strategic delay is incorporated between movements to allow for mechanical stabilization. This pause is essential for minimizing inertia-induced vibrations and ensuring the smooth operation of the 4-arm Delta structure.

4.4 Control System Implementation

4.4.1 Control Logic Overview

The control system integrates image processing, calibration, and actuator control into a single real-time workflow. The camera detects the target object and extracts its position in pixel coordinates, which are converted into real-world coordinates using the calibration model. These values are then transformed into servo commands and transmitted from MATLAB to Arduino, enabling real-time motion of the delta robot end-effector.

The system operates through three control modes: direct angle control, inverse kinematics, and vision-based tracking.

4.4.2 Servo Control and Offset Calibration

To achieve perfect placement, each servo has to be calibrated individually. practical testing revealed small differences amongst actuators. To account for these differences, specific pulse width restrictions were set for each servo during setup. These offsets enable for constant and coordinated action of the end-effector across all actuators.

```
myservo1.attach(3, 570, 2370);  
myservo2.attach(5, 600, 2400);  
myservo3.attach(6, 660, 2460);  
myservo4.attach(9, 670, 2470);
```

Figure 4.13: Servo-specific pulse width calibration parameters used to compensate for actuator offsets.

4.5 Servomotor Control

The servo motor is controlled using a Pulse Width Modulation (PWM) signal with a frequency of 50 Hz (20 ms period). The width of the pulse determines the angular position of the output shaft. Inside the servo, the signal is converted into a reference voltage and compared with a feedback signal from an internal potentiometer representing the actual shaft position. Any error between the desired and actual positions generates a corrective action until equilibrium is reached. In practice, pulse widths typically range between 0.6 and 2.4 ms, where 1.5 ms corresponds to the neutral position (90°).

The relationship between pulse width and angular position is illustrated in Figure. 4.14. Due to the high current demands of these actuators, a dedicated ATX-450W power supply is integrated into the system to ensure voltage stability and prevent logic resets of the microcontroller.

Additionally, a synchronization delay is implemented to allow the mechanical structure to reach equilibrium before subsequent movements, effectively dampening inertia-induced oscillations.

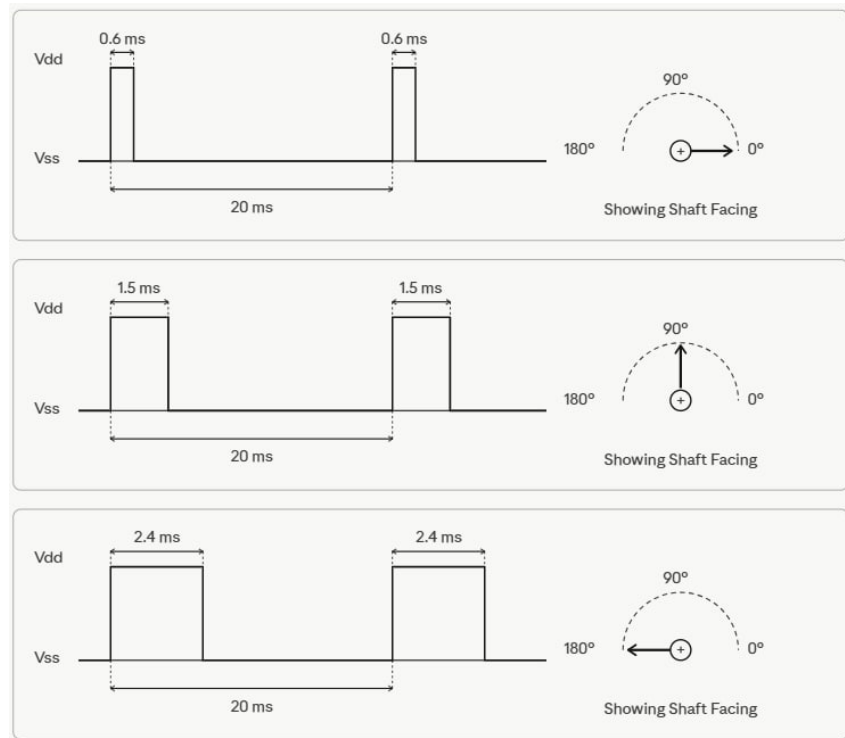


Figure 4.14: PWM Timing for different shaft positions.

4.6 conclusion

This chapter describes the delta robot system's software and control architecture. It included image processing, calibration, MATLAB-Arduino interface, and real-time control algorithms based on vision and kinematic modeling. The combination of these modules allows for precise and coordinated movements of the end effector. The following chapter will cover the system's practical applications, such as pick-and-place operations, drawing jobs, and real-world implementation of the three control modes.

Chapter 5 | Applications and Performance Evaluation of the Delta Robot

5.1 Introduction

This chapter covers the performance characteristics of the Delta robot prototype, including workspace, speed, accuracy, and load capacity. Despite time restrictions and limited resources that slowed complete industrial implementation, it highlights useful applications including pick-and-place operations and trajectory-based activities, showing the effectiveness of the suggested control system.

5.2 Delta Robot Performance

This section describes the experimental evaluation of the developed Delta robot. Figure 5.1 represents an actual view of the implemented system. The key performance qualities of the robot are determined by a series of tests, which include:

- Workspace (volume of operations)
- speed
- Accuracy
- Maximum payload

Following the mechanical and electrical designs provided in earlier chapters, the robot is tested by completing practical tasks. These tests validate the system's behavior and overall performance.

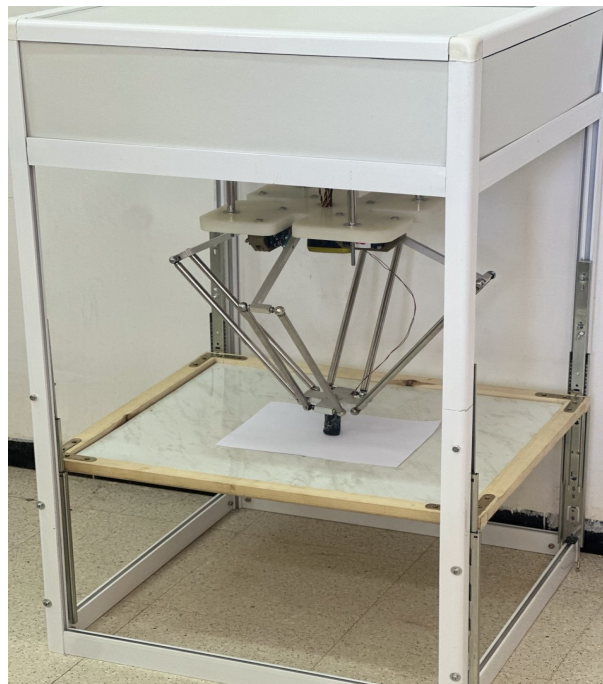


Figure 5.1: Real prototype of the four-arm Delta robot.

5.2.1 Workspace (Working Volume)

The Delta robot The workspace represents the set of all positions that can be reached by the end-effector (point P). The addition of a tool introduces a vertical offset, resulting in a downward shift of the effective working region.

Due to the finite dimensions of the robot structure and the limited range of motion of its joints, the displacement of the moving platform is restricted to a bounded region in space. Furthermore, additional constraints are imposed to avoid singular configurations and ambiguities in the coordinate transformations.

The determination of the workspace is carried out experimentally using the developed control interface.

Within this space, the motion of the robot is limited by three main types of constraints:

- Joint angle limits of the actuated arms.
- Mechanical constraints of passive joints (spherical joints).
- Possible collisions between robot links

To ensure safe operation and avoid singularities, the joint angles are restricted as follows:

To ensure safe operation, joint angles are limited within predefined ranges, preventing excessive motion and singularities.

$$-\gamma_{max} \leq \gamma_i \leq \gamma_{max}, \quad \text{with} \quad \gamma_{max} = 30^\circ$$

$$30^\circ \leq (\alpha_i + \beta_i) \leq 160^\circ$$

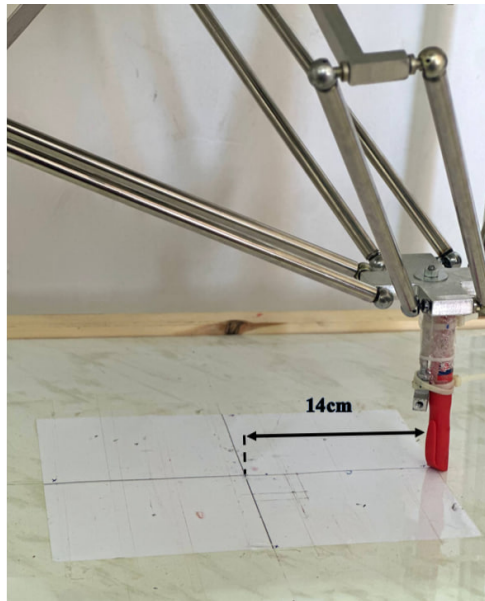


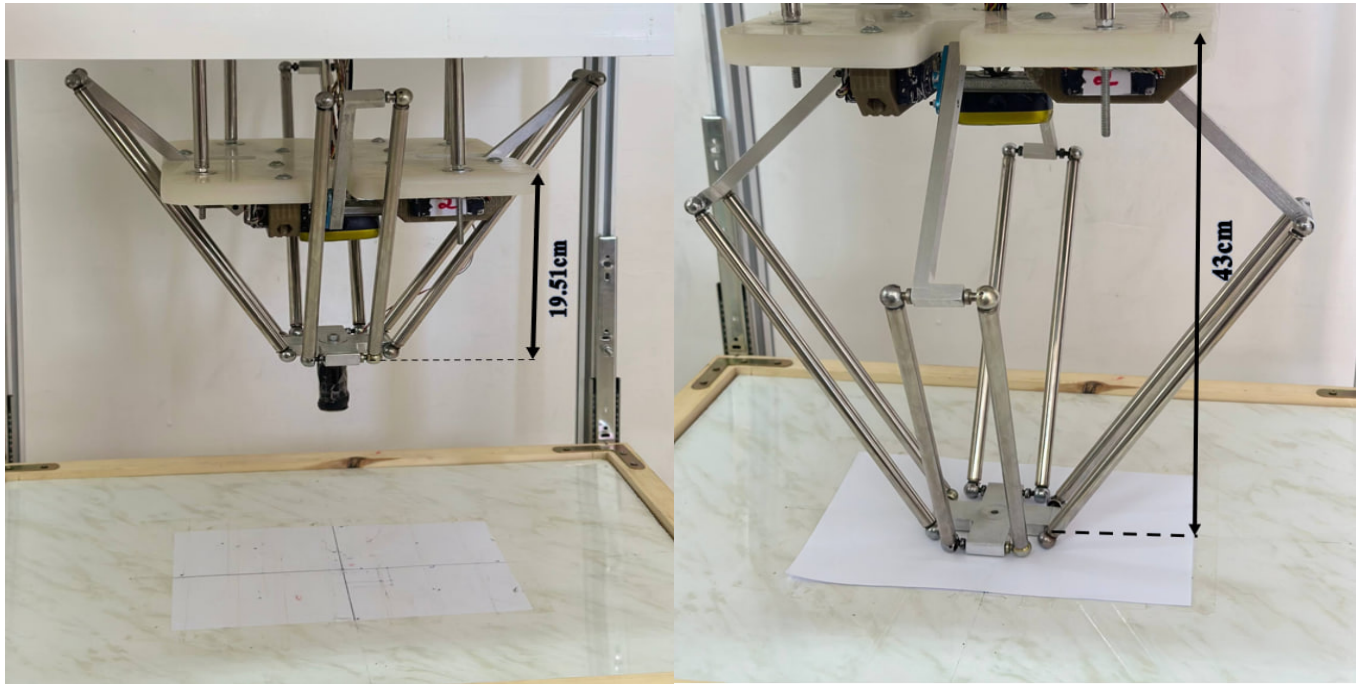
Figure 5.2: Characteristic angles defining the workspace of the Delta robot.

5.2.1.1 Minimum and Maximum Height

Among the key characteristics of the workspace are the minimum and maximum reachable heights of the robot. For the developed system:

- Minimum height: $H_{min} = 19.51$ cm
- Maximum height: $H_{max} = 43$ cm

These values are obtained experimentally using the control interface, as shown in the Figure. 5.3.



(a) Minimum reachable height of the Delta robot (H_{min})

(b) Maximum reachable height of the Delta robot (H_{max})

Figure 5.3: Minimum and Maximumr eachable height of the Delta robot

5.2.1.2 Maximum Displacement in X–Y Plane

The robot is also characterized by its maximum displacement along the horizontal plane:

- Along X-axis: $-14 \text{ cm} \leq x \leq 14 \text{ cm}$
- Along Y-axis: $-14 \text{ cm} \leq y \leq 14 \text{ cm}$

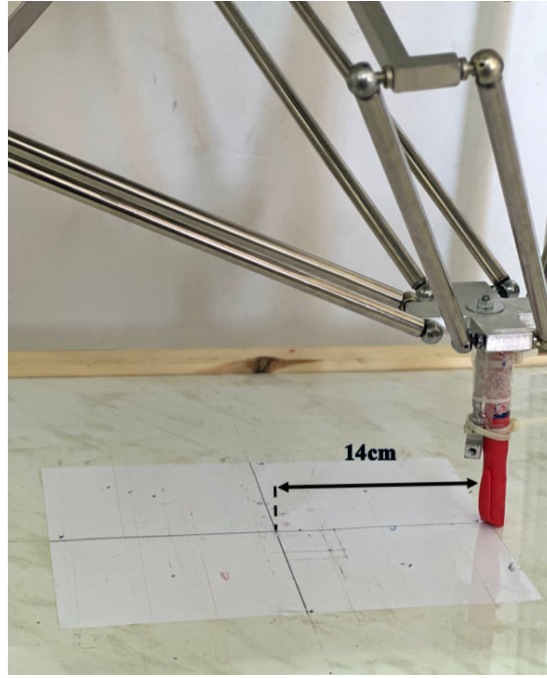


Figure 5.4: Experimental workspace of the robot in the X–Y plane

- **Remark:**

The workspace of the robot is limited due to both actuator and mechanical constraints. In practice, the servomotors do not operate over the full 0° to 180° range, but within a reduced interval to ensure stable operation.

Additionally, the passive joints allow only limited angular motion (approximately $\pm 30^\circ$), which further restricts the workspace and prevents mechanical damage.

5.2.2 Speed of the Delta Robot

According to the HS-5765MH servomotor datasheet, the actuator can spin 60° in 0.13 seconds under no-load conditions. However, in practice, arm length ($L = 50.1$ cm), structural mass, and moving platform weight increase the system's inertia, reducing the actual speed.

The robot's speed is experimentally estimated by recording a known displacement D over a measured time t , using:

$$V = \frac{D}{t}$$

Two speed values are distinguished. The first is the pure mechanical speed, based solely on the code motion parameters (steps = 30, stepDelay = 15 ms):

$$t_{\text{motion}} = 30 \times 15 \text{ ms} = 0.45 \text{ s} \quad \Rightarrow \quad V_{\text{max}} = \frac{0.35}{0.45} \approx 0.78 \text{ m/s}$$

The second is the effective operational speed, measured from the video recording ($D = 35$ cm, $t = 1.48$ s), which includes mechanical inertia and the programmed stabilization delays:

$$V_{\text{effective}} = \frac{0.35}{1.48} \approx 0.24 \text{ m/s}$$

The difference between the two values reflects the real impact of structural load and the stabilization delays intentionally integrated into the control code to prevent inertia-induced oscillations.

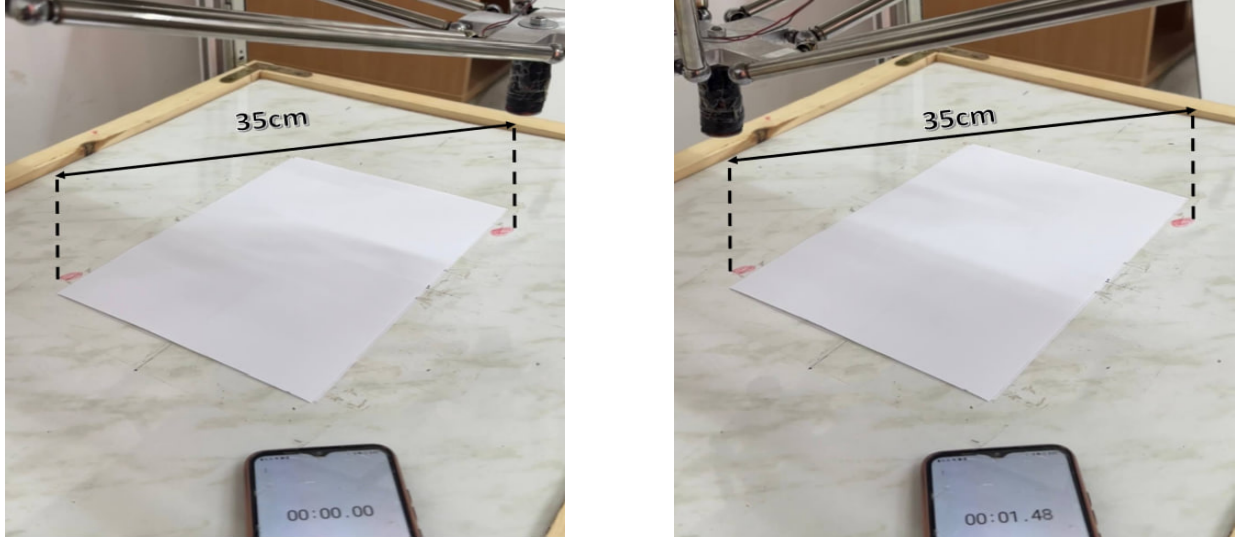


Figure 5.5: Determination of the robot speed using video analysis.

5.2.3 Accuracy

The positioning accuracy of the Delta robot is mainly influenced by the characteristics of the servomotors, the system modeling, and the control signal resolution. The kinematic model does not significantly increase the error under normal operating conditions, but it provides the basis for error propagation analysis.

A key parameter is the deadband of the servo, which represents the minimum variation of the control signal required to produce motion. In this study, a deadband value of $\text{Deadband} \approx 10 \mu\text{s}$ is considered, approximately corresponding to an angular variation of $\Delta\theta \approx 1^\circ$.

This angular error propagates to the end-effector and can be estimated by:

$$\Delta x = L \times \Delta\theta$$

with:

$$L = 50.1 \text{ cm}$$

$$\Delta\theta = 1^\circ \approx 0.017 \text{ rad}$$

$$\Delta x \approx 50.1 \times 0.017 \approx 0.85 \text{ cm}$$

This value represents a theoretical maximum error. In practice, the positioning error is also affected by modeling simplifications, mechanical tolerances, and thermal effects.

Servo heating can increase the error due to reduced response precision, while the long-arm structure of the Delta robot amplifies small angular deviations at the end-effector. The actual positioning error can be measured experimentally using the control interface.

Measured error:

$$\Delta x = 10 - 15\text{mm}$$

5.2.4 The maximum load

The maximum load of the robot is determined based on the actuator torque and the arm length.

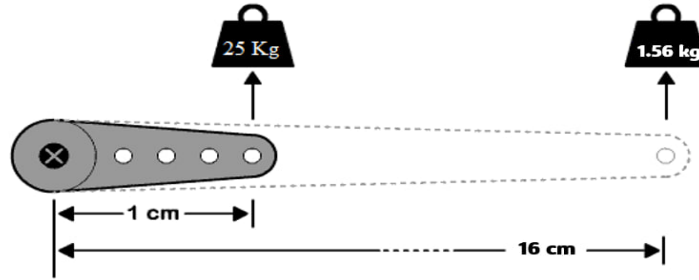


Figure 5.6: Calculating the load for the upper arm.
Each HS-5765MH servomotor provides a maximum torque of:

$$T \approx 25 \text{ kg} \cdot \text{cm}$$

The upper arm length is: $L = 16 \text{ cm}$

The Torque of the upper arm:

$$F = \frac{T}{L} = \frac{25}{16} \approx 1.5625 \text{ kg}$$

Since the robot uses four actuators, each servo is responsible for its own linkages and an equal share of the end effector (platform). The structural mass (M_{servo}) supported by each servo consists of one upper arm, two lower arms, and one-quarter of the platform weight:

- **Upper Arm:** 65 g
- **Lower Arms:** $177.6 \text{ g} \times 2 = 355.2 \text{ g}$
- **Platform Share (End Effector):** $159 \text{ g}/4 = 39.75 \text{ g}$

Because each servo needs to pull 459.95 g of structural weight while generating up to 1562.5 g of force, the system operates well within the safe zone, leaving plenty of capacity for the actual payload.

$$M_{\text{servo}} = 65 + (177.6 \times 2) + \frac{159}{4} \approx 459.95 \text{ g}$$

$$M_{\text{servo}} \approx 459.95 \text{ g}$$

Although the theoretical payload capacity is high, a safety margin is applied to account for dynamic effects, rapid acceleration, and mechanical losses. Therefore, the

practical payload is limited and the robot is tested with a lower load to validate the stability and accuracy of the system:

$$\text{Test load} = 0.552 \text{ kg}$$



Figure 5.7: Measurement of the test load.

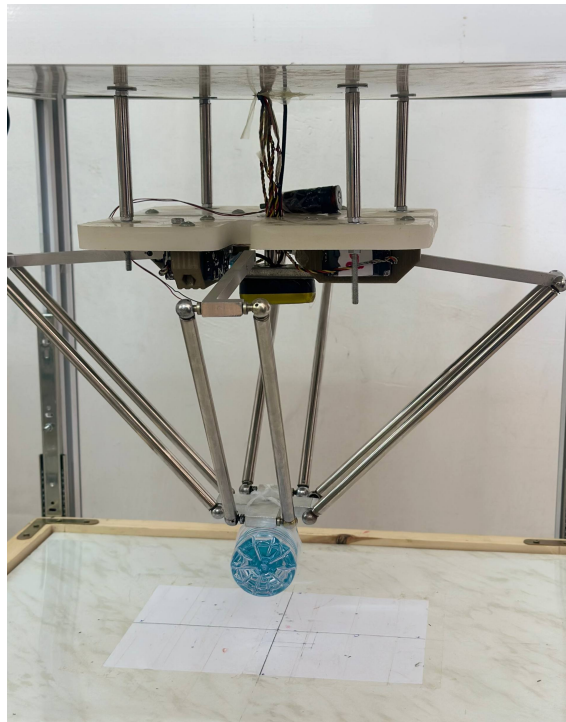


Figure 5.8: Load attached to the end-effector.

5.3 Practical applications of the robot

5.3.1 pick and place

The pick and place operation is one of the most common applications of Delta robots in industrial environments. It consists of picking objects from predefined positions and placing them in a target location.

In this work, the robot is used to manipulate metallic pieces using an electromagnet fixed to the end-effector (nacelle). The pieces are initially arranged on the X–Y plane, and the robot's task is to pick each piece and place it in a storage box. A total of 4 pieces are considered in this experiment. The robot follows a predefined sequence to move each piece from its initial position to the desired location.

The duration of the pick-and-place operation is determined from the programmed delays in the Arduino control sequence. By summing the delays associated with the manipulation of the four pieces, the loop execution time is estimated at:

$$T_{\text{loop}} = 13.8 \text{ s}$$

Including the initial 2 s delay in the setup phase, the total operation time is:

$$T_{\text{total}} = 15.8 \text{ s}$$

This result is consistent with the experimentally measured duration ($\approx 15 \text{ s}$), which corresponds to the total time required to transfer all pieces to the storage box, as shown in Figure. 5.9.

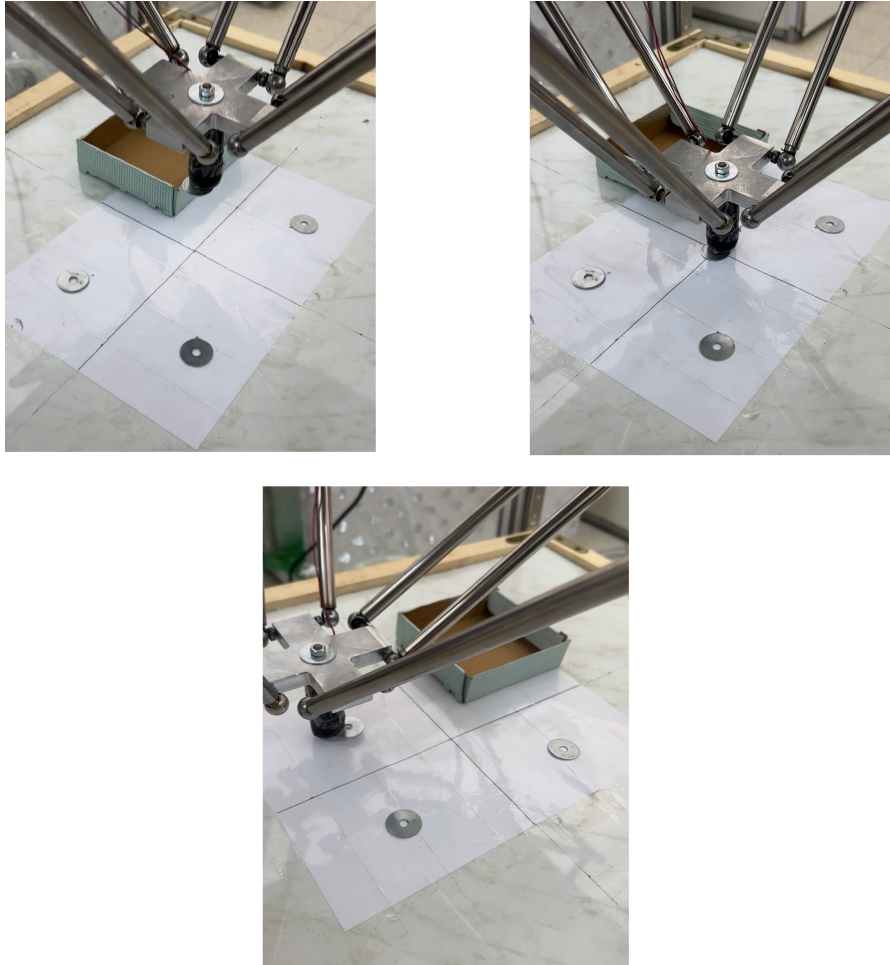


Figure 5.9: pick and place application

5.3.2 Blue Color Tracking Using Vision-Based Control

This application integrates a vision system with the Delta robot to enable automatic detection and manipulation of objects based on color. A webcam is used to capture the workspace, and MATLAB with the Image Processing Toolbox is employed to process the acquired images.

The system focuses on detecting a blue object among other elements using color-based segmentation. Once identified, the centroid of the object is extracted in pixel coordinates. A calibration process is then applied to convert these pixel values into real-world coordinates corresponding to the robot workspace.

Using the obtained position, the inverse kinematic model computes the required joint angles. These values are transmitted to the Arduino controller, which drives the servomotors to move the end-effector toward the target position. An electromagnetic gripper is then activated to pick the object and place it in a predefined storage location.

This sequence demonstrates the ability of the system to detect, locate, and manipulate objects autonomously based on visual information.

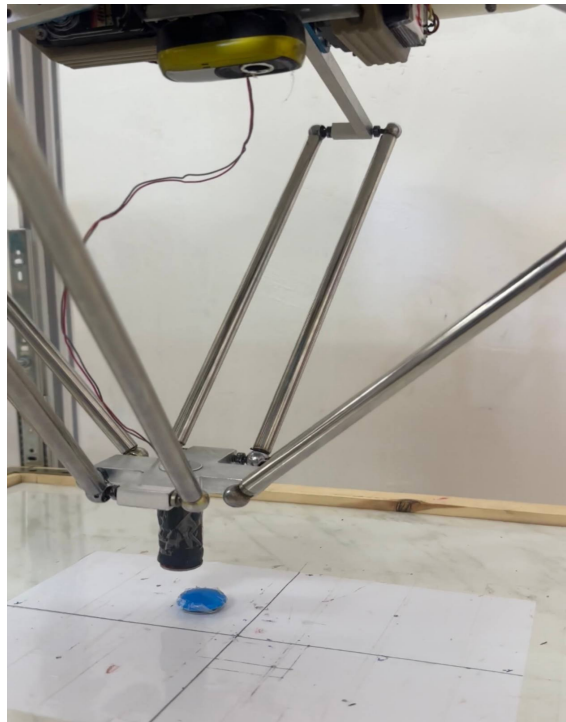


Figure 5.10: Vision-based object detection and sorting system

5.3.3 the delta robot for 2D shape creation

The Delta robot can also be used to generate two-dimensional (2D) shapes by following predefined trajectories. This is achieved by introducing the mathematical equation of the desired shape into the control system.

Such applications are widely used in industry for tasks that require planar motion, such as welding, painting, and drawing on specific materials such as ceramic or marble.

In this work, the robot is programmed to reproduce 2D trajectories based on mathematical models. Among the tested shapes, a four-and three-leaf clover and a square have been selected to demonstrate the capabilities of the system as in figure 5.26

The robot follows the calculated path in the X–Y plane, ensuring a continuous and smooth motion of the end-effector.

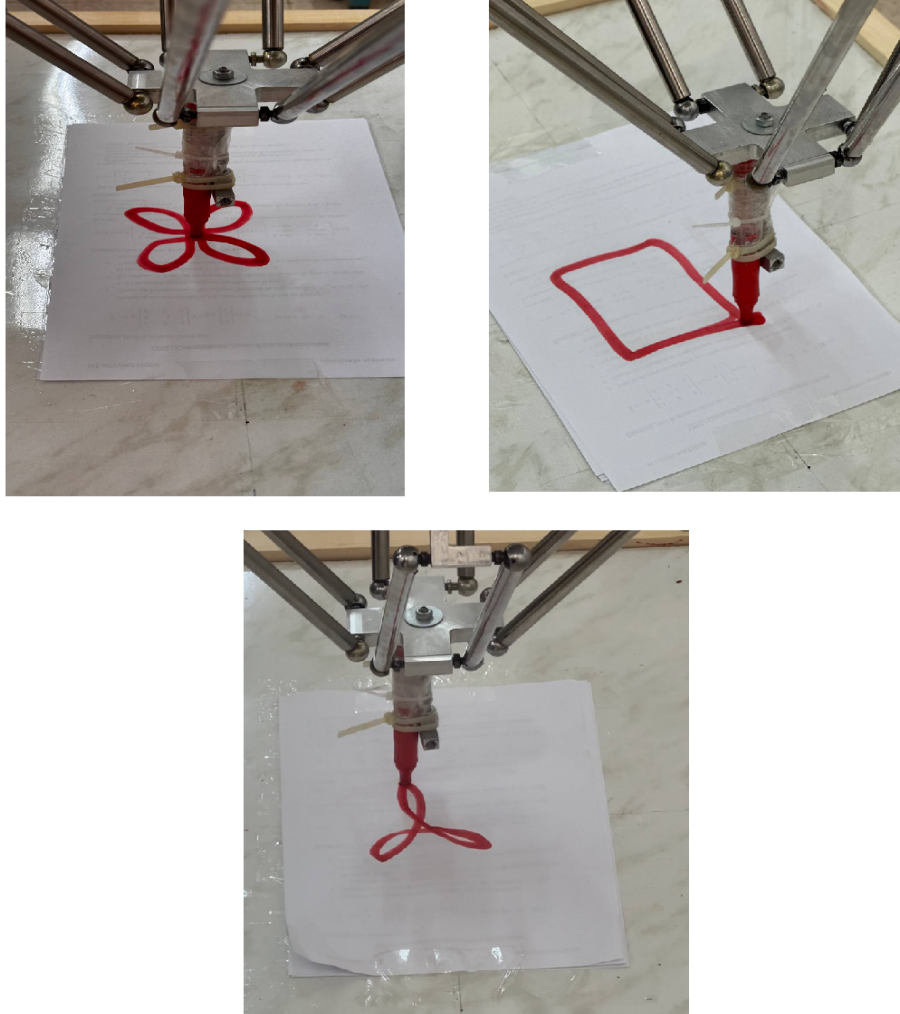


Figure 5.11: 2D shapes generated by the robot (four-and three-leaf clover and a square). factory system for object handling.

5.3.4 Bluetooth-Based Wireless Control

To control the Delta robot wirelessly, an HC-05 Bluetooth module is used. This module enables short-range communication (up to 10 meters) between a mobile device and the Arduino Uno.

The four joint angles ($\theta_1, \theta_2, \theta_3, \theta_4$) are sent directly from an Android application (Serial Bluetooth Terminal) to the Arduino Uno.

The data received are then processed and applied to control the servomotors. This method provides a simple and efficient way to control the robot without wired connections.

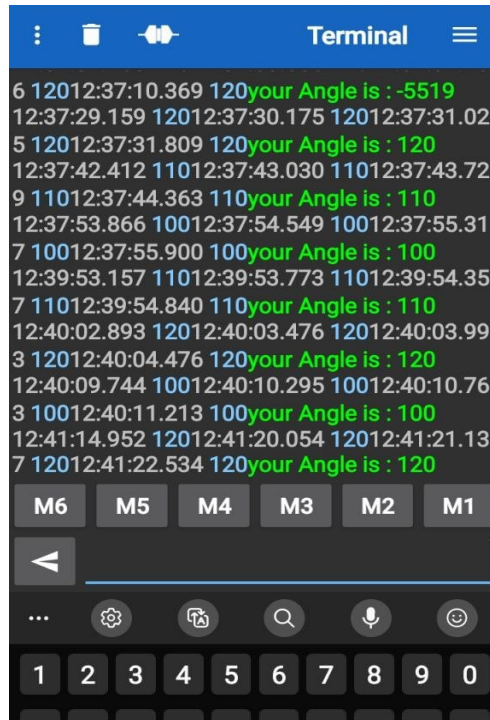


Figure 5.12: Transmission of joint angles via Serial Bluetooth Terminal application

5.4 Conclusion

Several characteristics were presented to evaluate the performance of the developed Delta robot, despite the limited resources available, particularly regarding mechanical components and servomotor precision. This chapter moved from theoretical design to practical implementation and experimental validation. The robot was tested in different tasks, including pick-and-place operations, vision-based object detection and sorting, and 2D shape generation. The obtained results show that the system achieves satisfactory performance within its operating limits. Overall, the robot can be effectively used for multiple basic industrial and educational applications.

General Conclusion

This thesis presented the design, modeling, implementation, and evaluation of a four-arm Delta parallel robot enhanced with a vision-based control system. The objective was to improve the limitations of conventional three-arm Delta robots, particularly in terms of stability, load distribution, and workspace performance, while integrating intelligent visual guidance.

The proposed system combined mechanical design, kinematic modeling, electronic control, and software implementation into a complete and functional robotic platform. The addition of a fourth arm contributed to better load distribution and structural rigidity, resulting in improved stability during operation. The kinematic modeling, including inverse kinematics, enabled accurate computation of joint angles required for precise end-effector positioning.

A vision system based on a webcam and MATLAB Image Processing Toolbox was successfully integrated. Using HSV color segmentation, the system was able to detect a blue object, extract its position in pixel coordinates, and convert it into real-world coordinates through calibration. These coordinates were then used to generate motion commands, demonstrating a complete pipeline from visual perception to robotic actuation.

The communication between MATLAB and Arduino allowed real-time control of the robot through serial transmission of computed joint angles. The control system ensured coordinated motion of the four servomotors, enabling smooth and synchronized operation of the manipulator.

Experimental results demonstrated the effectiveness of the developed system. The robot achieved satisfactory performance in terms of workspace, speed, accuracy, and maximum load. The effective operational speed reached approximately 0.24 m/s, while positioning accuracy ranged between 10 mm and 15 mm. The system was also capable of handling practical loads within safe operating limits. These results validate the feasibility of the proposed design despite limitations related to mechanical tolerances, actuator precision, and structural vibrations.

Several applications were successfully implemented, including pick-and-place operations, vision-based object detection and sorting, and 2D trajectory generation. These applications highlight the versatility of the system and its potential for both industrial and educational use.

Overall, this project represents a comprehensive and practical realization of a vision-guided parallel robot.

It provided valuable experience in integrating multidisciplinary concepts, including robotics, control systems, electronics, and image processing.

As future work, several improvements can be considered. These include enhancing the mechanical precision to reduce positioning errors, optimizing control algorithms for higher speed and stability, and extending the vision system to detect multiple objects or different features. Advanced control strategies based on artificial intelligence, such as neural networks or fuzzy logic, could also be explored. Additionally, extending the system toward 3D applications and more complex tasks would further increase its industrial relevance.

In conclusion, the developed four-arm Delta robot demonstrates improved performance and expanded capabilities compared to traditional designs, confirming the potential of this architecture for high-speed and intelligent robotic applications.

Bibliography

- [1] Bruno Siciliano and Oussama Khatib. *Springer Handbook of Robotics*. Springer, 3 edition, 2023.
- [2] John J. Craig. *Introduction to Robotics: Mechanics and Control*. Pearson, 5 edition, 2021.
- [3] Jean-Pierre Merlet. *Parallel Robots*. Springer, 2 edition, 2022.
- [4] T. Nguyen et al. Advances in parallel manipulators: Design and applications. *Robotics and Computer-Integrated Manufacturing*, 73:102201, 2022.
- [5] Toufik Bentaleb. Contribution à l'étalonnage géométrique des robots industriels. Thèse de magister, Ecole Militaire de Polytechnique, Alger, Algérie, 2006. Filière: robotique, automatique et informatique industrielle.
- [6] OuterReef Technologies. Irb 4400 - fast, compact and versatile industrial robot. <https://www.outerreeftech.com/products/irb-4400-fast-compact-and-versatile-industrial-robot>.
- [7] Motion Control Tips. PI's new high-resolution hexapod for photonics and optics is compact. <https://www.motioncontroltips.com/pis-new-high-resolution-hexapod-for-photonics-and-optics-is-compact/>.
- [8] David Corbel. *Contribution à l'amélioration de la précision des robots parallèles*. Thèse de doctorat, Université Montpellier II - Sciences et Techniques du Languedoc, 2008.
- [9] site ABB. <http://new.abb.com/products/robotics/industrial-robots/irb-360>. date de dernière consultation : 10/06/2015.
- [10] Mohsen Shahinpoor. Kinematics of a parallel-serial (hybrid) manipulator. *Journal of Robotic Systems*, 1992. Division of Engineering and Applied Science, California Institute of Technology, Pasadena, CA.
- [11] ABB Robotics. Yumi - IRB 14000 collaborative robot. <https://new.abb.com/products/robotics/collaborative-robots/yumi>.
- [12] Roland Siegwart, Illah R. Nourbakhsh, and Davide Scaramuzza. *Introduction to Autonomous Mobile Robots*. MIT Press, 2 edition, 2011.

-
- [13] Peter Corke. *Robotics, Vision and Control: Fundamental Algorithms in MATLAB*. Springer, 2011.
 - [14] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics*. MIT Press, 2005.
 - [15] Bruno Siciliano, Lorenzo Sciavicco, Luigi Villani, and Giuseppe Oriolo. *Robotics: Modelling, Planning and Control*. Springer, 2009.
 - [16] Robin R. Murphy. *Introduction to AI Robotics*. MIT Press, 2000.
 - [17] Bruno Siciliano and Oussama Khatib, editors. *Springer Handbook of Robotics*. Springer, 2 edition, 2016.
 - [18] Flowtech. Imi norgren iso 15552 double acting pneumatic cylinder (model: Pra/802080/m/50). Flowtech Product Catalogue.
 - [19] IQS Directory. Leading hydraulic actuator manufacturers. Industrial Quick Search Manufacturer Directory.
 - [20] Festo SE & Co. KG. *Electric Cylinders ESBF: Technical Documentation*. Festo SE & Co. KG, Esslingen, Germany, 2024.
 - [21] Thomson Industries, Inc. *Industrial Linear Actuators: Technical Features and Operation Principles*. Thomson Industries, Inc., Virginia, USA, 2025.
 - [22] Toufik Bentaleb. Contribution à l'étalonnage géométrique des robots industriels. Thèse de magister, filière : robotique, automatique et informatique industrielle, École Militaire de Polytechnique, Alger, Algérie, 2006.
 - [23] Reymond Clavel. Delta, a fast robot with parallel geometry. In *Proceedings of the 18th International Symposium on Industrial Robots*, 1988.
 - [24] Abdelhakim Cherfia. *analyse cinématique des robots parallèles*. Thèse de doctorat, filière : génie mécanique, université de Mentouri Constantine, 2007.
 - [25] R. Clavel. Robots parallèles. *Techniques de l'Ingénieur*, S(7710), July 1994.
 - [26] H. D. McClintock, F. Z. Temel, N. Doshi, J. S. Koh, and R. J. Wood. The millidelta: A high-bandwidth, high-precision, millimeter-scale delta robot. *Science Robotics*, 3(14), 2018.
 - [27] Design, construction, control, and analysis of linear delta robot. Master's thesis, Ohio University. Electronic Thesis and Dissertation.

-
- [28] A. Ayazbay, G. Balabyev, S. Orazaliyeva, A. Kashkynbayev, and P. K. Jamwal. Trajectory planning, kinematics, and experimental validation of a 3d-printed delta robot manipulator. *International Journal of Mechanical Engineering and Robotics Research*, 13(1):113–125, 2024.
- [29] Atom 3D Printers. Neutron - Open Source 3D Printer. MyMiniFactory. Accessed: 2026-06-16.
- [30] Modeling and design of a five degrees-of-freedom delta-like robot for fast pick-and-place applications. *Journal of Mechanisms and Robotics*, 16(7), 2023.
- [31] Juan Sandino, Fernando Vanegas, Frederic Maire, Peter Caccetta, Conrad Sanderson, and Felipe Gonzalez. Uav framework for autonomous onboard navigation and people/object detection in cluttered indoor environments. *Remote Sensing*, 12(20):3386, Oct 2020.
- [32] A. Parmiggiani, E. Ottonello, S. M. Kargar, and I. Palomba. Deltaflex – an additively manufactured delta robot with compliant joints: Virtual prototyping and experimental evaluation. *Journal of Mechanisms and Robotics*, 17, 2024.
- [33] Seyyed Masoud Kargar and Giovanni Berselli. A parametric model for tetra joints applied to compliant delta robot analysis. *International Journal of Intelligent Robotics and Applications*, pages 1–11, 2025.
- [34] Claudio Urrea, Carlos Domínguez, and John Kern. Modeling, design and control of a 4-arm delta parallel manipulator employing type-1 and interval type-2 fuzzy logic-based techniques for precision applications. *Robotics and Autonomous Systems*, 175:104661, 2024.
- [35] Components101. 4n36 optocoupler pinout, features, diagrams & datasheet. <https://components101.com>, 2020. Accessed: 2026-06-16.
- [36] International Rectifier. *IRFZ44N Advanced Power MOSFET*. Infineon Technologies AG, 2001. Datasheet.
- [37] Aliexpress. <https://www.aliexpress.com>. Last accessed: 10/06/2015.
- [38] GeeksforGeeks. Matlab rgb image representation. <https://www.geeksforgeeks.org/software-engineering/matlab-rgb-image-representation/>.

1. شرائح العملاء

المصانع و خطوط الانتاج
خطوط التعبئة والتغليف
مختبرات بحثية و جامعات

2. العلاقة مع العملاء

إنشاء موقع إلكتروني
تدريب تقني للمشغلين
صيانة دورية
متابعة ما بعد البيع

3. القنوات

موقع إلكتروني
التسويق الرقمي
معارض صناعية و تكنولوجية
"linkedin" شبكات مهنية

4. القيمة المقترحة

التقليل الأخطاء البشرية .
تكلفة منخفضة مقارنة
بالمستورد .
فرز الي سريع و دقيق للأجسام
نظام ذكي قابل للتطوير و
التخصيص حسب احتياجات
العمل .
رفع الانتاجية .
صيانة سهلة و مواد محلية .

5. الأنشطة الرئيسية

تصميم وتطوير روبوت دلتا .
الصيانة و الدعم الفني .
برمجة أنظمة الرؤية الحاسوبية.
التصنيع و التجميع و الاختبار .
تطوير خوارزميات الذكاء
الاصطناعي.
التسويق و المبيعات.

6. الموارد الرئيسية

الموارد البشرية: الطلبة أصحاب
المشروع، توظيف فنيين ومهندسين
فريق هندسة ميكانيكية وإلكترونية
برامج Sims, matlab
الموارد المادية: شراء القطع المطلوبة
من السوق المحلية أو عبر الإنترنت

7. الشركاء الرئيسيون

حاضنة الأعمال لجامعة غرداية.
الشركات المصنعة للمكونات .
جامعات ومخابر البحث.
شركات الأمان والسلامة
والاختبار.
موردو القطع الإلكترونية
والميكانيكية .
مطوري البرمجيات ونظم المراقبة
وتحليل البيانات.

9. هيكل التكاليف

1. شراء المكونات الإلكترونية و الميكانيكية .
2. تكاليف التصنيع و التجميع .
3. تكاليف العمال: تشمل تكاليف توظيف ودفع الأجور للعاملين في مشروعنا، مثل المهندسين والفنيين والعمال الذين يشاركون في عملية التصنيع والتجميع والاختبارات والصيانة.
4. تكاليف البحث والتطوير: تشمل هذه التكاليف تكلفة توظيف الخبراء لتطوير التكنولوجيا والبرمجيات المتخصصة.
5. تكاليف التسويق والترويج
6. الصيانة والدعم التقني

8. مصادر الإيرادات

1. من خلال بيع المنتج
2. عقود الصيانة الدورية
3. خدمات التخصيص و التطوير حسب الطلب
4. خدمات الإستشارة



الجمهورية الجزائرية الديمقراطية الشعبية
وزارة التعليم العالي والبحث العلمي
جامعة غرداية
حاضنة أعمال جامعة غرداية



رقم: 136 / ح.أ.ج.غ/2026

شهادة توظيف مشروع مبتكر وفق القرار 008 المعدل والمتمم للقرار 1275

أنا الممضي أسفله، السيد: د/ طالب أحمد نور الدين

مدير حاضنة الأعمال: جامعة غرداية

المقر الاجتماعي/ العنوان: المنطقة العلمية، ص ب 455، غرداية، 47000، الجزائر

بتاريخ: 2025/04/10

رقم علامة الحاضنة: 1004253146

طبيعة المشروع: مؤسسة ناشئة

أشهد أن الطالب(ة) / الطلبة التالية أسماؤهم:

الإسم واللقب	الطور الدراسي	التخصص	الكلية
أفراح مولاي عمار	M2	آلية وأنظمة	كلية العلوم والتكنولوجيا
سندس هامل	M2	آلية وأنظمة	كلية العلوم والتكنولوجيا

تحت إشراف الأستاذ(ة)/الأستاذة التالية أسماؤهم:

الإسم واللقب	الرتبة	التخصص	الكلية
رفيق علجي	أستاذ محاضر ب	أوتوماتيك	كلية العلوم والتكنولوجيا

تم توظيفه على مستوى حاضنة أعمال جامعة غرداية - بمشروع تحت اسم:

Modeling, Design and control of 4 arm Delta parallel manipulator using vision.

خلال السنة الجامعية: 2026/2025

سلمت هذه الشهادة بطلب من المعني للإدلاء بها في حدود ما يسمح به القانون.

حرر في غرداية بتاريخ: 2026/04/26

مدير الحاضنة
وزارة التعليم العالي والبحث العلمي
حاضنة الأعمال
جامعة غرداية
طالب أحمد نور الدين



غرداية في: 2026/07/05

شعبة: آلية
تخصص: آلية وأنظمة

شهادة ترخيص بالتصحيح والإيداع:

أنا الاستاذ: حامد بوخاري.

بصفتي المشرف المسؤول عن تصحيح مذكرة تخرج ماستر المعنونة بـ:

Modeling, Design and Control of a 4-arm Delta Parallel Robot

من انجاز الطلبة:

سندس هامل

أفراح مولاي عمار

التي نوقشت بتاريخ 2026 /06/ 21

أشهد ان الطلبة قد قاموا بالتعديلات والتصحيحات المطلوبة من طرف لجنة المناقشة وقد تم التحقق من ذلك من طرفنا وقد استوفت جميع الشروط المطلوبة.

مصادقة رئيس القسم

امضاء المسؤول عن التصحيح

