Arab Academy for Science, Technology and MaritimeTransport

College of Engineering and Technology

Mechatronics Department

B. Sc. Final Year Project

DESIGN AND CONTROL OF AN AUTONOMOUS TENNIS BALL PICKER

Presented By:

Ahmed Ehab Ahmed Hassan

Hassan Mohamed Ahmed Fouad

Islam Mohamed Hossam El-Din Farouk

Mohamed Hossam Salah Mohamed

Mohamed Moemen Mostafa Mohamed

Mostafa Tarek Seif El-Din Mansour

Supervised By:

Dr. Tamer Ismail

JULY 2024

DECLARATION

We hereby certify that this material, which We now submit for assessment on the program of study leading to the award of Bachelor of Engineering in *Mechatronics* is entirely our own work, that We have exercised reasonable care to ensure that the work is original, and does not to the best of our knowledge breachany law of copyright, and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of ourwork.

Signed: Ahmed Ehab Ahmed Hassan

Registration No.: 19105244

Signed: Hassan Mohamed Ahmed Fouad

Registration No.: 19105118

Signed: Islam Mohamed Hossam El-Din Farouk

Registration No.: 19104549

Signed: Mohamed Hossam Salah Mohamed

Registration No.: 19104447

Signed: Mohamed Moemen Mostafa Mohamed

Registration No.: 19104362

Signed: Mostafa Tarek Seif El-Din Mansour

Registration No.: 19104410

Date: July 2024.

ACKNOWLEDGMENT

Much gratitude and thanks are expressed to **Dr. Tamer Ismail,** who gifted a golden opportunity to do this wonderful project on the topic Autonomous Tennis Ball Picker, who also helped in doing a lot of research and gave knowledge about so many new things.

At last, we would like to extend our heartfelt thanks to our parents because without their help this project would not have been successful. Finally, we would like to thank our dear friends who have been with us all the time.

JULY 2024

ABSTRACT

In this project, we aimed to develop an autonomous tennis ball collector to alleviate the tedious task of manual collection during tennis practice. Tennis, a racket sport played individually or in teams, involves many balls scattered across a court during practice, making collection laborious. To address this, our autonomous/semi-autonomous mobile robot navigates using two independently-actuated wheels and a castor, collecting up to twenty-four balls into a basket via a spinning brush mechanism. The robot features onboard intelligence with cameras, sonars, and an embedded device, allowing for autonomous collection or control via a mobile app. Our design process, detailed in this document, included background research, concept modeling, structural modeling, and preliminary analysis leading to several prototype iterations. Despite challenges and unsatisfactory test results in the verification prototype, the final design shows promise with its structural framing and easy attachment for electrical components. Future enhancements may include global cameras and central image-processing systems for improved localization, path planning, and navigation. Although not all customer requirements were met and the project did not reach its full potential, it establishes a solid foundation for future development. We conclude with recommendations for continued innovation in tennis ball collection technology.

JULY 2024

DEDICATION

In Loving Memory of ***Hussein Hisham*** and ***Mohamed Gamal,***

This project is dedicated to the cherished memory of our dear friends, Hussein Hisham and Mohamed Gamal. Their presence in our lives was a source of joy, wisdom, and unwavering support. Though they may no longer be with us in the physical world, their spirits continue to inspire and guide us in all our endeavors.

Hussein and Mohamed left an indelible mark on our lives, and through this project, we celebrate their enduring legacy. Their friendship, kindness, and remarkable qualities enriched our journeys and brought light to our darkest days. With every challenge we overcome and success we achieve, we honor the profound impact they had on us.

As we embark on this project, we carry with us the love and memories of Hussein and Mohamed, serving as a constant reminder of their vibrant spirits. This dedication stands as a tribute to their lasting influence on our lives and our unwavering commitment to ensuring that their memory lives on in all that we accomplish.

With deep gratitude and love.

JULY 2024

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF REALISTIC CONSTRAINTS

1. Autonomous Tennis Ball Picker

Length	**105 cm**
Width	**38 cm**
Height	**29 cm**
Weight	**35kg**

Table 1: Robot Specs

2. Storing Area

Length	**22.5 cm**
Width	**30 cm**
Height	**10 cm**

Table 2: Storing Area

3. Electronic Tray

Length	**25 cm**
Width	**30 cm**
Height	**9 cm**

Table 3: Electronic Tray

The autonomous tennis ball picker can store up to 24 balls in the storing area

Chapter One

1.0 INTRODUCTION

1.1 Introduction

The Autonomous Tennis Ball Picker project is an innovative venture aiming to revolutionize tennis practice by automating the retrieval of tennis balls. This book details the journey of creating a robotic system that combines robotics, artificial intelligence, and mechanical engineering. The objective is to alleviate the tedious task of ball collection, allowing players to focus on their game and reduce physical strain. Through an in-depth exploration, the book documents the concept, design iterations, technological challenges, and the solutions that shaped the final product. The initiative started with understanding the needs of tennis players and the sport's dynamics, leading to the development of a robot that is both efficient and unobtrusive. The device employs intelligent navigation and collection mechanisms, ensuring a seamless integration into practice sessions. As the narrative unfolds, readers are taken through the rigorous process of prototyping, testing, and refining, highlighting the resilience and creativity required in the face of engineering challenges. The development process encapsulated not only the technical aspects but also the vision of enhancing sports practice with technology. The book discusses the potential transformative impact of such innovations on tennis training, emphasizing improved efficiency and player performance. It reflects on the broader implications for future sports training were technology augments athletic prowess. As the story of the Autonomous Tennis Ball Picker is told, it serves as an inspiring example of the intersection between technology and sports. This project illustrates the potential of engineering and innovation to enhance athletic training and opens the conversation for future advancements in sports technology. It's a testament to the power of persistence, innovation, and the continuous pursuit of improving the human experience in sports

Figure 1: Tennibot

1.2 Overview of The Autonomous Tennis Ball Picker

The Autonomous Tennis Ball Picker project introduces a robotic device aimed at improving tennis training efficiency by autonomously collecting balls from the court. Designed to address the tedious aspect of manual ball retrieval, this innovation significantly reduces players' and coaches' time and effort. The device merges robotics, artificial intelligence, and mechanical engineering to navigate and operate within the tennis court environment effectively. Central features include sophisticated navigation systems for obstacle avoidance, a robust collection mechanism to pick and store tennis balls, and a user-friendly interface allowing for autonomous or manual control. The development process was comprehensive, involving iterative design, prototyping, and rigorous testing to refine functionality and ensure reliability. The impact of this device extends beyond mere convenience; it represents a shift in how technology can be leveraged to enhance athletic training and performance. By automating ball collection, players can focus more on practice, improving skills without the interruption of collecting balls. The project not only serves tennis professionals but also sets a precedent for future sports-related innovations, highlighting the potential for technology to transform athletic training and efficiency. In essence, the Autonomous Tennis Ball Picker project is a testament to the power of innovation in sports technology, promising a more efficient, focused, and productive future for tennis training and beyond.

1.3 Problem Statement

The problem statement for the Autonomous Tennis Ball Picker project addresses the core issue it aims to solve within the context of tennis training and practice sessions. Here it is outlined: In the realm of tennis, both amateur and professional players spend a considerable amount of time collecting tennis balls during practice sessions. This manual process is not only time-consuming, reducing the actual time available for skill enhancement and training, but it also adds unnecessary physical strain on the players, which can lead to fatigue and reduce the overall effectiveness of the practice session. Coaches, too, are diverted from their primary task of coaching to assist in this mundane task. Furthermore, in professional training settings, the sheer volume of balls used can significantly disrupt the flow of practice, requiring frequent pauses for collection. Traditional ball collection tools, like hoppers and tubes, provide some relief but still require manual effort and do not significantly reduce the time spent on this task. The problem is thus twofold: 1) the need to reduce the time and physical effort spent on ball collection to maximize training efficiency and 2) the need for a solution that can adapt to various training environments and be accessible to a wide range of users, from individual enthusiasts to professional training facilities. There is a clear demand for an innovative solution that can autonomously navigate a tennis court and effectively collect tennis balls, thereby optimizing practice time, reducing physical strain, and enhancing the overall training experience. The

Autonomous Tennis Ball Picker project aims to address these issues, offering a technological advancement that benefits players and coaches alike by revolutionizing the way ball collection is approached in tennis practice sessions.

1.4 Project Description

The Autonomous Tennis Ball Picker is a project aimed at developing a robotic system to automate the collection of tennis balls in tennis practice sessions, enhancing training efficiency and reducing physical strain for players and coaches. This innovative robot employs advanced navigation, detection, and collection mechanisms to operate autonomously within the tennis court environment. Key features include an intelligent navigation system using sensors to avoid obstacles, an efficient mechanism for picking up and storing balls, and a user-friendly interface for easy operation and manual control. The durable design ensures adaptability to various court surfaces and longevity in outdoor conditions. The development process involved rigorous stages of conceptualization, prototyping, and iterative testing, leading to a refined final design. Each phase aimed to enhance the robot's functionality, efficiency, and user interaction. The impact of this project extends beyond convenience, offering a technological leap in sports training. It represents a significant advancement in integrating robotics into athletic practice, promising a future where technology and sports coexist to improve training outcomes and athlete performance. The Autonomous Tennis Ball Picker sets a new benchmark in sports innovation, aiming to revolutionize tennis practice by making it more efficient, focused, and enjoyable.

1.5 Approach to Find an Alternative Design

1.5.1 Goal of The Project

The project goal is to solve the tennis ball collecting problem by designing and implementing an autonomous mobile robot that meets the user requirements.

1.5.2 Achieving System Requirements

Once we have a firm grasp of the system requirements, we consider exactly how we go to meet them. At this stage in the process, we made a brain stormy to generate copious numbers of ideas to solve the problem and select viable approaches out of a great many candidate solutions. Number of solutions conceived that look likely to meet project requirements. To choose which candidate designs to move forward we construct prototype to serve to validate our initial ideas and demonstrate the potential of a given approach, serve to inform our estimates of how difficult the project is likely to be.

1.6 Alternative Design

From literature review and existing solutions. Our needs to design an autonomous / semi-autonomous mobile robot, the robot alternatively collect up to 24 tennis balls, faster than human. So we have to create new solution to meet our needs. In this section we listed the alternative ideas which suggested to help solving the collecting problem. The ideas are a result of open-ended questions, brain stormy and YouTube. We follow approach to get the candidate design which studying the existing systems and solutions which are like balls collecting in its principle, so we study the trash collecting method (garbage), crops collecting method, grass cutting method (lawn mower) and existing methods of collecting the balls.

1.6.1 Collecting Using Vacuum and Suction

The collecting mechanism uses an air to pick up the balls from ground like sweeper in work principal. This idea is type of trash collecting method. This method will collect all existing balls on ground. But this method requires very accurate positioning to work well, high power source and in addition the ground may contain a rubbish and objects other than balls. All these disadvantages do not match our needs.

1.6.2 Collecting Using Two Cylinders with Vertical Actuators

Its principal of work it uses two rotating cylinders attached with two vertical motors to collect the balls to the storage area like garbage. This method collects with high speed, can be used with scattered balls, but it needs a high accurate positioning and has a low collecting speed which does not matches our needs

1.6.3 Throwing Instead of Collecting

The principal of this idea is to have a mechanism to throw the balls to specific area using air pump stead of collect it from ground. This method does not match our needs because it collects the balls from ground taking long time.

1.6.4 Collecting Using Gripper Arm Tool

This idea uses a gripper arm to pick up the balls by catch each ball, then through it to storage area. This method is effective for small amount of balls because it needs a high accuracy of positioning to pick up the ball that will take too much time to make the collection. This alternative does not meet our need.

1.6.5 Collecting Using Multiple Channel Tennis Ball Collector

This idea has a novel with multiple channel tennis ball collector , comprises a pair of parallel acrylic discs joined by four springs, each was drilled a hole at their center from which they can be attached to an axle. To accommodate the tennis balls in between, the two discs are made of resilient material and four springs are attached. As the discs rolling through the tennis balls, the ball is then squeezed in between the discs, it is similar to crops collecting method. This method high accuracy of positioning to pick up the ball, so it does not meet our needs.

1.6.6 Collecting by Jump Over a Ball

The concept of this idea is to use a mechanism with tube shape to pick up the balls. This method will pick up all balls from ground and store them inside the shape itself. This design needs a high accurate positioning and has a low collecting speed which is not meet our needs.

1.7 Conclusion

In conclusion, the Autonomous Tennis Ball Picker project embodies a transformative approach to addressing the manual and time-intensive task of collecting tennis balls in tennis practice sessions. Through innovative design and the application of robotics and artificial intelligence, this project aims to significantly enhance the efficiency and focus of tennis training. As we reflect on the journey and achievements of the project thus far, it is evident that understanding the context and precedents in technology and sports is essential. This understanding will not only ground our work in a broader narrative but also illuminate the path forward. The next chapter, titled "Literature Review," is dedicated to exploring and critically analyzing existing research, technologies, and methodologies related to our project. It will delve into historical approaches to ball collection, review advancements in robotic automation, and draw parallels with similar technological solutions in sports and other industries. By examining the successes and shortcomings of existing technologies, we can better understand the niche our Autonomous Tennis Ball Picker fills and the potential it has to revolutionize tennis training. The literature review will provide a solid foundation for appreciating the innovative aspects of our project and set the stage for discussing how our work not only contributes to but also advances the field of sports technology and robotics. With a comprehensive understanding of the past and present, we can more effectively navigate the future of automated sports training aids.

Chapter Two

2.0 LITERATURE REVIEW

2.1 Literature Review and Existing Solutions

In this section, we present the most important papers published in IEEE/ASME Transactions on Mechatronics, about this subject. And we will present them in terms of designing, applying, function specification and equipment used in each in it.

2.1.1 An Intelligent Tennis Ball Collecting Vehicle Using Smart Phone Touch-Based Interface

Chen presented an intelligent tennis ball collecting vehicle which is an autonomous vehicle equipped with a novel tennis ball collector. The robot has a novel with multiple channel tennis ball collector comprises a pair of parallel acrylic discs joined by four springs; each was drilled a hole at their center from which they can be attached to an axle. To accommodate the tennis balls in between, the two discs are made of resilient material and four springs are attached in Figure (2). As the discs rolling through the tennis balls, the ball is then squeezed in between the two discs as shown in Figure (2) .

Figure 2: Robot a Novel With Multiple Channel Tennis Ball

2.1.2 Robotic Tennis Ball Collector “Tennibot”

H. Eletrabi presented an intelligent tennis ball collecting robot has a multiple sensors and local camera. When the robot detects a ball, the robot moves towards it and picks it up then the robot moves on to the next ball until it collects all balls. Tennibot solution is shown in Figure (3).

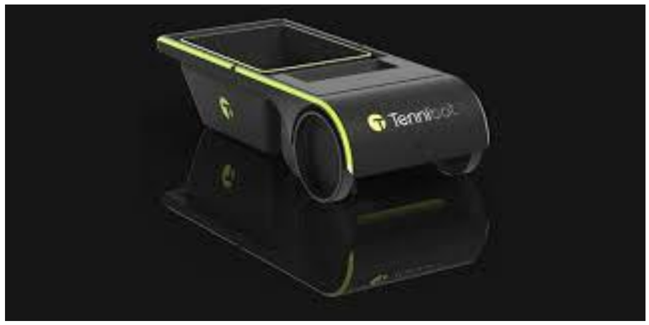


Figure 3: Tennibot Robot

2.1.3 Inwatec Tennis Ball Collector

Inwatec group used a Bosch AHM38G lawn mower, they unmount the cylinder and cut all the blades replace the metal blades (that cuts the grass) with two blades of some soft material (1-2 mm rubber) and mount the new blades with some bolts/nuts as shown in Figure (4).

Figure 4: Inwatec Tennis Ball Collector

2.1.4 Collecting Using Lawn Mower Mechanism

This mechanism collects the balls like grass cutting method (lawn mower) Figure (5) , this method collects the balls with high efficiency and in short time that meet our needs, but the problem of this solution it is not automatic, it requires a human to push it toward balls.

Figure 5: Lawn Mower Mechanism

2.1.5 Collecting Using a Rough Conveyor To Pick Up The Balls

The concept of this idea is to use an oblique conveyor attached to robot, the balls is picked up to storage area by moving a rough conveyor as shown in Figure (6) . [It looks like crops collecting method. this method will collect the balls with high efficiency, but the collecting speed is low which do not meet our need.

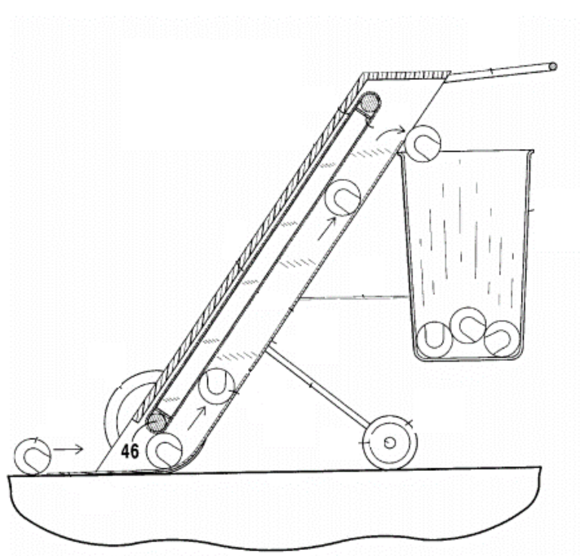


Figure 6: Rough Conveyor Collector

2.1.6 Collecting Using Velcro (Scotch)

This alternative idea uses a scotch to pick up the balls from ground, because the balls are made from cloth strips stitched together with thread and stuffed with feathers like fibers, it has an adhesion property with scotch as shown in Figure (7). This method will collect the balls with high speed. But in the long term this method does not effective, because it destroys the outside shape of balls, and it is required to change the scotch of mechanism from time to time.

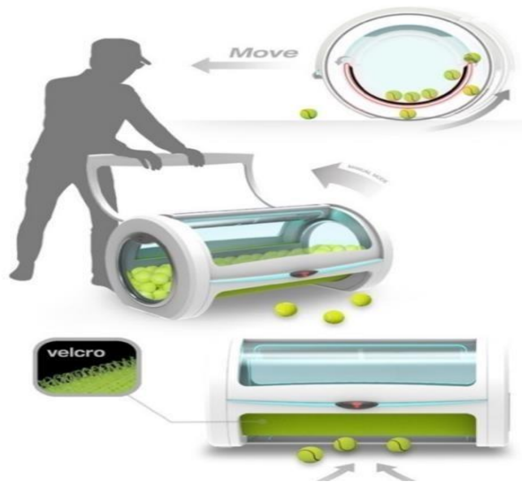


Figure 7: Velcro (Scotch)

2.1.7 Candidate Design

Based on brain storming and alternative designs, we proposed that the existing solutions do not match our needs so we have to create a new design. We take the design idea from two applications, the first one called "tennibot" Figure (8), we take the frame design from it, the movement of robot and the wheels design as shown in Figure (8), other wheels are casters used for robot stability.

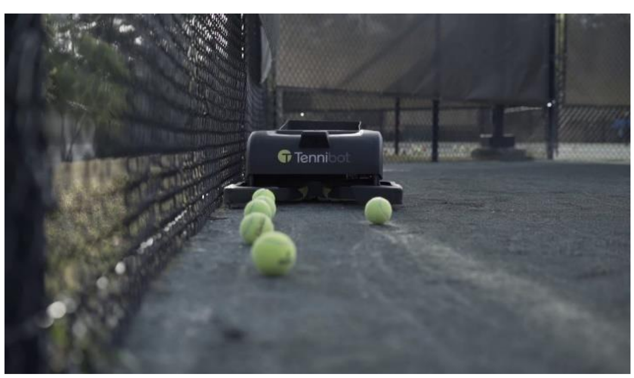


Figure 8: Tennibot Ball Collector Robot

3.0 MECHANICAL DESIGN

3.1 Robot Specifications

Dimensions	
Length	105 cm
Width	38 cm
Height	29 cm
Mass	
Chassis weight	18 kg
Weight with motors	35 kg
Wheel dimensions	
wheel diameter	16.8 cm
Material used	
Mild Steel	
Aluminum	
Plastic	
Wood	

Table 4: Robot Specifications

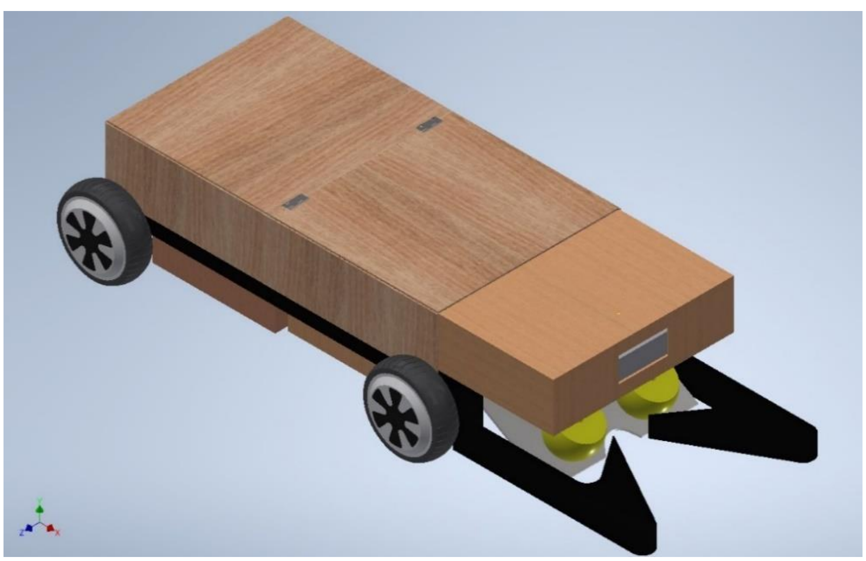


Figure 9: Selected Tennis Ball Collector

3.2 Design requirements

Establishing the project requirements is the initial stage in any project's design process. Stated differently, defining the requirements that your project is meant to fulfill. The next step is to create multiple designs and select the one that best meets your needs. The prerequisites for mechanical design are as follows:

1. Allow 24 balls to be collected in fifteen minutes.
2. A detachable storing compartment that makes emptying possible.
3. Rechargeable battery power sources
4. Ensure end users' safety.
5. User-friendly interface.

3.3 Design Through inventor and Material Selection

The term "Inventor" refers to Autodesk Inventor, a Computer-Aided Design (CAD) application known for its robust 3D modeling, simulation, and visualization tools. It stands out as a leading software in the fields of CAD, Computer-Aided Engineering (CAE), and Computer-Aided Manufacturing (CAM). The software interface of Inventor, as illustrated in Figure (10), provides a sophisticated environment for part design applicable to every component of any machine or project. It is also equipped with advanced features to perform stress, strain, and load analysis for comprehensive machine project assessments.

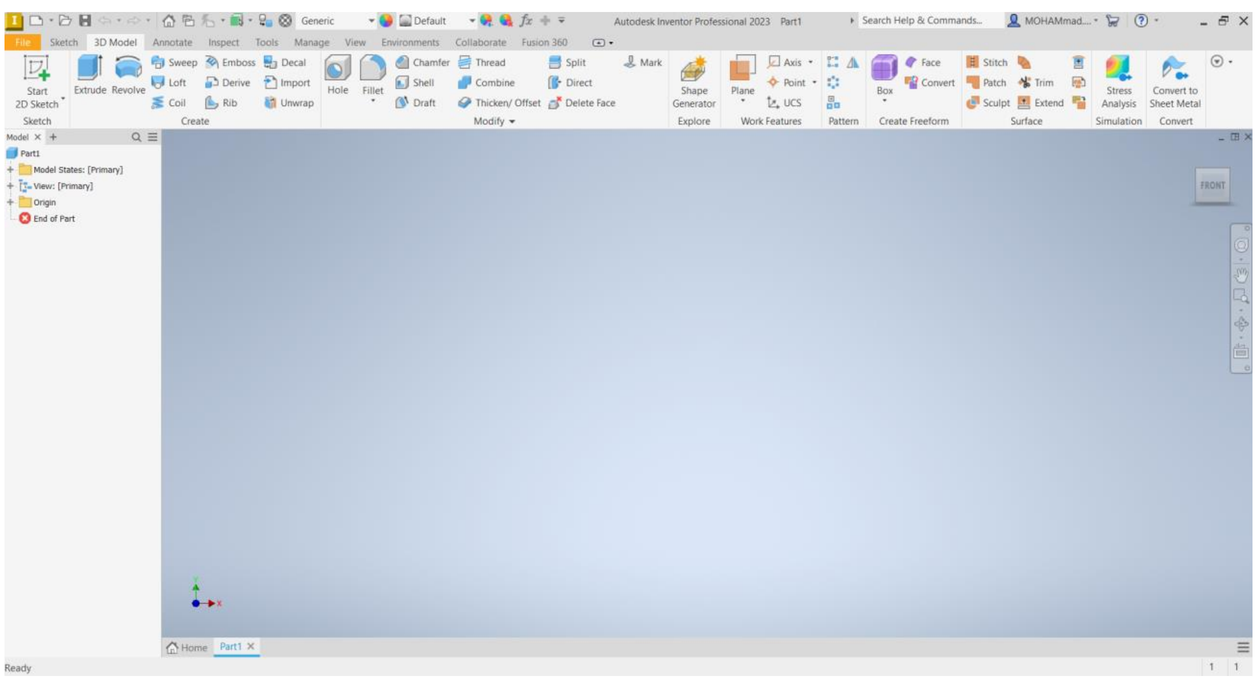


Figure 10: Inventor Software

3.4 Mechanical Frame

The robot mechanical frame consists of two main parts: the mild steel frame forming the body of the robot, and the supporting frame that carries all mechanisms, mechanical, and electrical parts, including the collecting mechanism. The mechanical frame must be sufficiently heavy and made from a material with high damping capacity. As shown in Figure (11), mild steel has been chosen for manufacturing the mechanical frame due to its suitable properties for this application. It has a cross-section of 3x3 cm square with 1mm thickness, which facilitates ease of screwing and serves other functional objectives.

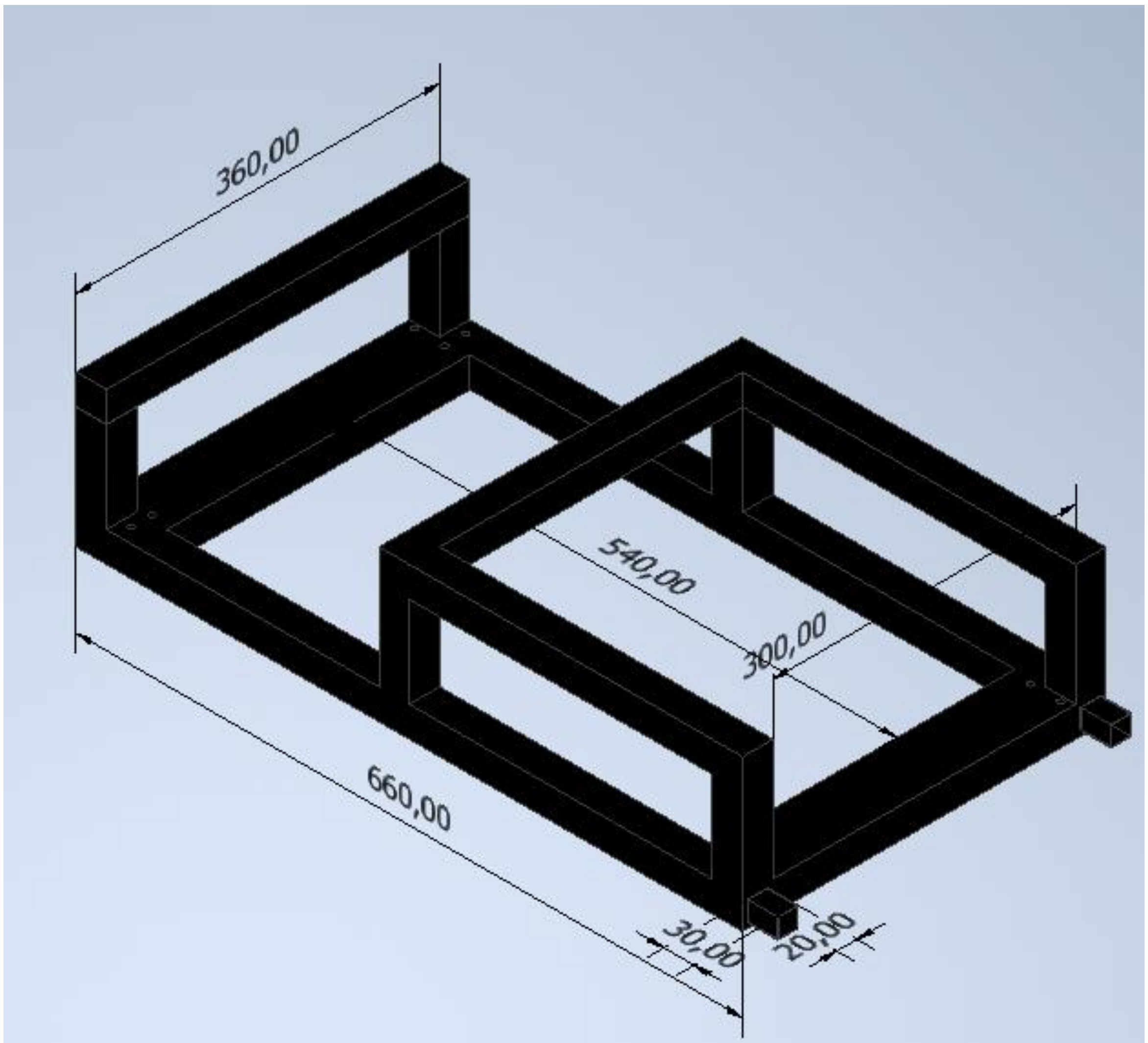


Figure 11: Mechanical Frame

3.5 Collecting Mechanism

This part consists of A launcher mechanism designed for collecting tennis balls typically involves two rollers , each one rotating against the other in order to efficiently gather and then eject the balls to a designated area or container as shown in Figure (12). The mechanism designed with following specifications:
It has a capability to connect with motor to rotate it.
It has elasticity compresses.

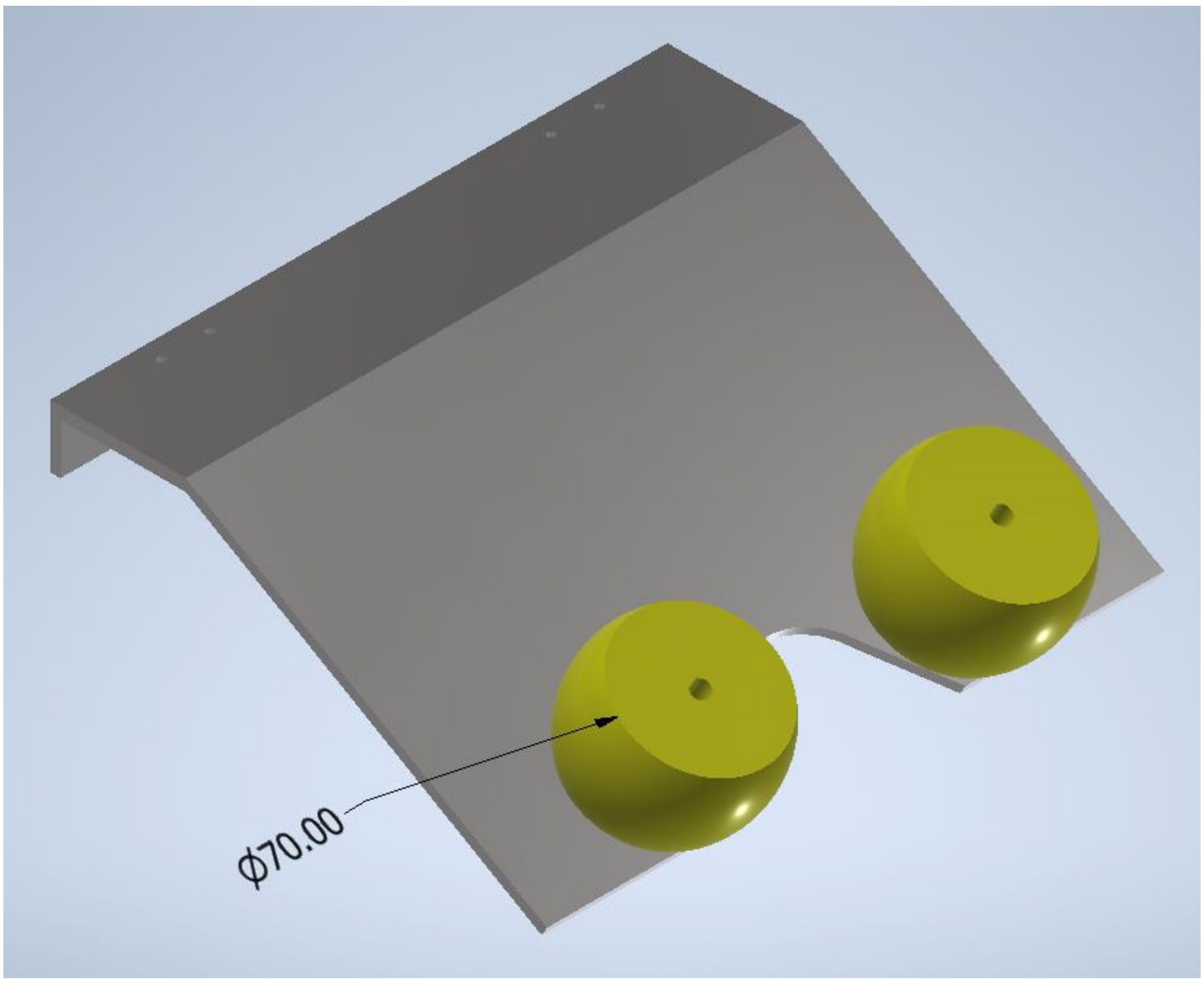


Figure 12: Collecting Mechanism

3.5.1 Ramp Specifications

Dimensions	
Length	12 cm
Width	30 cm
Mass	
Ramp weight	3.5 kg
Rollers Weight	0.2 kg
Motors weight	0.13 kg
Total Mechanism weight	3.8 kg
Materials used	
Stainless Steel	
Plastic	

Table 5: Ramp Specifications

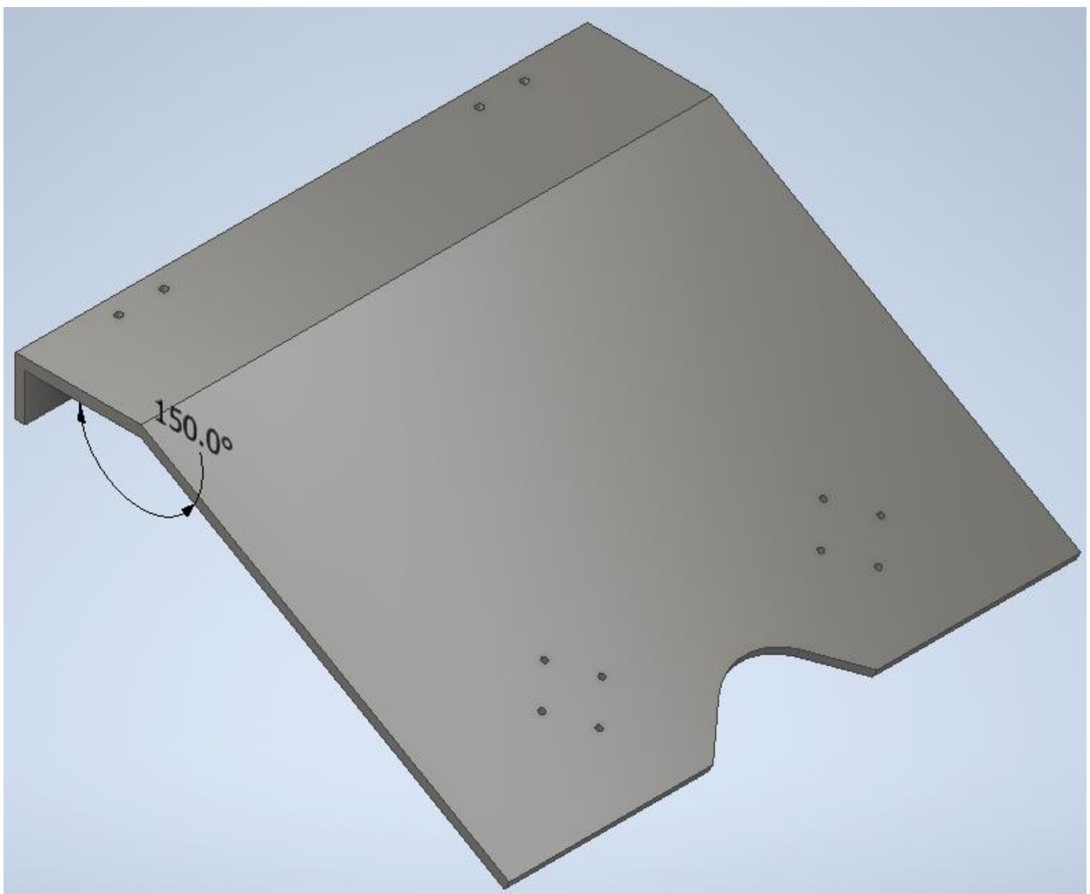


Figure 13: Ramp

3.5.2 Launchers

The rollers crafted from durable 3D-printed material, are strategically placed at the bottom of the ramp to efficiently collect tennis balls. These lightweight yet sturdy rollers are designed to scoop up balls effortlessly as the robot moves across the court. The precision of the 3D printing process ensures that the rollers have an optimal surface texture and durability, enhancing their ability to guide the balls smoothly up the ramp and into the storage area.

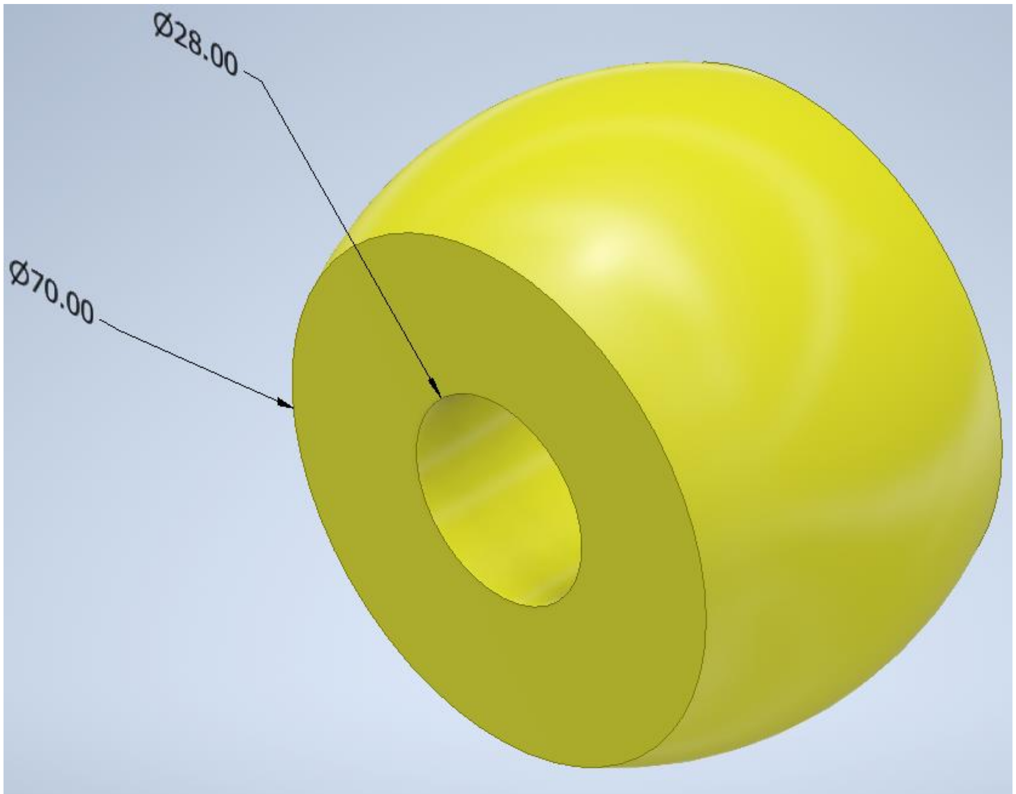


Figure 14: Rollers

3.5.3 Brushless Motor Dimensions

The BLDC A2212 motor datasheet was essential in designing the rollers for our autonomous tennis ball picker. Using the motor's precise dimensions from the provided figure, I crafted the rollers in Autodesk Inventor to ensure a perfect fit and efficient operation. This careful integration optimizes the performance of the roller system, enhancing the overall efficiency of the tennis ball picker.

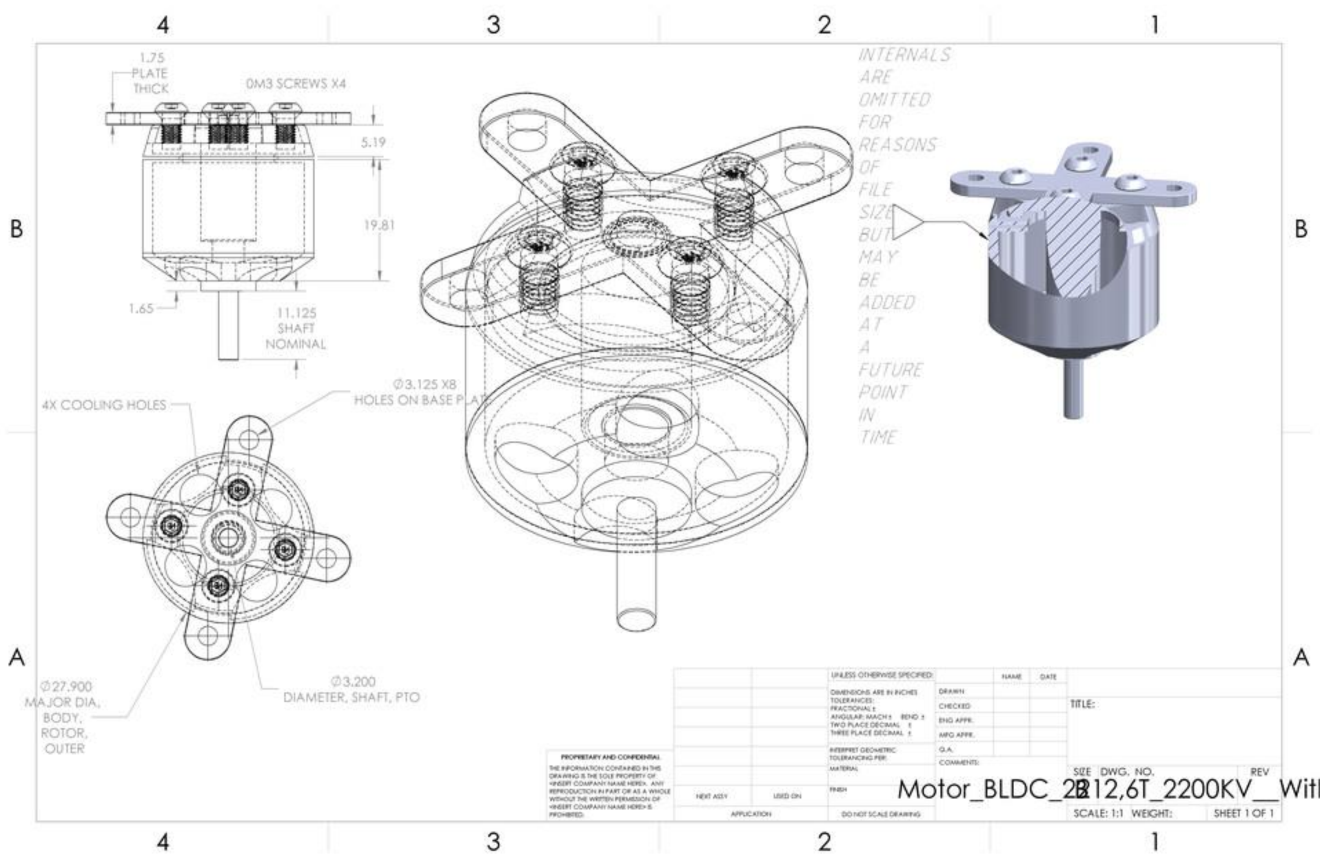


Figure 15: bldc Motor Dimensions

3.6 V-Shaped Arms

The primary function of these V-shaped arms is to collect tennis balls efficiently. When the robot moves forward, these arms can be adjusted to scoop up multiple tennis balls simultaneously. This design minimizes the need for the robot to repeatedly stop and collect one ball at a time, allowing it to cover the tennis court more quickly.

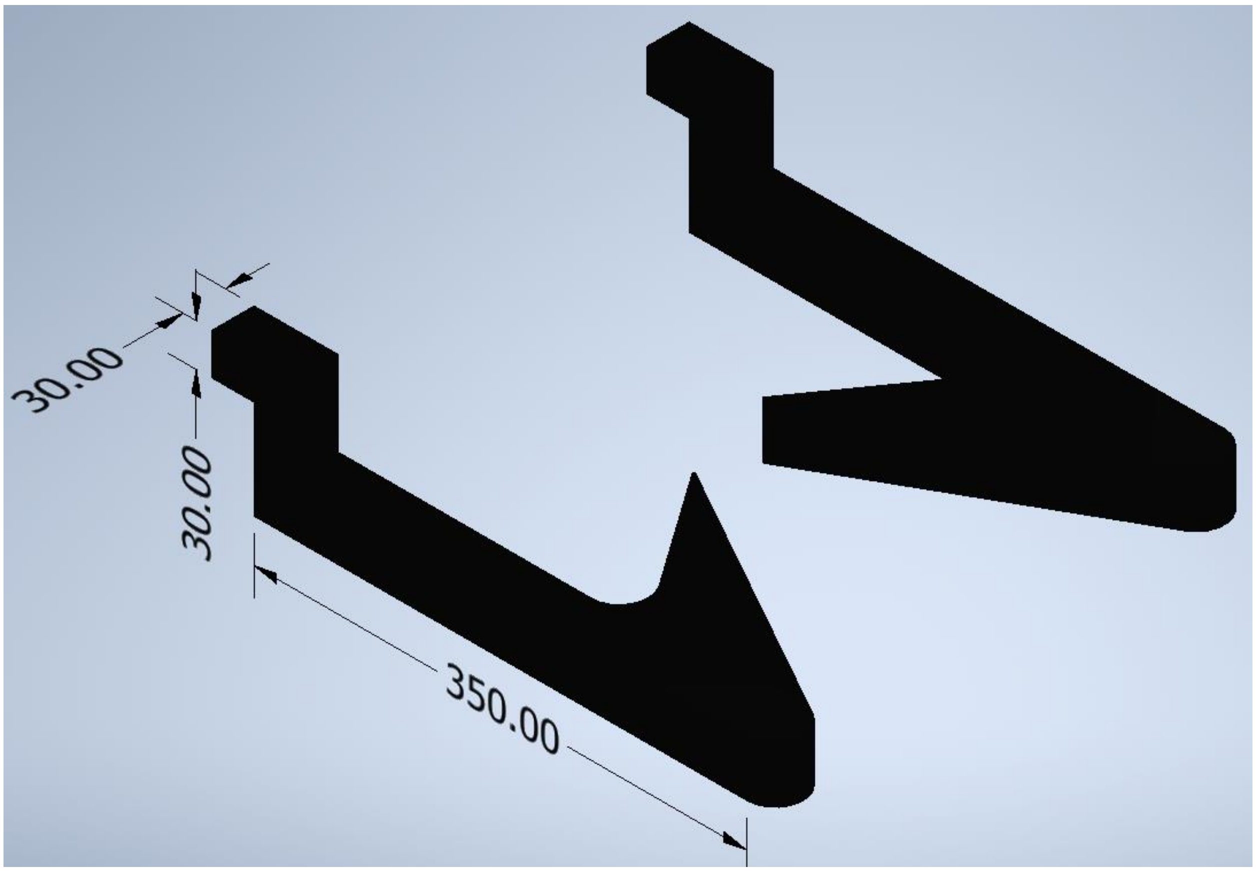


Figure 16: Left and Right V-shaped Arms

3.7 Covers

The Robot features covers crafted from sturdy 5mm thick wood, chosen for its durability and ease of attachment and detachment using screws. Wood was selected for its robust yet lightweight nature, making it ideal for forming protective enclosures around sensitive components like the upper, side, front, and back sections of the robot. These covers provide structural integrity while allowing convenient access for maintenance and repairs. The use of screws ensures secure fastening and quick removal when necessary, facilitating efficient servicing of the Robot without compromising on stability or protection. This design choice ensures that the covers not only enhance the robot's functionality but also maintain a practical and accessible approach to maintenance needs.

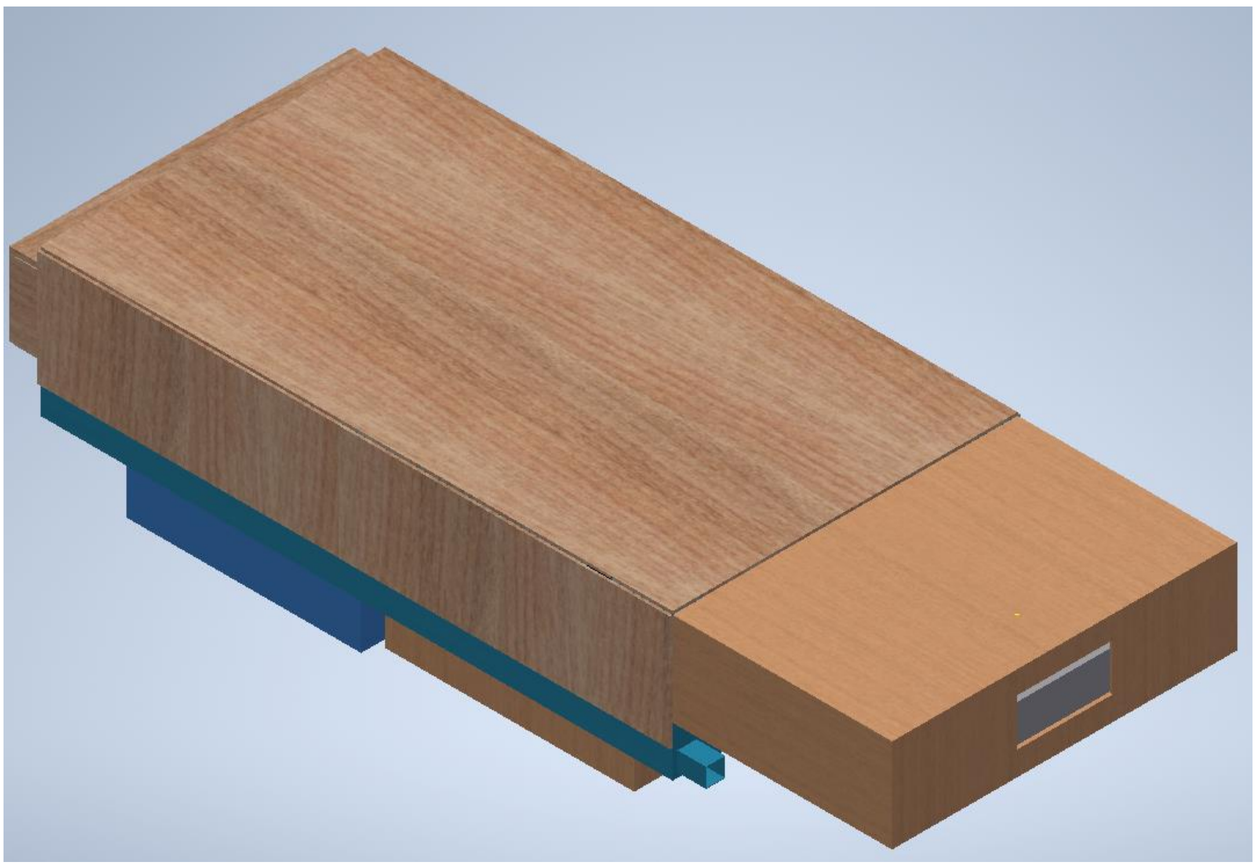


Figure 17: Covers

3.7.1 Front Cover

The cover designed to cover the head of robot, it is made from wood as shown in Figure (18).It is crafted from 5mm thick wood, showcases a blend of elegance and functionality. Mounted on a sturdy internal support, the camera is positioned at the center of the cover, providing optimal visibility and coverage.

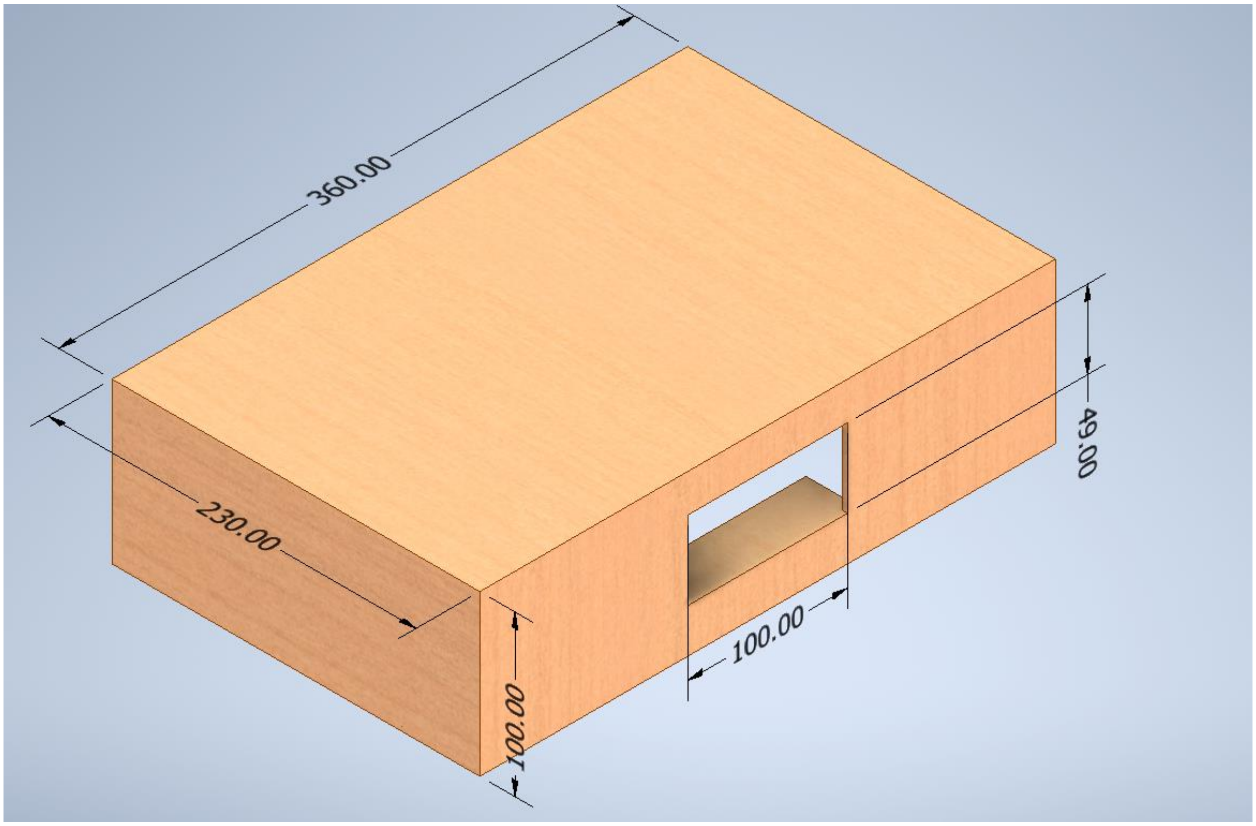


Figure 18: Front Cover

3.7.2 Camera Cover

The camera cover bracket is constructed from durable 3D-printed material, integrates advanced technology with modern design. The camera is securely mounted on a support inside the cover, ensuring stable and precise positioning for optimal functionality. This innovative 3D-printed cover provides a lightweight yet sturdy housing for the camera, protecting it from external elements while maintaining a sleek and streamlined appearance.

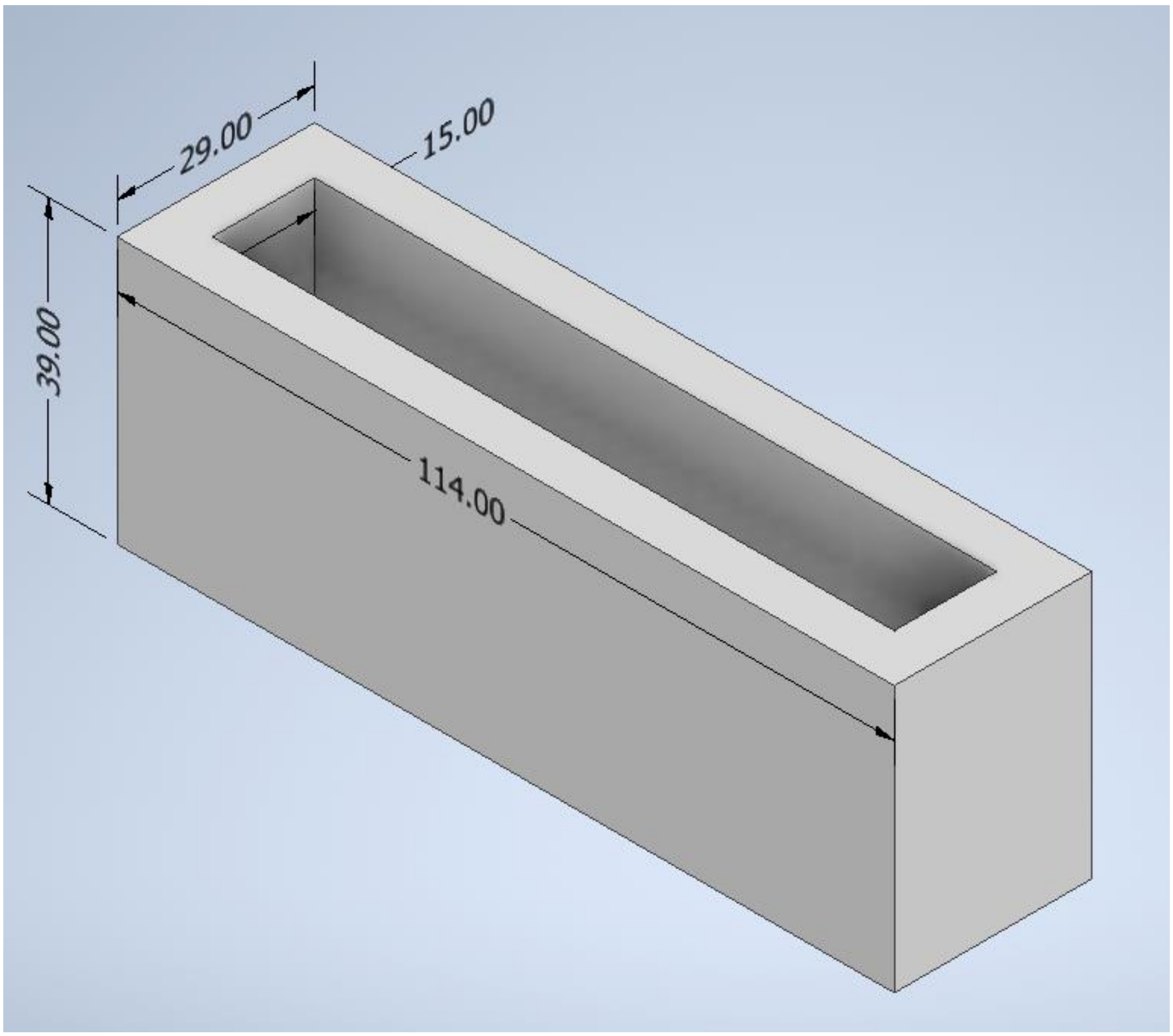


Figure 19: Camera Holder

3.7.3 Back Cover

The back cover of our autonomous tennis ball picker, made from 5mm thick sturdy wood, houses an additional battery and the control board for the back wheels. Designed for functionality and protection, this wooden cover ensures the secure placement of these crucial components. The additional battery provides extended operational time, while the control board manages the rear wheel movements, enhancing the overall maneuverability and efficiency of the robot. This integrated setup ensures that all electronic components are safeguarded from external elements while maintaining optimal performance.

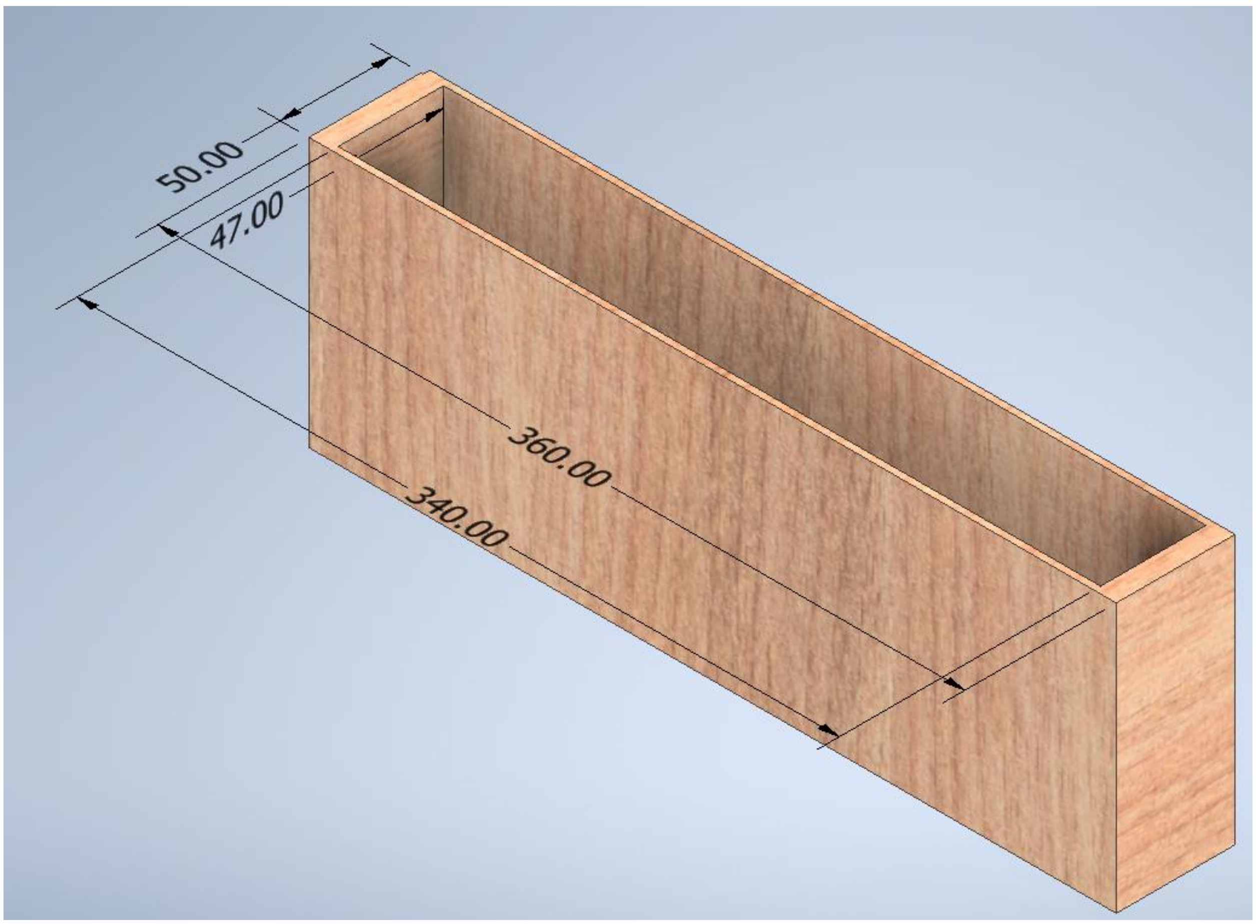


Figure 20: Back Cover

3.6.4 Side Covers

The side covers of our autonomous tennis ball picker, crafted from durable wood, are designed to protect and enclose the left and right sides of the device. These wooden covers provide robust shielding for internal components, ensuring they remain secure and safeguarded from external impacts. Their sturdy construction contributes to the overall durability and reliability of the tennis ball picker, maintaining a streamlined appearance while enhancing the device's structural integrity.

Figure 21: Side Cover

3.8 Wheels

A mechanical component called a wheel makes it easier to move big things, whether they are being transported or utilized to support loads or operate machinery. We are utilizing four hoverboard wheels of the same type to move our autonomous tennis ball picker, ensuring smooth and efficient movement across various surfaces. These wheels are designed to provide stability and agility, allowing the robot to maneuver effortlessly on the tennis court. Their robust construction and innovative design enable the Robot to navigate around obstacles and maintain traction, ensuring reliable performance during ball collection. The use of hoverboard wheels not only enhances the robot's mobility but also contributes to its overall durability and functionality, making it a versatile choice for tennis court automation. See Figure (22).

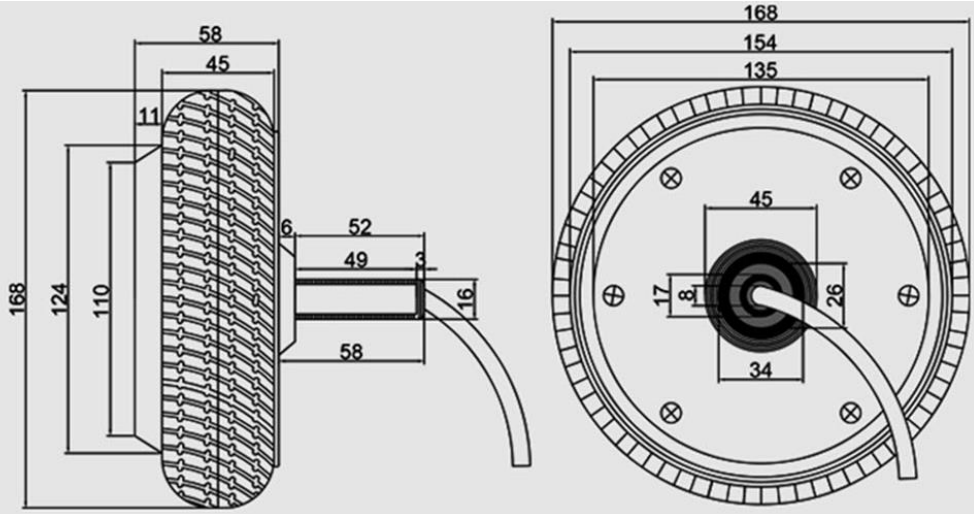
Figure 22: Driving Wheel

The following characteristics can be chosen for the driving wheels, which must be sturdy and rigid enough to support the robot's weight.

- Diameter 168 mm
- Tire width 45 mm
- Weight per wheel 3 kg

3.8.1 Driving Wheel Specs

The wheels of a hoverboard are typically designed for durability and stability, featuring a solid rubber construction to provide smooth rides on various surfaces. Each wheel is equipped with a powerful electric motor that delivers a maximum torque of up to 15 Nm, ensuring efficient acceleration and maneuverability. Hoverboards can generally support a maximum load of 220 pounds (100 kg), making them suitable for a wide range of riders while maintaining performance and safety.



Typical performance	V	A	W	Nm	rpm	Efficiency %
No Load	36.07	0.987	35.60	0.03	624.4	6.2
Eff_Max	36.08	7.601	274.2	3.91	508.2	75.8
Pout_Max	36.07	13.17	475.3	7.59	419.8	70.2
Torque_Max	36.07	13.22	477.1	13.21	58.9	17.1

Figure 23: Driving Wheel Specs

3.8.2 Motor Hub

The motor hub of a hoverboard, into which the wheel is mounted, is crafted from high-quality aluminum cast for superior strength and durability. This robust construction ensures the hub can withstand significant stress and maintain structural integrity under heavy loads and frequent use. The aluminum cast material also offers excellent heat dissipation, preventing overheating and extending the lifespan of the motor. Designed for precision and efficiency, the motor hub seamlessly integrates with the wheel to provide smooth and responsive control, contributing to the overall performance and reliability of the hoverboard.

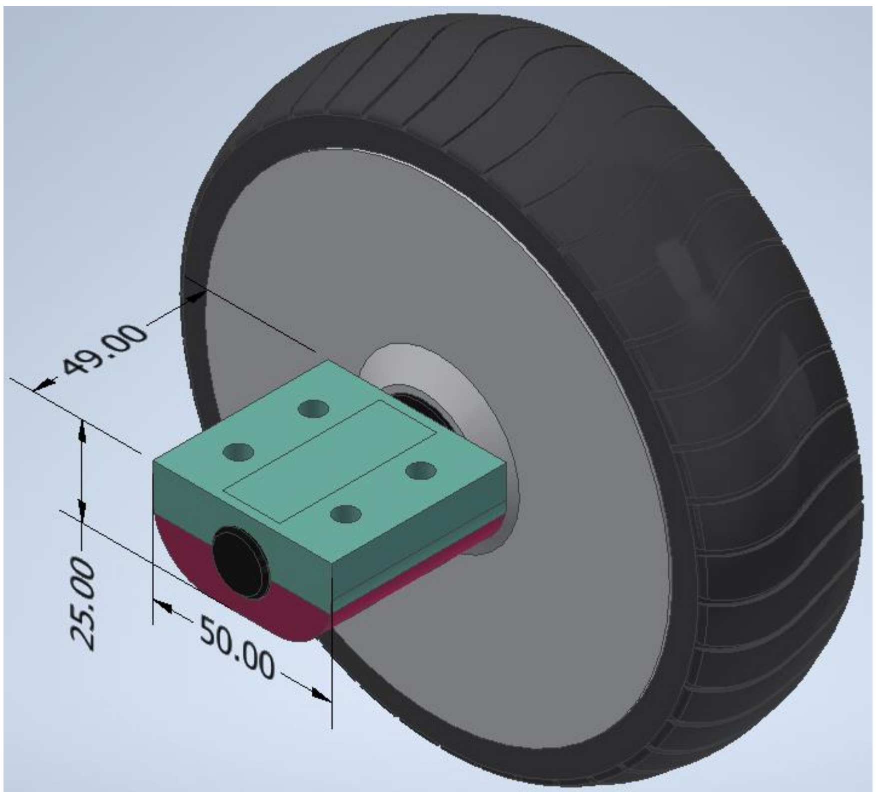


Figure 24: Motor Hub

3.9 Storing Area

This part designed to store the collected ball inside as shown in Figure (25), it has a capacity of twenty balls.

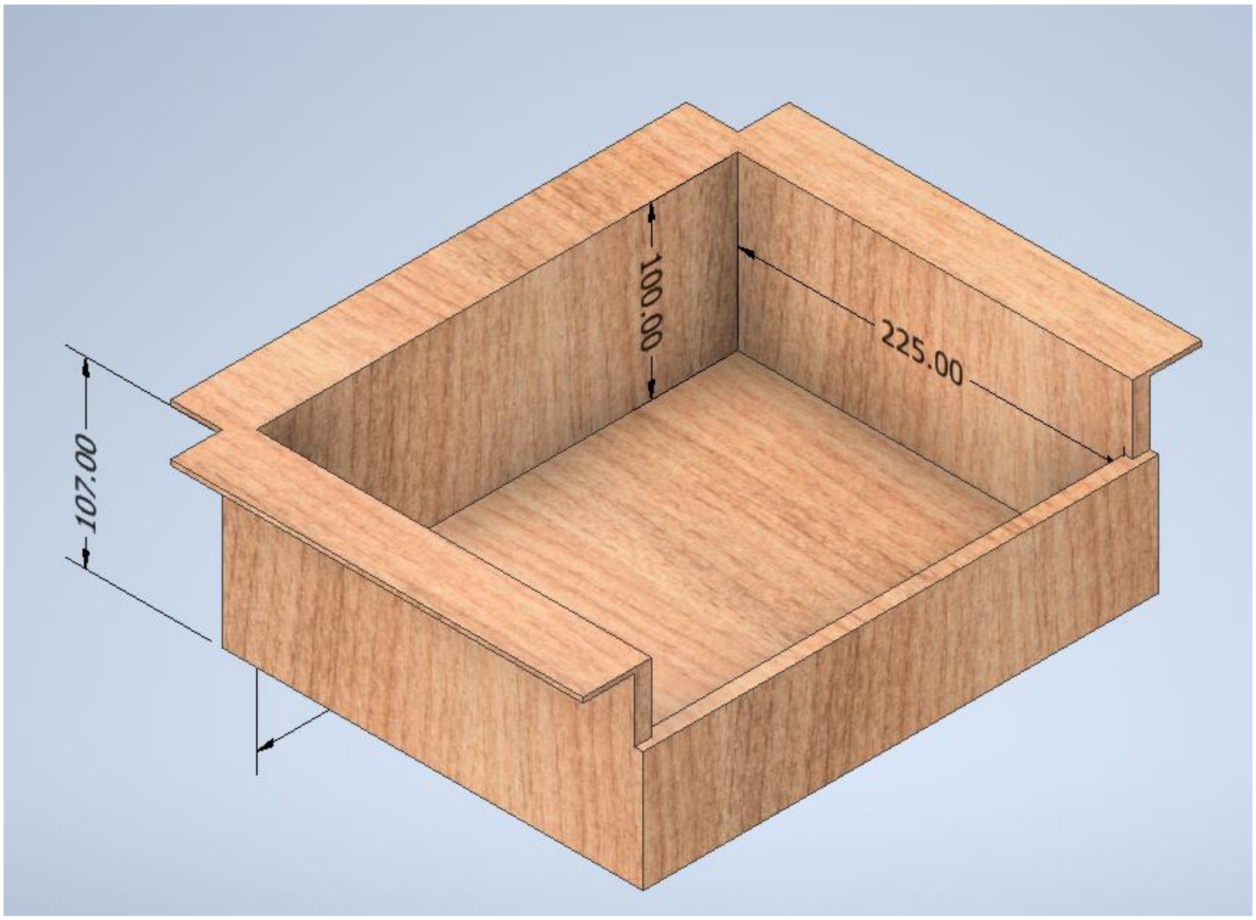


Figure 25: Storing Area

3.9.1 Upper Hinge Door

The hinge door is right above the storing are of the robot in the second half of the upper cover of the Robot, it provides convenient access to claim the collected tennis balls. Designed for easy operation, the hinge allows the door to swing open smoothly, revealing the stored balls inside. Inside the storing area of the Robot, there is a small net securely holding the collected tennis balls. This netting system ensures that the balls remain organized and easily accessible for retrieval. The net is designed to prevent the balls from rolling out while allowing users to conveniently remove them as needed.

Figure 26: Upper Hinge Door

3.10 Electronic Tray

The electronic Tray will be the house of all the electrical components , with hardware connections and it must be able to withstand all what is going to be involved in the hardware connection such as a Raspberry Pi, PCB, motor drivers, and all other components.

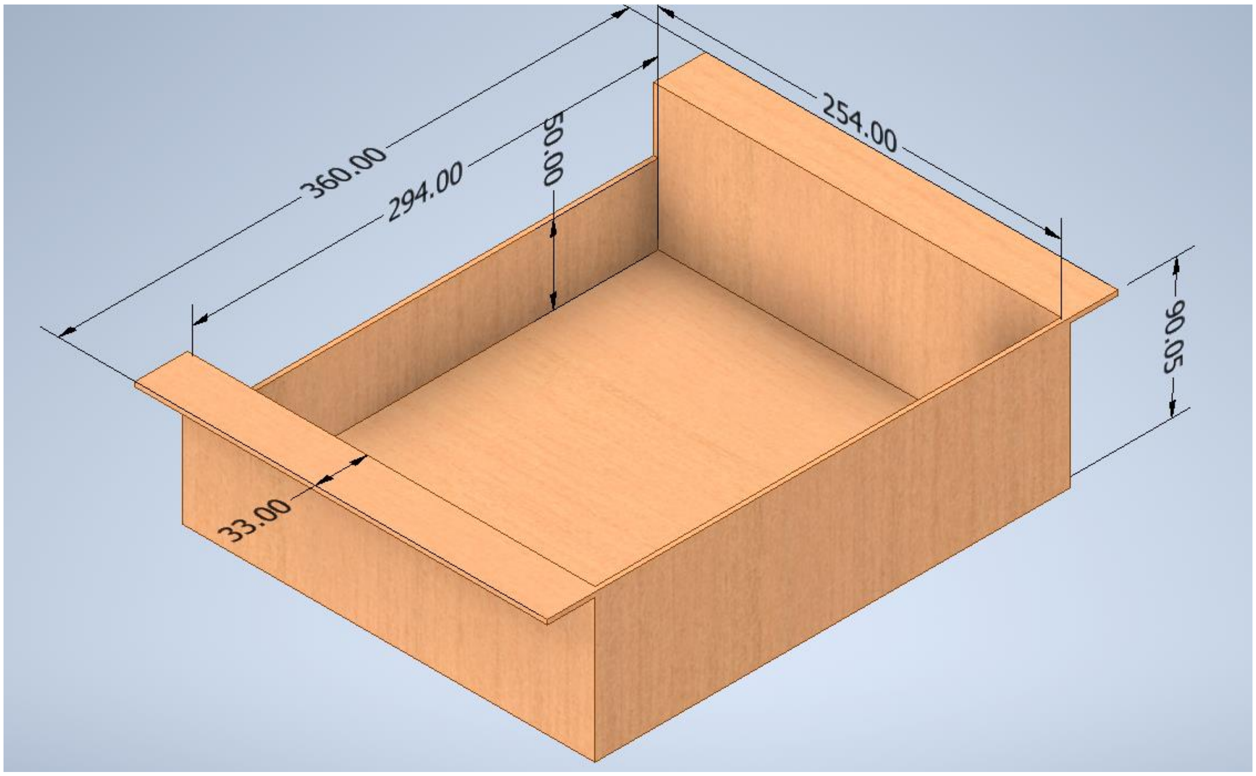


Figure 27: Electrical Components Housing

3.12 Stress Analysis

The stress analysis on a chassis made from welded mild steel involves evaluating the structural response to applied forces. When a force equivalent to 18 kg (approximately 176.58 N) is placed on the edges, the chassis experiences significant stress concentrations at these points. Additionally, a smaller force of 0.2 N is applied to the supporting beams where V-shaped supports are located. The analysis focuses on how these forces distribute across the chassis, identifying potential stress points and areas at risk of deformation or failure. The mild steel's properties, including its yield strength and ductility, are considered to ensure the chassis can withstand these loads without compromising its structural integrity. The welded joints and the overall design must be robust enough to distribute the forces efficiently, minimizing stress concentrations and ensuring safe and reliable performance.

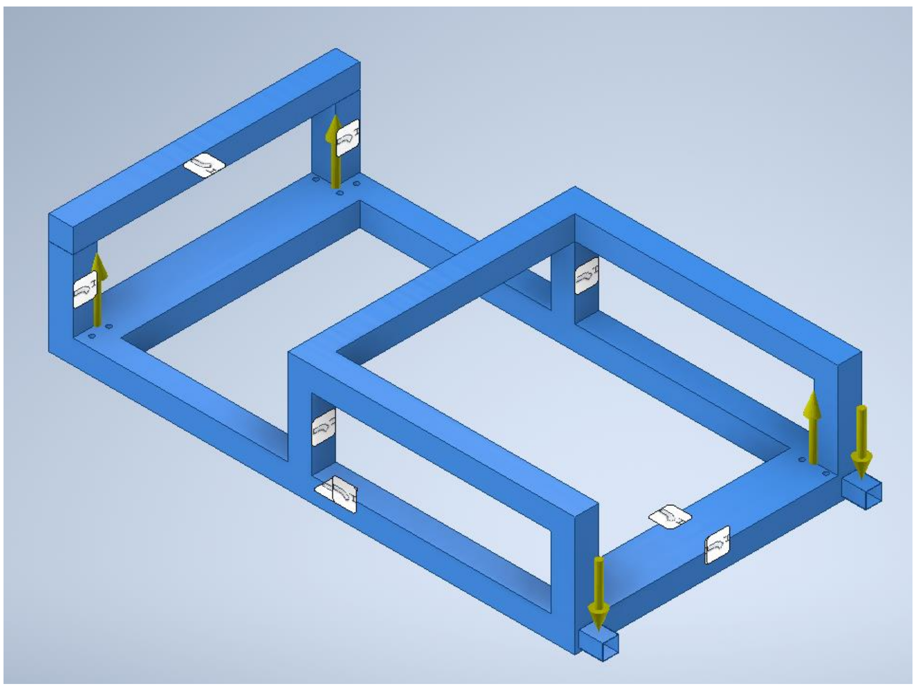


Figure 28: Stress Analysis

3.12.1 Von Misses

Von Mises stress is a criterion for determining if a given material will yield or fracture. It is used for ductile materials such as metals

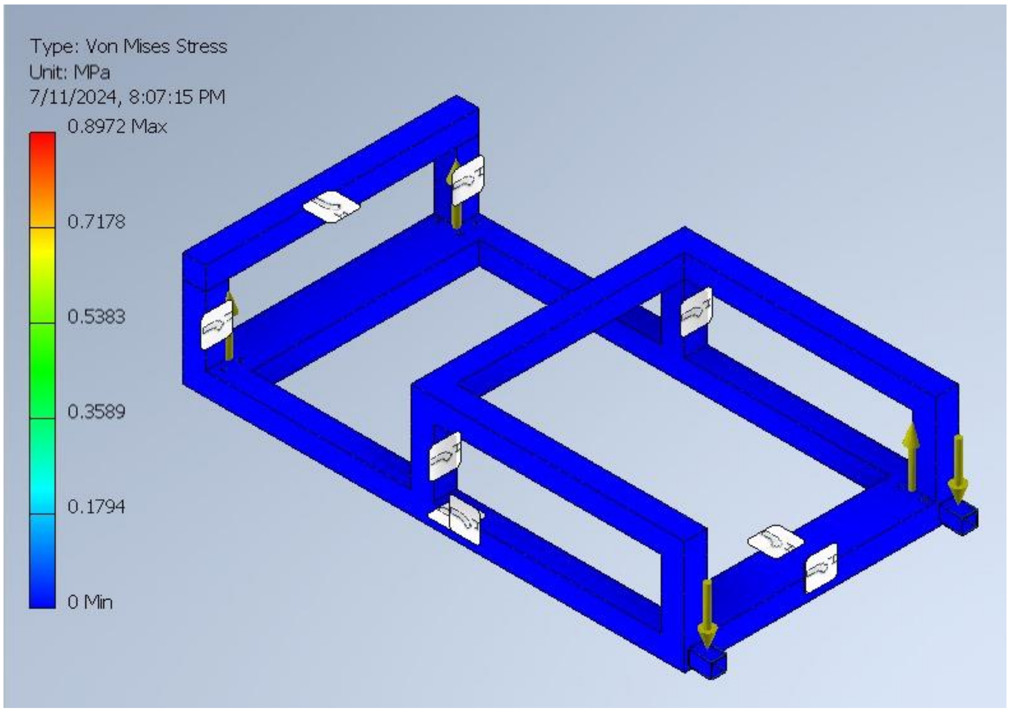


Figure 29: Von Misses

According to the maximum distortion energy theory, yielding occurs when the distortion energy reaches a critical value. This critical value, which is material specific, can be easily obtained through a simple tension test.

The figures below illustrate the curve obtained when studying the strain response of the uniaxial tension of a mild steel beam. These will help us understand why von Mises is important. The description of each emphasized point is as follows:

Elastic Limit: The elastic limit defines the region where energy is not lost during the process of stressing and straining. That is, the processes that do not exceed the elastic limit are reversible. This limit is also called yield stress. Above that limit, the deformations stop being elastic and start being plastic, and the deformation includes an irreversible part. The stress value of the elastic limit is used here as Sy.

Upper yield and lower yield: When mild steel is in the plastic range and reaches a critical point — called the upper yield limit —it will drop quickly to the lower yield limit, from which deformation happens at constant stress until it starts resisting deformation again.

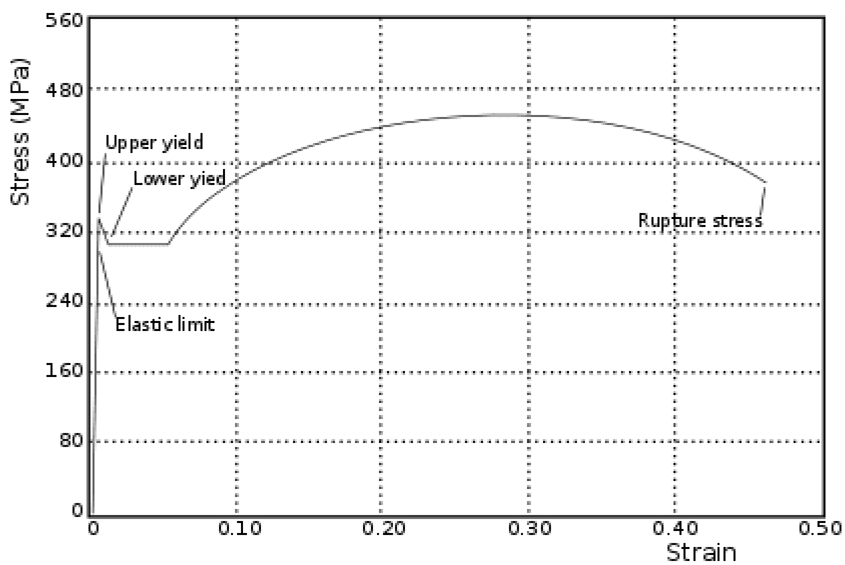


Figure 30: Stress-Strain Diagram of Mild Steel Showing Critical Stages When Under Uni-axial Tension.

3.12.2 Safety Factor

This factor is determined by comparing the material's ultimate strength to the maximum expected stress, typically ranging from 2.5 to 4 for brittle materials. The chosen safety factor ensures that even under unexpected conditions or overloading, the material will have a sufficient margin to prevent sudden and brittle failure, thereby enhancing the overall reliability and safety of the structure.

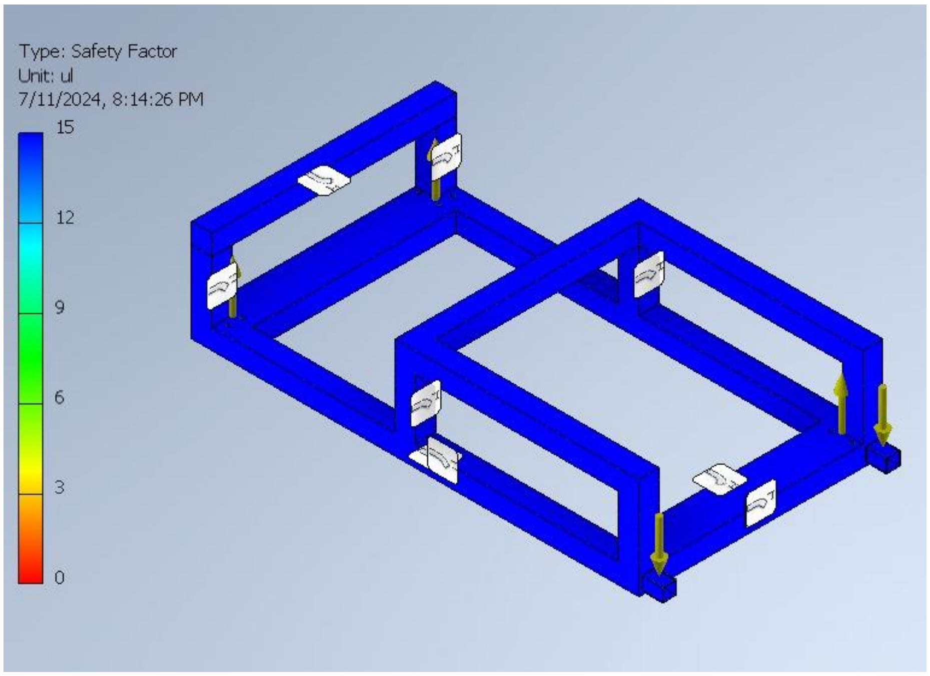


Figure 31: Safety Factor

3.12.3 Deformation

In stress analysis, displacement refers to the deformation or movement of a structure or component under applied loads. It indicates how much and in what direction points within the material have moved from their original positions. Displacement is a critical factor in understanding the behavior of materials under stress, as it helps in identifying areas of high strain and potential failure points.

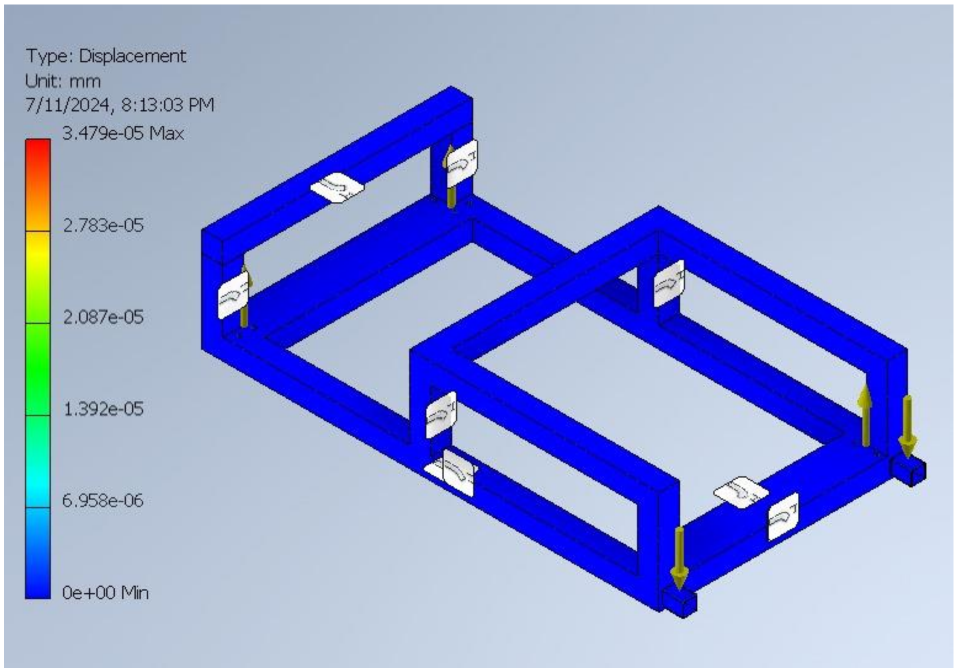


Figure 32: Deformation

Chapter Four

4.0 ELECTRICAL SYSTEM

4.1 Calculations

4.1.1 Hover Board Hub Motor Selection Calculations For Static Analysis

-Force = ma = 35 x 1.5 = 52.5 N

- Friction force= m x g x coefficient of friction = 35 x 9.81 x 0.02 = 6.867 N

- Total force to move the whole body= 52.5 + 6.867 = 59.367 N

- Total torque of motors= Force x radius of tyre = 59.367 x 0.084 =5 Nm

-Torque for 1 wheel = 5/2= 2.5 Nm

4.1.2 Hover Board Hub Motor Selection Calculations For Dynamic Analysis

Efficiency of Gearbox = η

Max RPM = N

Total Inertia = JT

Shaft Inertia = J1

Hub Inertia = J2

Shaft Inertia after Gearbox = Jeff

Resistance torque = TR

Friction = f

Coefficient of friction = μ Normal reaction assumption = R Length of shaft = L

Density of steel = ρ = 7800 kg/mm^3 Radius of Wheel = r

Internal and External Radius of Hub = Ri & Ro

Motor Torque = Tm

Requirements

Calculations

1) $Tm = TR + JT\ddot{\theta}$

$J1 = 0.5 * \rho * \text{pi} * L * r^4$ [L=45 mm, r=33.5 mm, ρ=7800 kg/mm^3]

$= 0.5 * 7800 * \text{pi} * 45 * 10^{-3} * (33.5 * 10^{-3})^4 = 6.94 * 10^{-4}$ kgm^2

$Jeff = J1 \div \eta * N^2$

$= 6.94 * 10^{-4} \div (0.4*40^2)$

$= 1.084 * 10^{-6}$ kgm^2

[η=40% from data sheet]

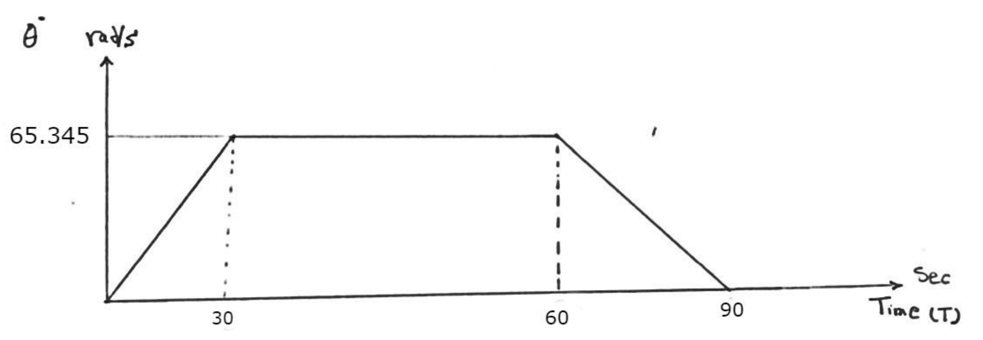
$J2 = 0.5 * \rho * \text{pi} \ (Ro^4 - Ri^4) * L$ [L=52 mm, Ro=22.5mm ,Ri=17mm , ρ= 7800 kgm^2]

$= 0.5 * 7800 * \text{pi} * (52) * 10^{-3} * ((22.5 * 10^{-3})^4 - (17 * 10{-}3)^4))$

$= 1.1007 * 10^{-4}\ kgm^2$

JT= Jeff + J2
$= (1.084 * 10^{-6}) + (1.1007 * 10^{-4}) = 1.1115 * 10^{-4}\ kgm^2$

2) To get Ӫ



$\dot{\Theta}\ max = 2 * pi * 624/60$

$= 65.345$ (rad/sec)

$\ddot{\Theta}\ acc = 65.34/30 = 2.178$ (rad /sec^2)

$\ddot{\Theta}$ const = 0

$\ddot{\Theta}\ dec = -2.178$ (rad/sec^2)

3) The Resistance Force

$f = \mu * R$ ($\mu = 0.05\ for\ Dry\ Concrete$) $= 0.05 * (35 * 9.81) = 17.167$ N

$TR = f * r$

$= 17.167 * 84 * 10^{-3} = 1.442$ Nm

4) The Motor Torque

$Tm = TR + J\ddot{\Theta} = 1.442 + (1.115 * 10^{-4} * 2.178) = 1.45\ Nm$

Torque of Hover board hub motor = 13.21 Nm

4.1.3 BLDC Motor Selection Calculations For Static Analysis

-Force = ma = 0.06 x 1.5 = 0.24 N

- Friction force= m x g x coefficient of friction = 0.06 x 9.81 x 0.1 = 0.0588 N

- Total force to move the whole body= 0.24 + 0.05 = 0.3 N

- Total torque of motors= Force x radius of tyre = 0.3 x .=0.046 = 0.0138 Nm

-Torque for 1 wheel = 0.0138/2= 0.0069 Nm

4.1.4 BLDC Motor Selection Calculations For Dynamic Analysis

Efficiency of Gearbox = η

Max RPM = N

Total Inertia = JT

Shaft Inertia = J1

Hub Inertia = J2

Shaft Inertia after Gearbox = Jeff

Resistance torque = TR

Friction = f

Coefficient of friction = μ Normal reaction assumption = R Length of shaft = L

Density of steel = ρ = 7800 kg/mm^3 Radius of Wheel = r

Internal and External Radius of Hub = Ri & Ro

Requirements

Motor Torque = Tm
Calculations

1) $Tm = TR + JT\ddot{\theta}$

$J1 = 0.5 * \rho * \text{pi} * L * r^4$ [L=25 mm, r=8 mm, ρ=7800 kg/mm^3]

$= 0.5 * 7800 * \text{pi} * 25 * 10^{-3} * (8 * 10^{-3})^4 = 1.25 * 10^{-6}$ kgm^2

$Jeff = J1 \div \eta * N^2$

$= 1.25 * 10^{-6} \div (0.25*20^2)$

$= 1.25 * 10^{-8}$ kgm^2

[η=25% from data sheet]

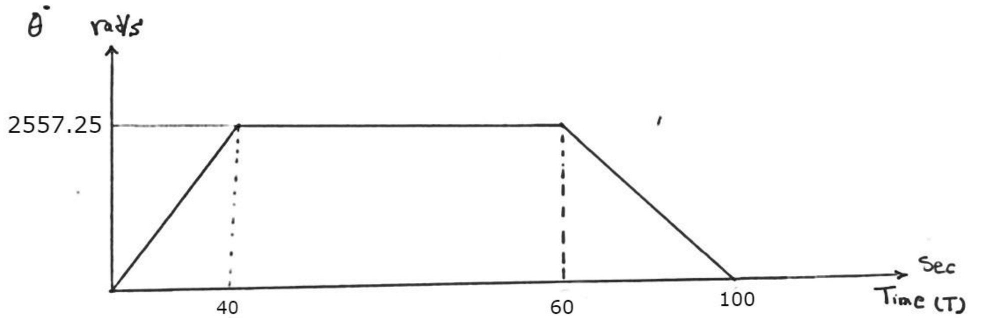
$J2 = 0.5 * \rho *$ pi $(Ro^4 - Ri^4) * L$ [L=13.5 mm, Ro=1.58 mm ,Ri=1.3 mm , ρ= 7800 kgm^2]

$= 0.5 * 7800 *$ pi $* (13.5) * 10^{-3} * ((1.58 * 10^{-3})^4 - (1.3 * 10{-}3)^4))$

$= 5.583 * 10^{-10}\ kgm^2$

JT= Jeff + J2
$= (1.25 * 10^{-8}) + (5.583 * 10^{-10}) = 1.305 * 10^{-8}\ kgm^2$

2) To get $\ddot{\Theta}$



$\dot{O}\ max = 2 *$pi $* 24420/60$

= 2557.25 (rad/sec)

$\ddot{\Theta}\ acc = 2557.25 / 40 = 63.93$ (rad /sec^2)

$\ddot{\Theta}$ const = 0

$\ddot{\Theta}\ dec = -63.93$ (rad/sec^2)

3) The Resistance Force

$f = \mu * R = 0.2 * (0.06 * 9.81) = 0.117$ N

$TR = f * r$

$= 0.117 * 0.046 = 5.382 * 10^{-3}$ Nm

4) The Motor Torque

$Tm = TR + J\ddot{\Theta} = 5.382 * 10^{-3} + (1.305 * 10^{-8} * 63.93) = 0.005\ Nm$

Torque of BLDC motor = 0.0935 Nm

4.1.5 Battery Selection For Raspberry Pi 4

-Name of The Battery: Power Bank

-Battery Capacity = 20 Ampere hour

-Current of Raspberry Pi Microcontroller = 3 A

- Operating Time = (20 / 3) x 60 = 400 minutes

4.1.6 Battery Selection For BLDC Motor

-Name of The Battery: Lithium battery

-Battery Capacity = 4.8 Ampere hour

-Current of BLDC motor = 21.5 A

- Operating Time = (4.8 / 21.5) x 60 = 13.3 minutes

4.1.7 Battery Selection For Hover Board Hub Motor

-Name of The Battery: Two wheel balancing car battery

-Battery Capacity = 4.4 Ampere hour

-Current of Hover Board Hub Motor = 13.22 A

-Operating Time = (4.4 / 13.22) x 60 = 20 minutes

4.2 Component Selection

4.2.1 Raspberry Pi 4 B (4 GB)

Single Board Computer, 4x1.8 GHz, 4 GB RAM, 2x USB 2.0, 2x USB 3.0, 40 Pin GPIO, CSI (camera port), DSI (display interface), Gigabit LAN, WLAN, Bluetooth 5.0.

Figure 33: Raspberry Pi 4B

Processor	Broadcom BCM2711, quad-core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
Memory	4GB LPDDR4
Connectivity	2.4 GHz and 5.0 GHz IEEE 802.11b/g/n/ac wireless LAN, Bluetooth 5.0, BLE Gigabit Ethernet 2 × USB 3.0 ports 2 × USB 2.0 ports.
GPIO	Standard 40-pin GPIO header
Input power	5V DC via USB-C connector (minimum 3A1) 5V DC via GPIO header (minimum 3A1) Power over Ethernet (PoE)–enabled (requires separate PoE HAT)

Table 6: Raspberry Pi 4B

4.2.2 ASUS ROG Eye S

Figure 34: ASUS ROG Eye S Camera

Type	ASUS ROG Eye S
Resolution	11.9 MP
Sensor Diagonal	7.05 mm
Pixel size	1.4×1.4 µm
Picture output	RAW12/10/8, COMP8
Focus	Motorized Auto Focus
Lens	Standard (75°) or Wide-angle (120°)
Interfaces	CSI, 200mm FPC included

Table 7: ROG EYE Camera Specs

4.2.3 Brushless Motor 2200KV

This A2212/6T 2200KV Brushless DC Motor for RC Quadcopters Planes Boats Vehicles and DIY Kits. BRUSHLESS DC MOTOR is one of the most popular high-speed brushless motor designed specifically for RC hobbyists. The motor is outrunner type where outside case rotates while inside stays fixed.

Figure 35: Brushless DC Motor

Number of Cells	2-3s LiPo; 6-10 cell NiMh (2V to 12V)
Model Weight	58 g
Max Current	12A/60s
Max Watts	239 W
Load Current	21.5A
Shaft Size	3.17mm
Dimensions	27.5×30 (mm)
Minimum Recommended ESC	18A

Table 8: Brushless DC Motor

4.2.4 ESC 30A Brushless Speed Controller

Brushless Motor Speed Controller ESC 30A is an electronic circuit to vary the speed, direction and possible to act as a dynamic brake, of a brushless motor.

Figure 36: ESC 30A Brushless Speed Controller

Model	Emax BLHeli Series 30A
Burst Current	40A for 10 sec
Constant Current	30 A
BEC	5V / 2A
Suitable Batteries	2~4S
Dimensions (mm)	52 x 26 x 7
Weight (gm)	28
Application	BLDC Motors, RC Planes etc.

Table 9: Brushless Motor Driver

4.2.5 Driving Motors

Figure 37: Hoverboard Wheel

Typical performance	V	A	W	Nm	rpm	Efficiency %
No Load	36.07	0.987	35.60	0.03	624.4	6.2
Eff_Max	36.08	7.601	274.2	3.91	508.2	75.8
Pout_Max	36.07	13.17	475.3	7.59	419.8	70.2
Torque_Max	36.07	13.22	477.1	13.21	58.9	17.1

Table 10: Hoverboard Wheel Specs

4.2.6 Lithium Battery

A two from this rechargeable lithium-ion battery is used to supply the hoverboard board

Figure 38: Lithium-Ion Battery

Capacity	4400mAh
Weight	0.32kg
Voltage	36V
Type	Lithium-ion Battery Rechargeable
Dimensions	135 mm x 85 mm x 58 mm

Table 11: Lithium-Ion Battery

4.2.7 Power Bank

A rechargeable lithium-ion battery is used to supply the Raspberry Pi 4

Figure 39: Lithium-Ion Battery

Capacity	5000mAh
Weight	0.32kg
Voltage	6V
Write	Lithium-ion
Type	Battery Rechargeable
Dimensions	71 mm x 47 mm x 100 mm

Table 12: Lithium-Ion Battery

4.2.8 Arduino Mega 2560

Figure 40: Arduino Mega 2560

Processor	ATmega2560
Memory	256 KB flash memory, 8 KB SRAM, 4 KB EEPROM
Connectivity	UART, I2C, SPI
GPIO	54 digital I/O pins (15 PWM), 16 analog pins
Input power	7-12V DC (recommended)

Table 13: Arduino Mega 2560

4.2.9 Hoverboard (STM32F)

We used two of these control boards to manage the rear wheel movements effectively.

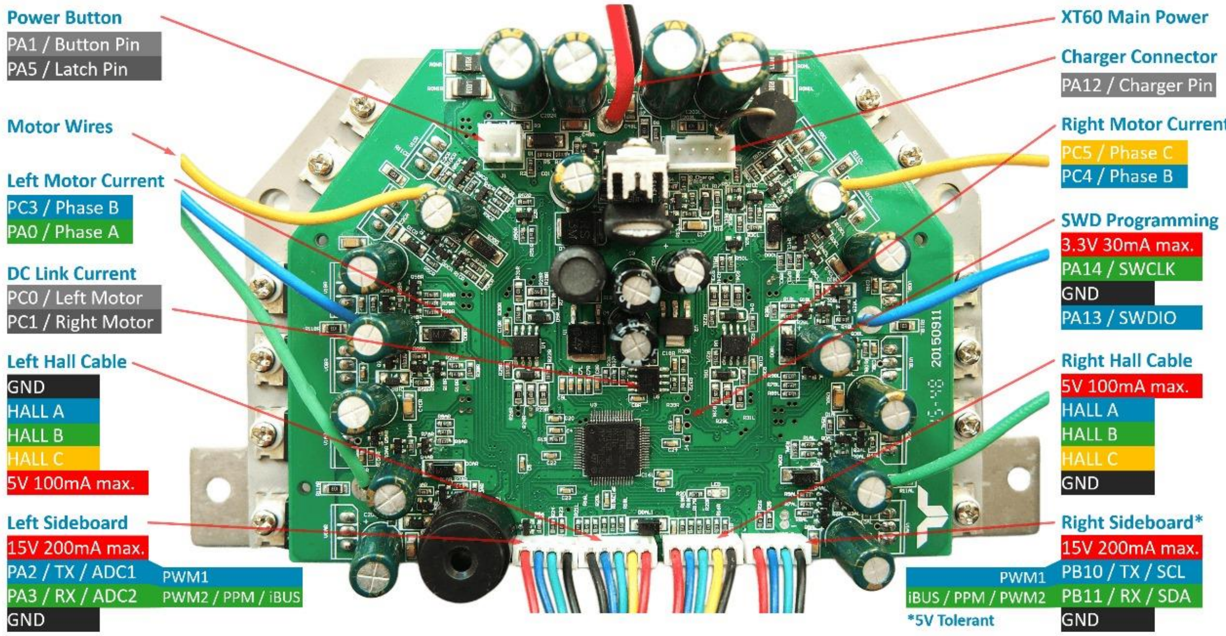


Figure 41: Motor Driver

Table 14: Motor Driver

Processor	Microcontroller or processor unit responsible for executing control algorithms and managing sensor data.
Memory	Storage capacity (RAM, ROM, Flash) for program execution, data storage, and firmware updates.
Connectivity	Options for Bluetooth, Wi-Fi, or other communication protocols used for app control, firmware updates, or data logging.

4.2.10 Battery Case Holder

18650 Battery Case Holder 3 cells Black Plastic Storage Box Case Holder For Battery 3 x 18650 Cell Box to a achieve a total volt of 12v , Without Cover

Figure 42: Battery Case Holder

No. of cells	Hold 3 standard 18650 batteries
Wire length	15 CM
Output	11.1 V
Dimension	76 x 60 x 18 mm
Weight	18 gm
Material	High-quality plastic material

Table 15: Battery Case Holder

4.2.11 Buck Converter

The DC/DC buck module applications the input voltage is higher than the output voltage of the buck field, such as battery, power transformer

Figure 43: Buck Converter

Input voltage range	4.8V (DC)
Output voltage range	1.25-36V (DC)
Output current	0-5 A
Dimension	6.6*3.9*1.8 cm

Table 16: Buck Converter

4.2.12 Li-lon Rechargeable

Figure 44: Li-lon Battery

Battery Type	18650 Rechargeable Battery
Capacity	6800mAh
Weight	0.017kg
Voltage	3.7v
Dimensions	18 mm x 65 mm

Table 17: Li-lon Battery

4.3 Electrical Scheme

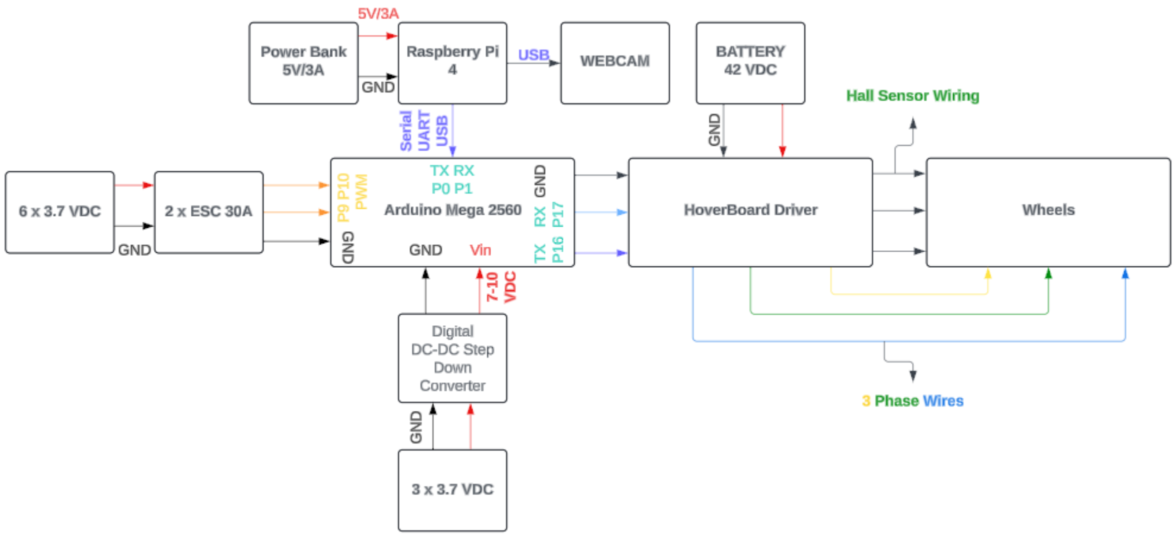


Figure 45: Electrical Scheme

C h a p t e r F i v e

5.0 CONTROL AND SOFTWARE

5.1 PREFACE

This chapter introduces the core aspects of control and software within our project, the Autonomous Tennis Ball Collector. It begins by highlighting the advanced features integrated into our project: Ball Detection and Automated Collection.

It then delves into the system's flowchart and block diagram. These visuals clearly illustrate how the hardware and software components are structured and interact within the system.

Continuing, the chapter explores the implementation of the project design, focusing on the integration and utilization of UART communication between the Raspberry Pi 4 and the Arduino Mega 2560. This setup ensures seamless data exchange and coordination between the components. The Raspberry Pi handles the higher-level processing tasks, while the Arduino Mega 2560 manages the real-time control of the motors and sensors, creating a robust and efficient system for the autonomous tennis ball collector.

5.2 High-Level Control

The high-level control in the Autonomous Tennis Ball Collector project is centered around the strategic management and coordination of the system's operations to ensure efficient and reliable performance. At the core of this control system is the Raspberry Pi 4, which serves as the central processing unit, overseeing the integration of camera data, decision-making algorithms, and communication protocols.
the x-coordinate of the detected ball to the Arduino, which then processes this information to control the movement of the collector.

- **Raspberry Pi 4**

The Raspberry Pi 4 Model B with 8GB of RAM is the latest and most powerful version of the Raspberry Pi single-board computer. It was released in 2019.
Processor and Performance:
- The Raspberry Pi 4 uses a 64-bit quad-core Arm Cortex-A72 CPU running at 1.5GHz, which is a significant upgrade from the Cortex-A53 CPUs used in previous models.

- In benchmarks, the Raspberry Pi 4 8GB offers around 3x the raw CPU performance compared to the Raspberry Pi 3 Model B+.
- The upgraded 8GB of LPDDR4-3200 SDRAM provides ample memory for running larger applications and multitasking.
- This model is capable of delivering desktop-class performance for many everyday computing tasks.

Ports and Connectivity:
- It has dual-band 802.11ac wireless LAN and Bluetooth 5.0 for modern wireless connectivity.
- The Gigabit Ethernet port enables fast wired networking.
- There are 2 USB 3.0 and 2 USB 2.0 ports for connecting peripherals.
- Dual Micro-HDMI ports support 4Kp60 video output to two monitors simultaneously.
- The 40-pin GPIO header provides extensive options for interfacing with external hardware.

Power and Thermal Management:
- The Raspberry Pi 4 8GB can be powered via USB-C, supporting power delivery up to 5V/3A.
- Improved thermal design with a heat sink and optional active cooling fan help maintain stable performance under load.
- Power-over-Ethernet (PoE) support is available via a separate PoE HAT accessory.

Software and Ecosystem:
- It supports a wide range of Linux distributions optimized for the Arm architecture, including Raspberry Pi OS (formerly Raspbian), Ubuntu, Fedora, and more.
- The Raspberry Pi ecosystem includes a vast library of community-developed software, projects, and add-on boards/HATs.
- The platform is well-suited for tasks like media serving, retro gaming, home automation, robotics, and general-purpose computing.

- **ASUS ROG EYE S**

The ASUS ROG Eye S is a specialized camera designed primarily for gaming and streaming applications, offering several advanced features tailored for high-performance use cases. Here's a discussion on its key aspects:

Processor

The ASUS ROG Eye S camera utilizes a high-performance processor to handle real-time video processing and feature extraction tasks. Specific details about the exact processor model used in the ROG Eye S are typically not disclosed by manufacturers. However, it is designed to efficiently process high-resolution video streams and implement sophisticated algorithms for features like depth sensing and facial tracking.

Depth Sensing

Depth sensing capability is a significant feature of the ASUS ROG Eye S, enhancing its functionality beyond basic video capture. Depth sensing allows the camera to perceive the distance of objects within its field of view. This capability is particularly useful in applications such as virtual reality (VR), augmented reality (AR), and gesture recognition, where precise spatial understanding is essential for interaction and immersion.

Connectivity

The ASUS ROG Eye S camera typically connects to devices via a USB interface. This ensures compatibility with a wide range of platforms, including Windows PCs and laptops, making it versatile for gaming setups, streaming rigs, and professional environments. The USB connectivity provides a reliable and high-bandwidth link for transmitting high-definition video and depth data in real time.

Power Consumption

Power consumption is an important consideration for any peripheral device, especially one used intensively in gaming and streaming scenarios. While specific power consumption figures for the ASUS ROG Eye S may vary based on usage and settings, it is designed to be energy-efficient to minimize strain on the host system and ensure prolonged usage without excessive heat generation.

Additional Features

- **High-Quality Video**: Capable of capturing video at high resolutions and frame rates, ensuring clear and smooth footage during gameplay and streaming.
- **Customizable RGB Lighting**: Features RGB lighting that can be synchronized with other ASUS Aura Sync-compatible devices for personalized lighting effects.
- **AI-Powered Features**: Some models of ASUS ROG Eye S cameras may incorporate AI-powered features such as background removal or facial recognition, enhancing user interaction and content creation capabilities.

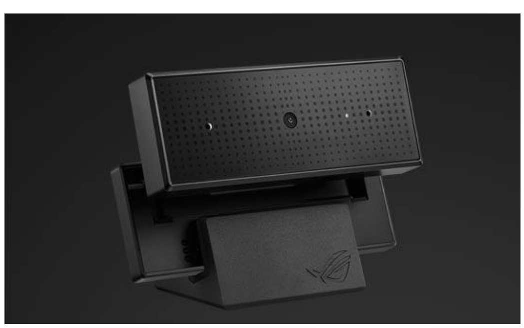


Figure 46: ASUS ROG Eye S Camera

5.2.1 Benefits of High-Level Control

- Efficiency: By integrating advanced image processing algorithms and real-time data processing with TensorFlow Lite, the high-level control ensures that the collector operates efficiently, minimizing the time and energy required to collect all tennis balls on the court.
- Reliability: The robust communication and control strategies implemented through serial USB and precise motor control enhance the system's reliability, enabling it to handle various operational scenarios with minimal errors.
- Scalability: The modular design of the high-level control allows for easy integration of additional features or sensors in the future, facilitating the scalability and enhancement of the project.
- In summary, the high-level control in the Autonomous Tennis Ball Collector project manages the complex interactions between the camera, processing unit, and motor control. By leveraging the strengths of the Raspberry Pi 4, TensorFlow Lite, and Arduino Mega 2560, this system achieves a cohesive, efficient, and intelligent solution capable of autonomously collecting tennis balls with precision and reliability.

5.2.2 Software

High-level control in the autonomous tennis ball collector project is orchestrated by the Raspberry Pi 4, which utilizes advanced processing capabilities to handle tasks such as object detection and decision-making. This layer of control involves sophisticated algorithms and machine learning models to detect tennis balls in real-time, process their coordinates, and communicate essential information to the Arduino for further action. By leveraging TensorFlow Lite for efficient model inference, the Raspberry Pi 4 ensures that the system operates with both precision and speed, essential for the dynamic environment in which the ball collector functions.

A crucial part of the high-level control code involves capturing video frames from the webcam, processing these frames to detect tennis balls, and extracting their x-coordinates.

```python
import argparse
import sys
import time
import cv2
from tflite_support.task import core
from tflite_support.task import processor
from tflite_support.task import vision
import utils
import numpy as np
```

Figure 47: Libraries

The script imports several necessary libraries, such as argparse for command-line argument parsing, sys for system-specific parameters, time for time-related functions, cv2 for OpenCV, tflite_support for TensorFlow Lite support, utils for utility functions.

```python
    # Initialize the object detection model
    base_options = core.BaseOptions(
        file_name=model, num_threads=num_threads, use_coral=enable_edgetpu)
    detection_options = processor.DetectionOptions(
        max_results=3, score_threshold=score_threshold)
    options = vision.ObjectDetectorOptions(
        base_options=base_options, detection_options=detection_options)
    detector = vision.ObjectDetector.create_from_options(options)
```

Figure 48: Object Detection Initialization

- **base_options**: Configures the basic options for the model, including the model file, number of threads, and whether to use EdgeTPU.
- **detection_options**: Sets detection-specific options like the maximum number of results and the score threshold.
- **options**: Combines the base and detection options into a single configuration.
- **detector**: Creates an object detector using the specified options.

```python
        # Convert the image from BGR to RGB as required by the TFLite model.
        rgb_image = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)

        # Create a TensorImage object from the RGB image.
        input_tensor = vision.TensorImage.create_from_array(rgb_image)

        # Run object detection estimation using the model.
        detection_result = detector.detect(input_tensor)
```

Figure 49: Run Object Detection

- **rgb_image = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)**: Converts the captured image from BGR format (used by OpenCV) to RGB format (required by TensorFlow Lite).
- **input_tensor = vision.TensorImage.create_from_array(rgb_image)**: Creates a TensorImage object from the RGB image, which is needed for inference.
- **detection_result = detector.detect(input_tensor)**: Runs the object detection model on the input tensor (image) and stores the detection results.

```python
    # Draw keypoints and edges on input image and calculate centroids
    frame, centroids = utils.visualize_and_get_centroids(frame, detection_result)

    # Print and send the centroid of object 1 if detected
    if centroids:
        x, y = centroids[0]  # Get the centroid of the first detected object
        # Only send if there is at least one detected object
        arduino.write(f'{x}\n'.encode())
```

Figure 50: Extracting X-Coordinate

- **`frame,centroids = utils.visualize_and_get_centroids(frame, detection_result)`**: Visualizes the detection results on the frame and calculates the centroids of the detected objects.

- **`if centroids`**: Checks if any centroids were detected (i.e., if there are any detected objects).

- **`x,y = centroids[0]`**: Retrieves the coordinates (x, y) of the centroid of the first detected object.

- **`arduino.write(f'{x}\n'.encode())`**: Sends the x-coordinate of the first detected object's centroid to an Arduino device via serial communication. The x-coordinate is encoded as a string followed by a newline character (`\n`) and then converted to bytes before being written to the Arduino.

5.3 Low-Level Control

5.3.1 DRIVETRAIN

Field Oriented Control (FOC) Motor Control

Field Oriented Control (FOC), also known as vector control, is an advanced method used for controlling AC motors, particularly permanent magnet synchronous motors (PMSM) and induction motors. It provides high performance and efficiency by controlling the motor's magnetic field and torque in a precise manner.

FOC works by decoupling the torque and flux components of the motor current. This allows independent control of each component, akin to controlling a DC motor, which simplifies the control strategy and enhances performance. The process begins with sensing the motor's phase currents and rotor position (or speed). These three-phase currents are then transformed into a two-axis orthogonal system using Clarke Transformation. The next step involves converting the alpha and beta components to a rotating reference frame aligned with the rotor flux using Park Transformation. This results in direct (d-axis) and quadrature (q-axis) components, which represent the magnetic flux and torque-producing currents, respectively.

Using PI (Proportional-Integral) controllers, the d-axis and q-axis currents are regulated independently. The d-axis current is typically controlled to maintain the desired flux level, while the q-axis current is controlled to produce the desired torque. The controlled d-axis and q-axis currents are then transformed back to the alpha and beta components using inverse Park Transformation, followed by converting these components back to three-phase currents using inverse Clarke Transformation. Finally, PWM signals are generated based on the transformed three-phase currents to drive the motor inverter, controlling the motor's stator voltages.

Field Oriented Control offers numerous advantages. It improves efficiency by optimizing motor performance and reducing energy consumption. It enhances performance by providing smooth and precise control over a wide range of speeds and torque, improving dynamic response. FOC enables rapid acceleration and deceleration, making it ideal for applications requiring high-speed variations. It ensures smooth operation by minimizing torque pulsations, leading to quieter motor operation. The control method is versatile, applicable to various types of AC motors, including PMSMs and induction motors. Moreover, FOC provides stable operation under different load conditions, enhancing the reliability of the motor drive system.

In conclusion, Field Oriented Control significantly enhances the performance, efficiency, and control of AC motors, making it a preferred choice in applications ranging from industrial automation to electric vehicles. Its ability to independently manage flux and torque, along with its adaptability to various motor types, underscores its importance in modern motor control systems.

5.3.2 Communication

Universal Asynchronous Receiver / Transmitter (UART)

UART (Universal Asynchronous Receiver/Transmitter) is a communication protocol used for asynchronous serial communication. It is widely used in embedded systems and microcontroller applications to facilitate data transfer between devices. Here are the key aspects of UART:

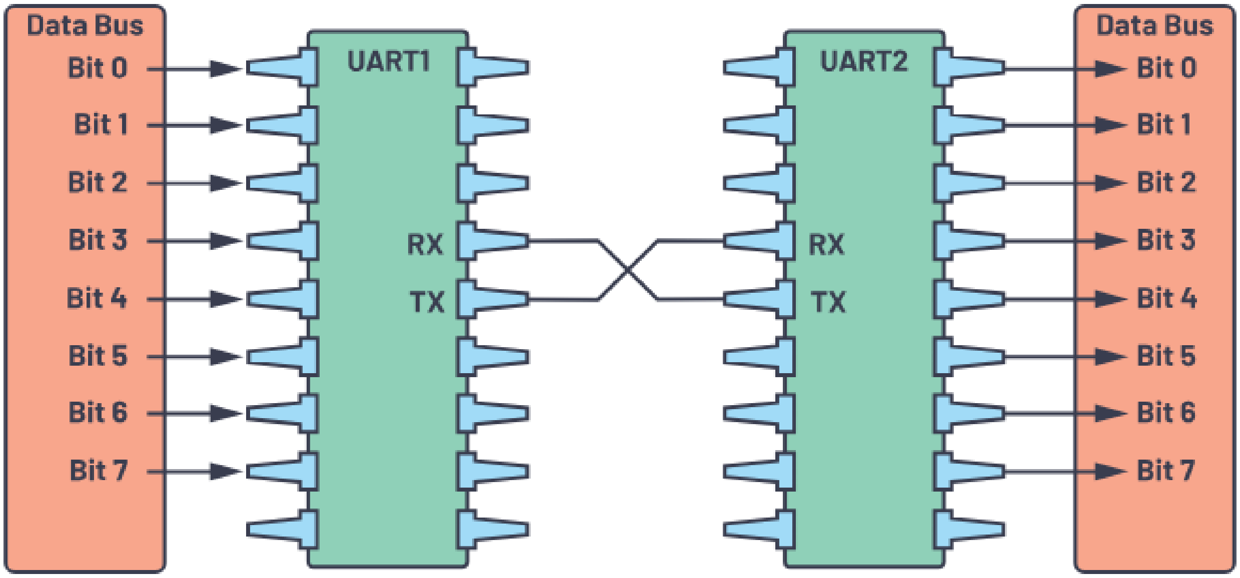


Figure 51: UART

Basics of UART Communication

1. **Asynchronous Communication**: UART communication is asynchronous, meaning there is no shared clock signal between the transmitting and receiving devices. Both devices must agree on the data rate (baud rate) before communication.
2. **Data Frame Structure**: UART transmits data in a structured frame, which typically consists of:
   - **Start Bit**: Indicates the beginning of a data frame. It is a single low bit (0).
   - **Data Bits**: The actual data being transmitted. This can be 5 to 9 bits long, but 8 bits is the most common.
   - **Parity Bit (Optional)**: Used for error checking. It can be even or odd parity.
   - **Stop Bit(s)**: Indicates the end of the data frame. It can be 1, 1.5, or 2 bits long.
3. **Baud Rate**: The speed of communication is defined by the baud rate, which is the number of bits transmitted per second. Common baud rates include 9600, 115200, etc.

4. **Full Duplex Communication**: UART can transmit and receive data simultaneously through two separate lines:
    - **TX (Transmit)**: The line used for sending data.
    - **RX (Receive)**: The line used for receiving data.

How UART Works

1. **Transmission**:
    - The transmitting UART converts parallel data from the data bus into serial form.
    - It adds start, parity, and stop bits to create a data frame.
    - The data frame is sent bit-by-bit over the TX line.
2. **Reception**:
    - The receiving UART detects the start bit on the RX line.
    - It reads the data bits according to the agreed baud rate.
    - It checks the parity bit for errors (if parity is used).
    - It removes the start, parity, and stop bits and converts the serial data back into parallel form for the data bus.

Error Detection

- **Parity Bit**: Used to detect single-bit errors. It adds an extra bit to ensure the number of 1s in the data frame is even (even parity) or odd (odd parity).
- **Framing Error**: Occurs if the expected stop bit is not detected.
- **Overrun Error**: Occurs if the receiver buffer is full and cannot store incoming data.

Advantages of UART

- **Simplicity**: Easy to implement and understand.
- **Low Cost**: Requires minimal hardware, typically just a pair of wires.
- **Widely Supported**: Standard in many microcontrollers and peripherals.

Disadvantages of UART

- **Limited Distance**: Effective communication range is relatively short without additional line drivers.
- **Limited Speed**: Slower compared to synchronous communication protocols like SPI and I2C.

- **No Clock Signal**: Requires both devices to have pre-configured baud rates, which can lead to synchronization issues if the clocks drift.

Applications of UART

- **Embedded Systems**: Commonly used for communication between microcontrollers and peripherals.
- **Serial Consoles**: Used in debugging and configuring embedded systems.
- **GPS Modules**: Communication with GPS receivers.
- **Wireless Communication Modules**: Such as Bluetooth and Wi-Fi modules.

Example of UART Communication

Consider a simple example where a microcontroller communicates with a GPS module:

1. **Initialization**: Both devices are configured to use the same baud rate, e.g., 9600 bps.
2. **Data Transmission**: The GPS module sends location data in NMEA format.
3. **Data Reception**: The microcontroller receives the data, processes the start, data, parity, and stop bits, and then uses the data.

Why Did We Use It

the Raspberry Pi 4 sends the x-coordinate extracted from the object detection model to the Arduino Mega 2560 through serial communication, the setup involves the Rpi4 acting as the controller and the Arduino managing the physical movement based on received coordinates.

5.3.3 Software

In the intricate world of software development for embedded systems and real-time applications, mastering the nuances of low-level control code stands as a cornerstone of success. This chapter delves deep into the pivotal lines of code that orchestrate direct interactions between software and hardware components. These lines, often unseen yet profoundly impactful, dictate how processors, peripherals, and sensors collaborate to execute precise instructions in real-time.

Understanding the Foundation: At the heart of every embedded system lies a foundation built upon meticulous lines of code. From initializing hardware registers and configuring interrupt handlers to managing memory and ensuring timing constraints, these snippets of code wield immense influence over the system's responsiveness and reliability.

```
#define HOVER_SERIAL_BAUD   115200      // [-] Baud rate for HoverSerial (used to communicate with the hoverboard)
#define SERIAL_BAUD         9600      // [-] Baud rate for built-in Serial (used for the Serial Monitor)
#define START_FRAME         0xABCD       // [-] Start frme definition for reliable serial communication
#define TIME_SEND           100         // [ms] Sending time interval
#include <Arduino.h>
#include <SoftwareSerial.h>
#include <Servo.h>

SoftwareSerial HoverSerial(16,17);        // RX, TX
```

Figure 52: Low-Level Libraries

`#include <SoftwareSerial.h>` → The line `#include <SoftwareSerial.h>` is essential when you need additional serial communication ports on an Arduino board that has limited hardware serial ports. Including this library allows you to set up and use multiple serial ports using digital pins, enabling communication with multiple serial devices simultaneously.

`SoftwareSerial HoverSerial(16,17);`

- **`SoftwareSerial`**: This is the class provided by the SoftwareSerial library that allows for serial communication on digital pins.
- **`HoverSerial`**: This is the name of the `SoftwareSerial` object being created. You can choose any valid identifier name.
- **`(16, 17)`**: These are the parameters passed to the `SoftwareSerial` constructor. They specify the RX (receive) and TX (transmit) pins for the software serial port.

```
typedef struct{
   uint16_t start;
   int16_t  steer;
   int16_t  speed;
   uint16_t checksum;
} SerialCommand;
SerialCommand Command;
```

Figure 53: This Structure is Written as Required By HoverBoard Controller

`SerialCommand Command;`

- **`SerialCommand`**: This is the alias name of the structure defined earlier.
- **`Command`**: This is a variable of type `SerialCommand`. It is an instance of the structure, which means it contains all the members (`start`, `steer`, `speed`, and `checksum`) defined in the structure.

```
  // Create command
  Command.start    = (uint16_t)START_FRAME;
  Command.steer    = (int16_t)uSteer;
  Command.speed    = (int16_t)uSpeed;
  Command.checksum = (uint16_t)(Command.start ^ Command.steer ^ Command.speed);

  // Write to Serial
  Serial2.write((uint8_t *) &Command, sizeof(Command));
}
```

Figure 54: Assigning Values to Structure Members

`Command.start = (uint16_t)START_FRAME;`

- This line assigns the value of `START_FRAME` to the `start` member of the `Command` structure.
- **`(uint16_t)`**: This cast ensures that `START_FRAME` is treated as an unsigned 16-bit integer.
- **`START_FRAME`**: This is a constant defined elsewhere in the code, typically used to mark the beginning of a command packet.

`Command.steer = (int16_t)uSteer;`

- This line assigns the value of `uSteer` to the `steer` member of the `Command` structure.
- **`(int16_t)`**: This cast ensures that `uSteer` is treated as a signed 16-bit integer.
- **`uSteer`**: This is a variable representing the steering value.

`Command.speed = (int16_t)uSpeed;`

- This line assigns the value of `uSpeed` to the `speed` member of the `Command` structure.
- **`(int16_t)`**: This cast ensures that `uSpeed` is treated as a signed 16-bit integer.
- **`uSpeed`**: This is a variable representing the speed value.

`Command.checksum = (uint16_t)(Command.start ^ Command.steer ^ Command.speed);`

- This line calculates a checksum for the command and assigns it to the `checksum` member of the `Command` structure.
- **`(uint16_t)`**: This cast ensures that the result of the bitwise XOR operation is treated as an unsigned 16-bit integer.

- **`Command.start ^ Command.steer ^ Command.speed`**: This bitwise XOR operation combines the values of `start`, `steer`, and `speed` to produce the checksum. The checksum is used to verify the integrity of the data.

```
Serial2.write((uint8_t *) &Command, sizeof(Command));
```

- This line sends the entire `Command` structure over the serial port `Serial2`.
- **`(uint8_t *) &Command`**: This cast converts the address of the `Command` structure to a pointer to an array of bytes (`uint8_t *`). This is necessary because `Serial2.write()` expects a pointer to a byte array.
- **`sizeof(Command)`**: This specifies the number of bytes to send, which is the size of the `Command` structure.

The provided code assigns values to the members of the `Command` structure, calculates a checksum for data integrity, and sends the entire structure over the `Serial2` port. This approach ensures that the command data is transmitted in a well-organized and error-checked format, making it easier to handle on the receiving end.

```
void continue_forward(void){
  for(int i = 0; i<1000;i++)
  {
    timeNow = millis();
```

Figure 55: Calculating The Time Since The Program Started Running

```
timeNow = millis();
```

- **Function**: `millis()` is a built-in function in the Arduino environment.
- **Return Value**: It returns an unsigned long integer representing the number of milliseconds that have elapsed since the program started running. This timer overflows and resets to zero approximately every 50 days.

The `millis()` function is often used for tasks like:

- **Timing Events**: Determining how much time has passed between events.
- **Delays without Blocking**: Creating delays without halting the entire program (unlike the `delay()` function).
- **Scheduling**: Scheduling tasks to run at specific intervals.

```
Serial.begin(SERIAL_BAUD);
Serial.flush();
Serial.println("Hoverboard Serial v1.0");
delay(100);
Serial2.begin(HOVER_SERIAL_BAUD);
Serial2.flush();
```

Figure 56: Initializing UART and Serial Communication

```
Serial.flush();
```

- **Purpose**: This function is used to wait for the transmission of outgoing serial data to complete.
- **Use Case**: It is useful when you want to make sure that all data in the serial buffer has been sent before moving on to the next part of your code.

5.4 System Block Diagram

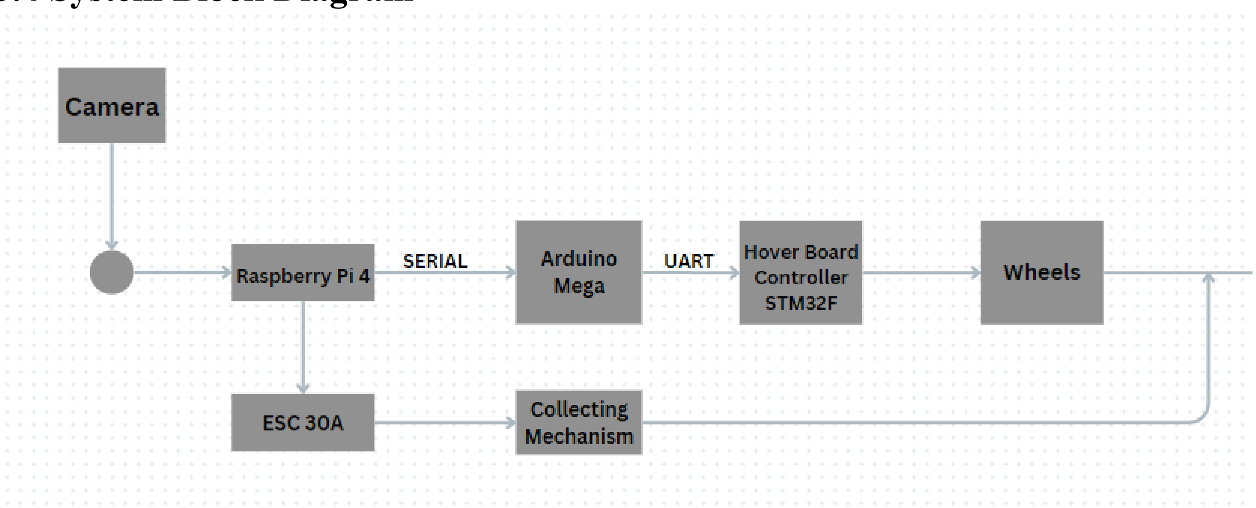


Figure 57: Sytem Block Diagram

5.5 Matlab Implementation

Simscape is a simulation software developed by MathWorks that is integrated with MATLAB and Simulink. It provides a platform for modeling and simulating physical systems across various domains, including mechanical, electrical, hydraulic, and thermal systems. Simscape enables engineers and researchers to create accurate and realistic models of complex physical systems without needing to derive and solve the underlying mathematical equations manually. Here's an overview of Simscape and its key features:

- **Key Features of Simscape:**

1. Multidomain Modeling:
   - Simscape supports modeling across multiple physical domains, allowing users to combine mechanical, electrical, hydraulic, thermal, and other types of systems within a single simulation environment.

2. Component Libraries:
   - Includes a wide range of predefined components such as resistors, capacitsors, motors, gears, and valves, making it easy to build complex models.
   - Users can also create custom components to fit specific needs.

3. Physical Network Approach:
   - Models are built using a physical network approach, where components are connected based on their physical relationships, similar to how they would be connected in the real world.

4. Integration with MATLAB and Simulink:
   - Seamlessly integrates with MATLAB for data analysis, visualization, and scripting.
   - Works within the Simulink environment, allowing for easy combination of control systems and physical models.

5. Real-Time Simulation:
   - Supports real-time simulation and testing, which is essential for hardware-in-the-loop (HIL) testing and rapid control prototyping.

6. Solver Options:
   - Offers various solvers to handle different types of physical systems, ensuring accurate and efficient simulation results.

7. Parameterization and Customization:

- Allows parameterization of models using MATLAB variables and scripts, enabling easy modification and optimization.
- Users can write custom functions and components using the MATLAB language.

8. Visualization and Animation:
- Provides tools for visualizing simulation results, including scopes, dashboards, and 3D animations, to better understand system behavior.

- **Benefits of Using Simscape:**

1. Simplifies Complex Modeling:
- Eliminates the need to manually derive and solve differential equations, saving time and reducing errors.

2. Enhances Collaboration:
- Facilitates collaboration between engineers from different disciplines by providing a common platform for modeling and simulation.

3. Improves Design and Testing:
- Enables thorough testing and validation of designs before physical prototypes are built, reducing development costs and time.

4. Accelerates Development:
- Speeds up the development process by allowing rapid prototyping and iterative testing within the simulation environment.

5. Increases Accuracy:
- Provides accurate and realistic models that closely mimic the behavior of real-world systems, leading to more reliable and effective designs.

- **Typical Use Cases for Simscape:**

1. Automotive Engineering:
- Modeling and simulation of vehicle dynamics, powertrains, and control systems.

2. Aerospace Engineering:
- Simulation of aircraft systems, including flight dynamics, control surfaces, and propulsion systems.

3. Industrial Automation:
   - Design and testing of industrial machinery, robotics, and control systems.

4. Energy Systems:
   - Modeling of renewable energy systems, power electronics, and electrical grids.

5. Biomedical Engineering:
   - Simulation of physiological systems and medical devices.

- **Conclusion:**

Simscape is a powerful tool for modeling and simulating physical systems across multiple domains. Its integration with MATLAB and Simulink, along with its comprehensive libraries and solver options, makes it an essential tool for engineers and researchers. By simplifying complex modeling tasks and providing accurate simulations, Simscape enhances the design, testing, and optimization processes, ultimately leading to better-engineered systems and products.

5.5.1 Simscape Diagrams

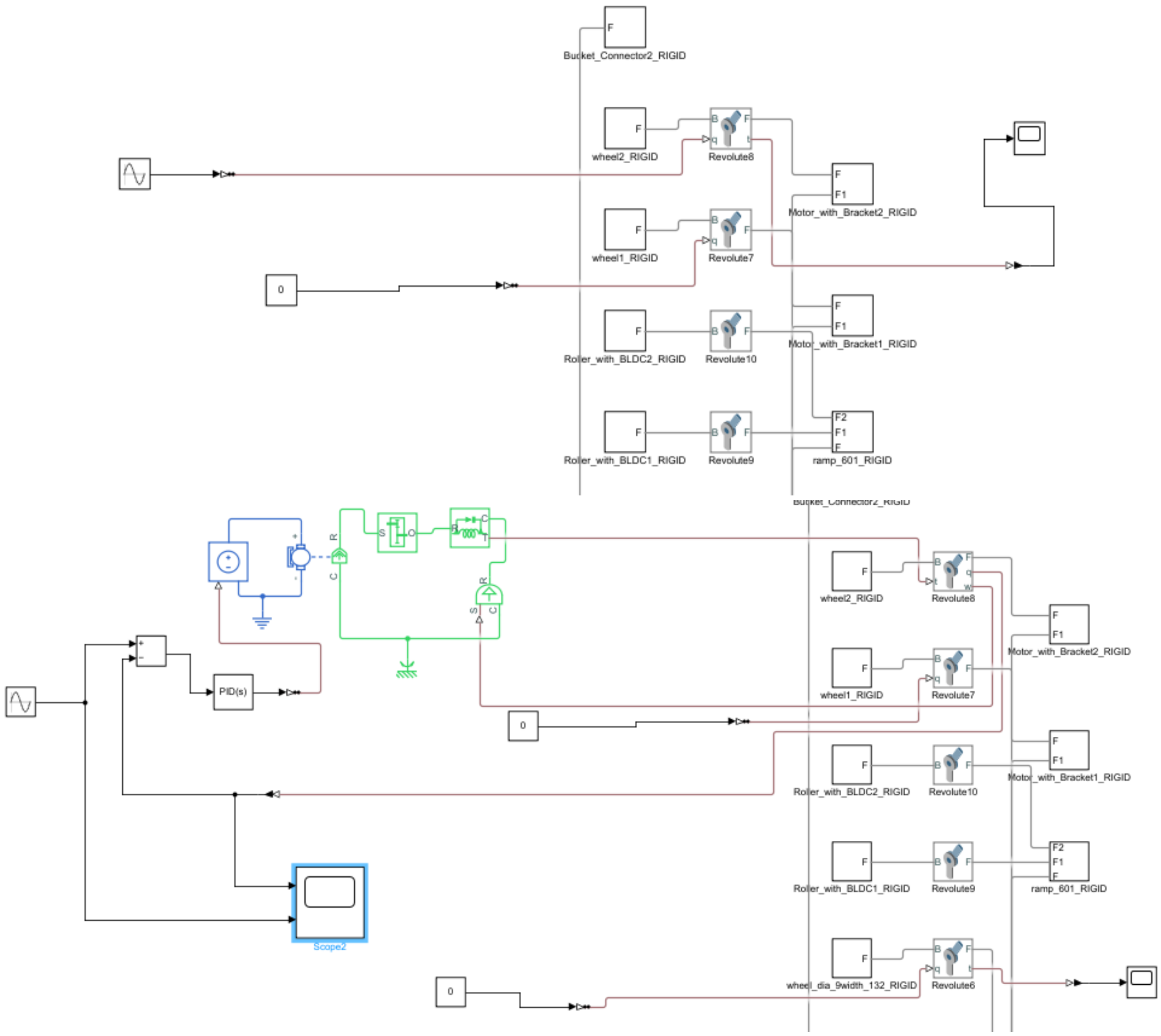


Figure 58: Simscape Design

5.5.2 Simscape Graphs

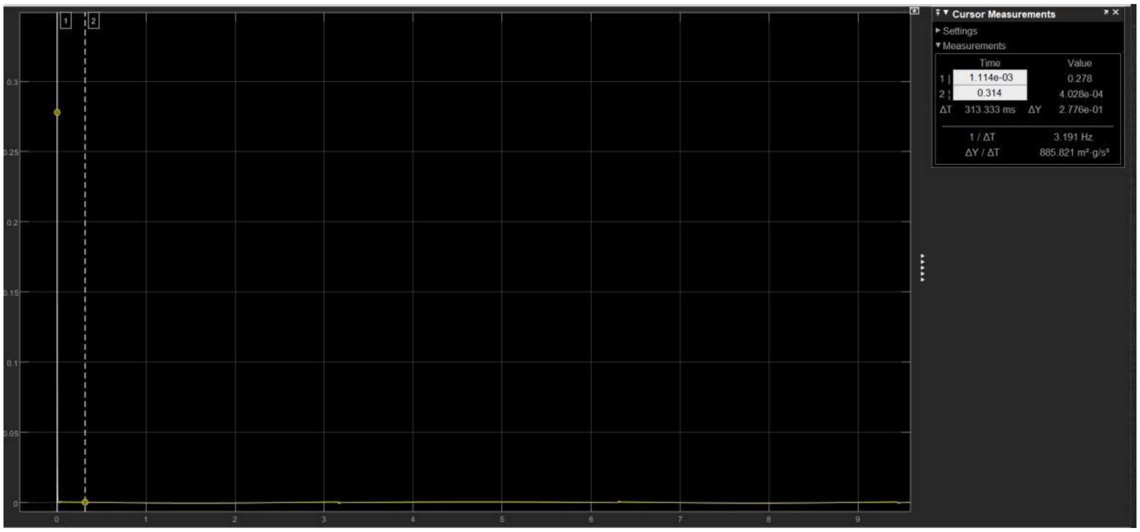


Figure 59: Torque - Time Graph

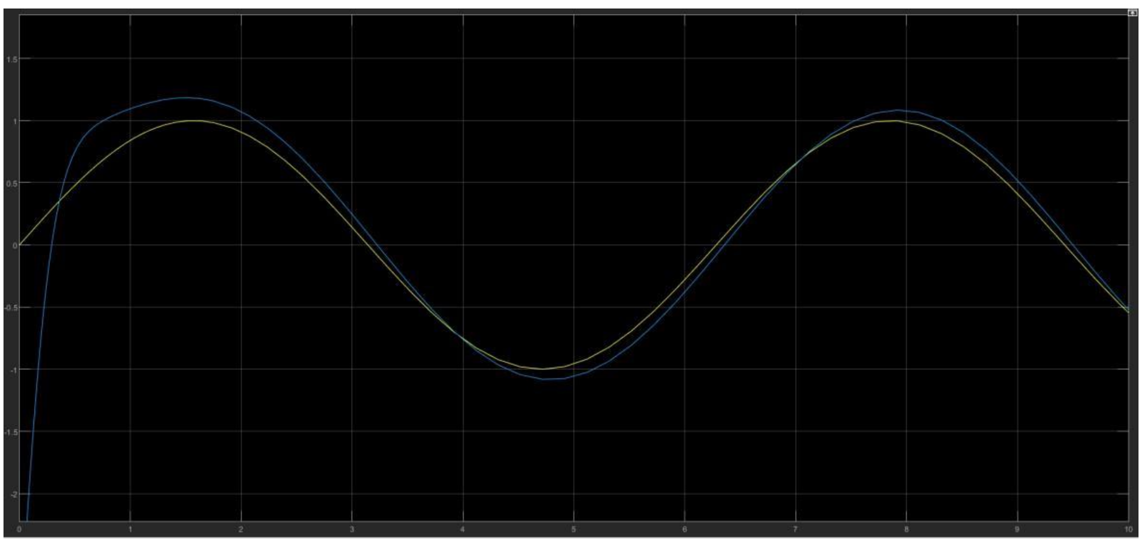


Figure 60: Speed - Time Graph

5.6 Printed Circuit Boards

PCBs (Printed Circuit Boards) are better than breadboards for most projects because they are more reliable and durable. Breadboards can have loose connections, but PCBs provide permanent, stable connections. They reduce electrical noise and are better for high-frequency circuits.

PCBs are compact and can be custom-shaped, making them ideal for small devices. They can handle complex circuits and multiple layers, unlike breadboards. They can also be mass-produced consistently, which is important for commercial products.

PCBs look professional with neat layouts and printed labels. They manage heat well, which is crucial for high-power circuits. They support automated assembly, making production faster and reducing labor costs. Testing and debugging are easier with PCBs because they can include test points.

While breadboards are useful for quick prototypes and simple projects, PCBs are essential for reliable, durable, and high-performance electronics. Their reliability, compactness, and professional look make them the preferred choice for most electronic applications.

5.6.1 Software

Eagle (Easily Applicable Graphical Layout Editor) is a popular PCB (Printed Circuit Board) design software used by electronics engineers, designers, and hobbyists to create schematic diagrams and PCB layouts. Here's a detailed overview of Eagle software

Typical Use Cases for Eagle Software:

1. **Prototyping**: Quickly design and test electronic circuits.
2. **Product Development**: Create PCB layouts for commercial products.
3. **Educational Projects**: Ideal for learning electronics and PCB design.

Schematic Capture

1. **Component Placement**:
   - EAGLE provides an extensive library of components, making it easy to add parts to your schematic. You can also create custom components if needed.
   - Commands like ADD facilitate adding components, and MOVE helps in adjusting their positions.
2. **Net Connections**:
   - The NET command is used to draw connections between components. Logical connections can be highlighted using SHOW.
   - The LABEL and NAME commands help in naming and labeling different nets for better organization and clarity.
3. **Hierarchy and Organization**:
   - Allows for the creation of hierarchical designs, enabling complex schematics to be broken down into manageable sub-circuits.

4. **Design Rule Check (ERC)**:
    - The ERC checks the schematic for common electrical errors, ensuring connections are logically sound.

PCB Layout

1. **Component Placement and Routing**:
    - Once the schematic is complete, you can switch to the board layout view where you place components on the PCB.
    - The ROUTE command is used to manually draw traces connecting components. RATSNEST helps in optimizing these connections by displaying the shortest possible routes.
2. **Auto-Router**:
    - EAGLE includes an auto-router feature that can automatically create routes for your PCB, saving time and effort for complex designs.
3. **Design Rule Check (DRC)**:
    - Ensures that the PCB design adheres to the predefined manufacturing rules, checking aspects like trace width, spacing, and component placement.
4. **Layer Management**:
    - Supports multi-layer PCB designs, allowing you to route connections across different layers of the board.
    - The LAYER command is used to manage and switch between different layers.
5. **Gerber File Generation**:
    - EAGLE can generate industry-standard Gerber files required for PCB manufacturing. This includes files for copper layers, solder masks, and silkscreen.

Component Libraries

1. **Extensive Built-In Libraries**:
    - EAGLE comes with a vast library of components from various manufacturers, making it easy to find the parts you need for your design.
2. **Custom Libraries**:
    - Users can create and manage their own libraries, adding new components or modifying existing ones to suit their needs.
3. **Third-Party Libraries**:
    - Many third-party libraries are available, provided by companies like SparkFun and Adafruit, which can be easily integrated into EAGLE.

Simulation and Testing

1. **SPICE Simulation**:
    - EAGLE integrates with SPICE (Simulation Program with Integrated Circuit Emphasis) to simulate and test circuits before moving to the PCB layout stage.

3D Visualization

1. **Fusion 360 Integration**:
    - EAGLE integrates with Autodesk Fusion 360, allowing you to create 3D models of your PCB designs. This is particularly useful for visualizing the final product and ensuring mechanical compatibility with enclosures and other hardware.

User Interface and Usability

1. **Customizable Interface**:
    - The EAGLE interface is customizable, allowing users to arrange toolbars and windows to suit their workflow.
2. **Scripting and ULPs**:
    - User Language Programs (ULPs) allow users to write custom scripts to automate tasks and extend the functionality of EAGLE.

Collaboration and Version Control

1. **Team Collaboration**:
    - EAGLE supports collaborative design, making it easier for teams to work together on complex projects. Designs can be shared and versioned using Autodesk's cloud services.
2. **Version Control**:
    - Integration with version control systems helps in tracking changes and managing different versions of a design.

Documentation and Support

1. **Comprehensive Documentation**:
    - EAGLE provides extensive documentation and tutorials, making it accessible for both beginners and advanced users.
2. **Active Community and Support**:
    - An active user community and support forums provide help and resources for troubleshooting and learning new features.

5.6.2 Final PCB

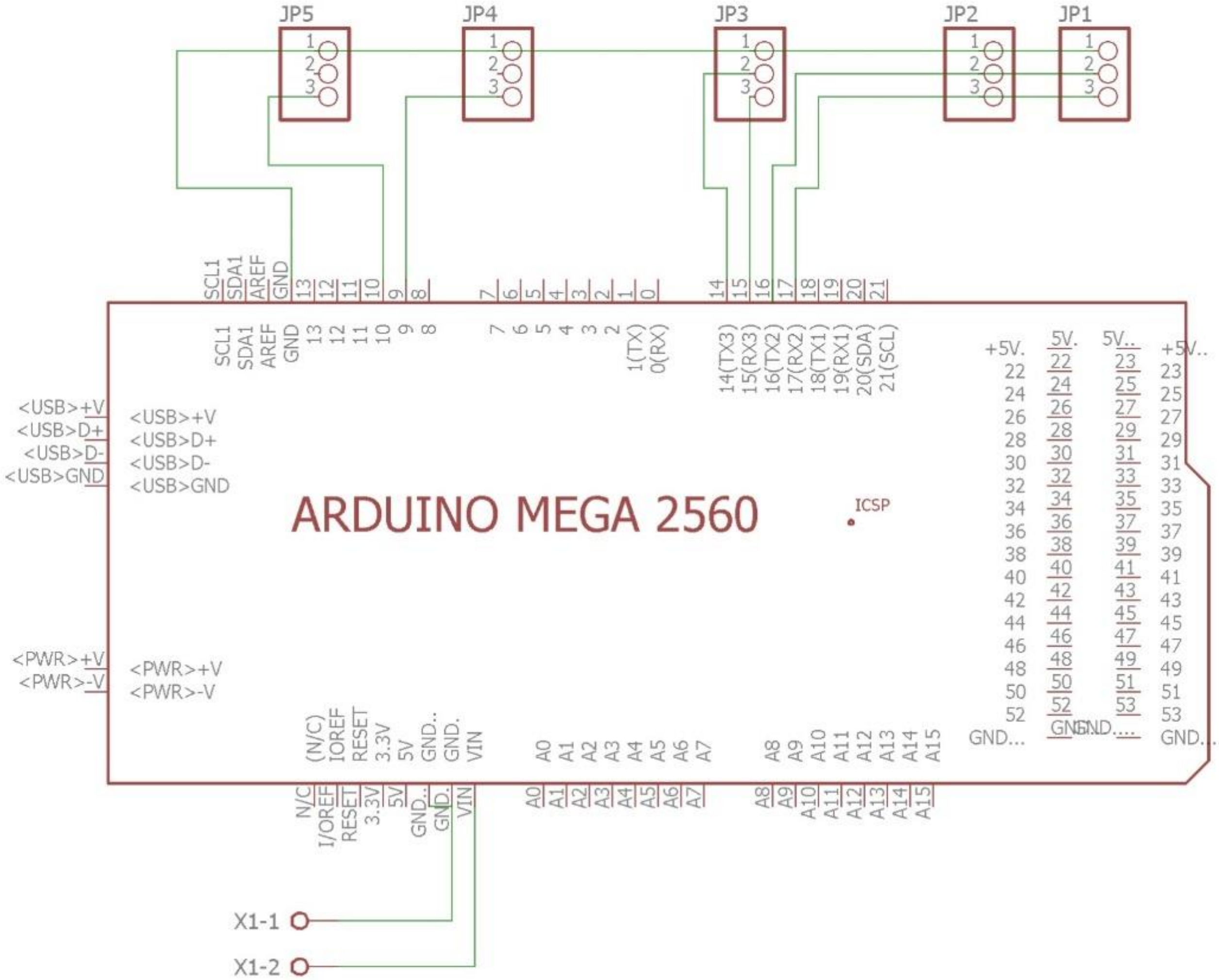


Figure 61: PCB Layout

5.7 Control Flow Chart

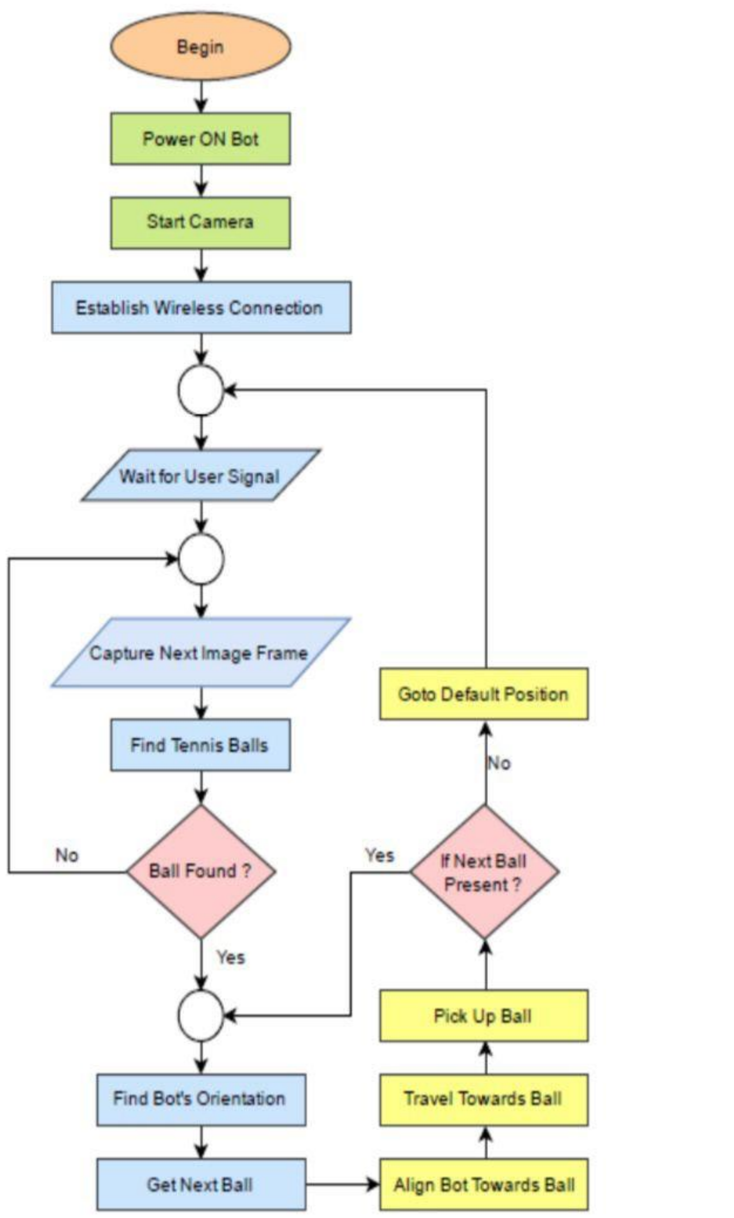


Figure 62: System Flow Chart

5.8 Methodology

This research is focused on enhancing the capabilities of an autonomous tennis ball picker equipped with a launcher mechanism designed as well as with a camera to locate and collect tennis balls. accuracy take precedence in this project, with an emphasis on achieving effective performance. The chosen methodologies prioritize the device's reliability and user-friendly operation.

The research for this autonomous tennis ball picker project can be broadly divided into two main areas: mechanical design and control systems. To gather relevant insights, we conducted extensive research by reviewing previous research papers related to similar projects.

In terms of mechanical design, we employed Autodesk Inventor to rigorously assess the device's design. This allowed us to ensure that the design meets the required parameters concerning stress, strain, and safety factors, while also facilitating a more comprehensive understanding of the device's mechanical aspects.

Turning our attention to the control methodology, after thorough research, we observed that the majority of similar projects were using ROS (Robot Operating System) as their preferred programming language and testing software. However, given our specific project requirements, which aim to minimize processing power demands, we decided to opt for MATLAB as our preferred programming language. MATLAB's open-source nature and high-level control capabilities make it an ideal choice for our autonomous tennis ball picker, simplifying the controller selection process while ensuring precision and efficiency in control.

5.8.1 Methodology Flowchart

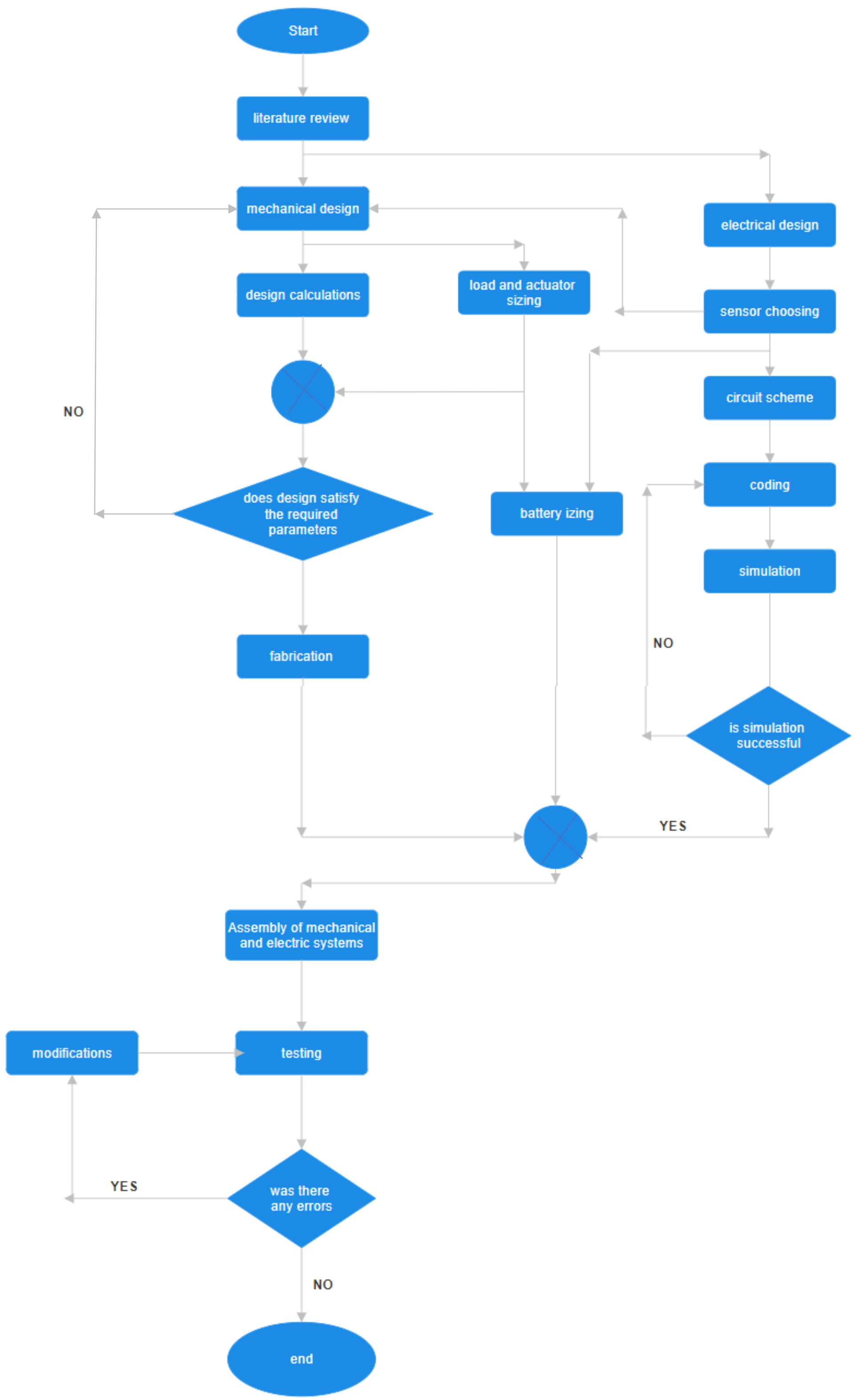


Figure 63: Methodology Flowchart

First, gathering information from previous research papers could be the hardest but the most important thing to get a successful product in any field. moreover, it’s also important to develop the previous inventions and to have a more efficient new one.

The Autonomous Tennis Ball picker have 2 main designs, mechanical & electrical and both serve each other:

In the mechanical design and after getting the most suitable sensors, design calculations, load and actuator sizing process was built. Moreover, battery sizing was calculated in order to get the suitable power needed to operate the robot. Furthermore, if the design doesn’t satisfy the required parameters some edits can be added to the mechanical design if not then the fabrication process will start and here both designs mechanical & electrical will be combined by starting the assembly process of both systems. Electrical design started by getting the specs of the specified user weight, how many balls its going to collect in order to know some parameters such as torque, power…etc for driving motors. Sensor choosing was taken to get the best sensors & motors to be used. In addition, circuit scheme was built to get the connections needed for the sensors chosen.

Furthermore, MATLAB was used because of it’s accuracy and simplicity. Its accuracy in calculating the dc motors parameters by using sim scape, Simulink properties. Using Eagles software to design and picturize the control circuit that contain sensors and all the components embedded together. Finally, simulation was done to get the results and to ensure that there are no modifications needed in the code.

Finally, testing process will start on the user in order to check that everything is working according to the plan, errors will be taken in consideration and if there are more modifications needed it will be done for more accuracy.

C h a p t e r S i x

6.0 FEASIBILITY ANALYSIS

Feasibility research is a preliminary study conducted to assess the viability of a proposed project or idea. It involves analyzing various aspects of the project to determine whether it is practical and achievable within the given constraints and resources. The primary goal is to identify potential challenges and determine if the project is worth pursuing.

For our project, the autonomous tennis ball collector, we conducted comprehensive market and price research. We identified the costs for each component, raw material, and manufacturing process, focusing on local suppliers known for high quality and competitive prices. We included detailed research on the best prices for the market these days, ensuring that we selected components and materials that offered the best balance between quality and affordability.

Through this feasibility study, we evaluated all relevant factors to ensure our project could be completed within a reasonable budget while maintaining high standards of quality and efficiency. This research allowed us to make informed decisions, optimize our resource use, and increase the likelihood of project success.

In summary, a feasibility study is crucial for mitigating risks, ensuring financial and technical viability, complying with legal requirements, allocating resources effectively, understanding market potential, and managing time. By thoroughly evaluating the feasibility of a project before starting, we can make informed decisions, optimize resource use, and increase the likelihood of project success.

Component	Amount	US Dollars	Egyptian Pound
Raspberyy Pi 4 8GB	1	$156.38	7500.00 EGP
Arduino Mega 2560	1	$36.68	1750.00 EGP
Hover Board Hub Motor	4	$54.21	2600.00 EGP
ASUS ROG EYE S Camera	1	$93	4460.41 EGP
BLDC Motor 2200KV	2	$16.26	780.00 EGP
Standard ESC 30A	2	$27.11	1300.00 EGP
LM2596	1	$1.67	80.00 EGP
Rechargeable Li-ion Battery Pack 36V	2	$43.78	2100 EGP
JOYROOM JR-PBF01 30W Power Bank	1	$38.99	1870 EGP
Hover Board Controller	1	$19.81	950 EGP
3-Cell AAA Battery Holder	3	$1.69	81 EGP
Li-ion 18650 Battery 4200mAh	9	$14.07	675 EGP
3D Printed Rollers	2	$16.68	800 EGP
3D Printed Arms	2	$33.36	1600 EGP
Raw Materials			
Mild Steel	-	-	800.00 EGP
Wood	-	-	1600.00 EGP
Stainless Steel	-	-	500.00 EGP
ABS Plastic	-	-	2400 EGP
Manufacturing			
Bending	-	-	1000.00 EGP
CNC turning	-	-	1500.00 EGP
Painting	-	-	1500.00 EGP

Table 18: Feasibility Analysis

C h a p t e r   S e v e n

7.0 CONCLUSION

In conclusion, studies concerning autonomous tennis ball pickers has shown how robotics and artificial intelligence have huge potential to transform the sports industry. These innovative machines have shown to be incredibly successful and efficient at collecting tennis balls from the court because of the development and use of advanced technologies like computer vision, machine learning, and autonomous navigation systems. Researchers have used computer vision algorithms that make use of image processing techniques to accomplish accurate ball detection. These algorithms enable the autonomous pickers to detect and monitor tennis balls in real-time through analyzing the visual data acquired by cameras placed on the pickers. The study's conclusions have brought to light the many advantages that autonomous tennis ball pickers offer to the sport. First of all, they greatly decrease the physical burden on coaches and players, allowing them up for concentrating more on their performance and training. These devices free up players' time and energy by automating the tiresome chore of gathering tennis balls, allowing them to make the most of their training sessions and advance their skills. Additionally, automatic tennis ball pickers improve the general effectiveness and output of tennis practice sessions. These devices guarantee a steady and unbroken practice tempo because of their quick and accurate ball collection from anywhere on the court. This raises the number of rounds players can complete while also improving the quality of training, which speeds up the development of new skills. Moreover, the use of autonomous tennis ball pickers encourages training grounds that are safer. It is much less likely for players or coaches to get injuries from slips, falls, or overexertion when they are not required to physically collect balls. As a result, players are free to concentrate just on their performance and skill, free from interruptions or possible mistakes. Furthermore, the study has clarified whether autonomous tennis ball pickers could be economically advantageous. These devices may need a large initial expense, but overall, the advantages exceed the disadvantages. These devices have the potential to save tennis facilities and academy an enormous amount of money over time by eliminating the need for laborious tasks and lowering the danger of accidents. It is essential to remember that the capabilities and possible uses of autonomous tennis ball pickers are still being investigated in this work. Autonomous tennis ball pickers have advantages that go beyond practicality. They help with recycling efforts and cut down on waste in tennis by having the capacity to gather and store used balls. We could expect more advancements in technology's performance, effectiveness, and flexibility as it develops. More advancements in this area of study and research will surely result in even more advanced devices that are capable of meeting the unique demands and specifications of various tennis venues.

8.0 CITATION AND REFERENCING

[1] I. Elamvazuthi et al., “Development of an Autonomous Tennis Ball Retriever Robot As an Educational Tool,” Procedia Comput. Sci., vol. 76, no. Iris, pp. 21–26, 2015.

[2] H. K. Chen and J. M. Dai, “An intelligent tennis ball collecting vehicle using smart phone touch-based interface,” Proc. - 2016 IEEE Int. Symp. Comput. Consum. Control. IS3C 2016, pp. 362–365, 2016.

[3] Haitham Eletrabi, Robotic Tennis Ball Collector “Tennibot”, pages. Available at http://www.tennibot.com. (last visit date November 2017)

[4] Inwatec, Inwatec Tennis Ball Collector - www.inwatec.dk, Youtube, Published on 4/09/2012, 0:24, https://www.youtube.com/watch?v=d1Fco37SC2E

[5] Mailman, C. J. (2012). U.S. Patent No. 8,313,396. Washington, DC: U.S. Patent and Trademark Office.

[6] The Trash Pack Street Sweeper TV Spot. (n.d.). Retrieved December 12, 2017, from https://www.ispot.tv/ad/777E/the-trash-pack-street-sweeper

[7] Chawla, N., Alwuqayan, W., Faizan, A., & Tosunoglu, S. (2012). Robotic Tennis Ball Collector, (February 2018), 1–6.

[8] Chen, H. K., & Dai, J. M. (2016). An intelligent tennis ball collecting vehicle using smart phone touch-based interface. Proceedings - 2016 IEEE International Symposium on Computer, Consumer and Control, IS3C 2016, 362–365. http://doi.org/10.1109/IS3C.2016.100.

[9] ExMark Lawn Mowers. (n.d.). Retrieved December 10, 2017, from http://more.crunchybeachmama.com/2013/03/exmark-lawn-mowers.

[10] Valdes-Rodriguez, R. (2000). U.S. Patent No. 6,079,930. Washington, DC: U.S. Patent and Trademark Office.

[11] Yu, Y., & Kim, S. (2014, July 31). TENNIS BALL BOY. Retrieved December 11, 2017, from https://ifworlddesignguide.com/entry/133917-tennis-ball-boy

[12] 10-S Tenn-Tube - 20 Ball. (n.d.). Retrieved December 15, 2017, from http://www.10-s.com/Products/Ball-Tubes/IF2003

[13] AHM 38 G | Hand mowers | Lawnmowers | Garden. (n.d.). Retrieved December 13, 2017, from https://www.bosch-garden.com/gb/en/garden-tools/garden-tools/ahm-38-g 3165140578929-199958.jsp#tab_technical.
85

[14] M.dhgate.com. (n.d.). 18" 20" Men's Large Vintage Genuine Leather laptop Travel Wheeled Duffle hand luggage bag Carry On Rolling Duffel bags men. Retrieved December 30, 2017, from https://m.dhgate.com/product/18-quot-20-quot-men-039-slarge-vintage-genuine/387536439.html.

[15] Siegwart, R., Nourbakhsh, I. R., & Scaramuzza, D. (2011). Introduction to Autonomous Mobile Robots. MIT Press. Retrieved from https://books.google.ps/books?id=4of6AQAAQBAJ

[16] Siciliano, B., Sciavicco, L., Villani, L., & Oriolo, G. (2010). Robotics: Modelling, Planning and Control. Springer London. Retrieved from https://books.google.ps/books?id=jPCAFmE-logC

[17] Ahmad Abu Hatab, R. D. (2013). Dynamic Modelling of Differential-Drive Mobile Robots using Lagrange and Newton-Euler Methodologies: A Unified Framework. Advances in Robotics & Automation, 02(02). https://doi.org/10.4172/2168-9695.1000107.

[18] Neĭmark JI, Fufaev NA (1972) Dynamics of Nonholonomic Systems: Translations of Mathematical Monographs. American mathematical Society.

[19] Bloch A, Crouch P, Baillieul J, Marsden J (2003) Nonholonomic Mechanics and Control.

[20] Joshi, B., Shrestha, R., & Chaudhary, R. (2014). Modeling , Simulation and Implementation of Brushed DC Motor Speed Control Using Optical Incremental Encoder Feedback DC Motor Model Simulink Modeling , Simulation and Parameter estimation. Proceeding of IOE Graduate Conference, (November), 497–505.

[21] Richard C Dorf and Robert H Bishop. Modern control systems. Pearson, 2011.

[22] Anvari, I. (2013). Non-holonomic Differential Drive Mobile Robot Control & Design: Critical Dynamics and Coupling Constraints (Doctoral dissertation, Arizona State University).

[23] Tzafestas, S. G. (2014). Introduction to Mobile Robot Control. Introduction to Mobile Robot Control. https://doi.org/10.1016/B978-0-12-417049-0.00001-8

[24] F. (n.d.). CHEER 4 FTC. Retrieved March 1, 2018, from http://cheer4ftc.blogspot.com/p/motors.html

[25] Tennis. (n.d.). In Wikipedia. Retrieved October 8, 2017, from http://en.wikipedia.org/wiki/Psychology.

[26] A. (n.d.). Retrieved November 17, 2017, from https://www.amazon.co.uk/Razor-electricscooter-E90-
Spacer/dp/B00DKDBKEE/ref=sr_1_1?s=sports&ie=UTF8&qid=1527614352&sr=1-1&keywords=140mm scooter wheel.

[27] A. (n.d.). Retrieved November 18, 2017, from https://www.amazon.co.uk/JD-Bug200mm-Wheel
86
Bearings/dp/B01EI520CS/ref=sr_1_3?s=sports&ie=UTF8&qid=1527614020&sr=1-3&keywords=200mm scooter wheel

[28] Pitsco Education: TETRIX®. http://www.tetrixrobotics.com/ (2014). Accessed 10 Apr 2017.

[29] https://www.xometry.com/resources/materials/pc-abs/

[30]https://www.simscale.com/docs/simwiki/fea-finite-element-analysis/what-is-von-mises-stress/.

[31] https://safetyculture.com/topics/factor-of-safety/.

[32]https://www.degruyter.com/document/doi/10.1515/geo-2020-0051/html#:~:text=Displacement%20and%20deformation%20studies%20involve,geometric%20changes%20in%20the%20structure.

[33] https://www.youtube.com/watch?v=1X32--5F3gQ&ab_channel=SOLIDCAMPRO.

[34] https://www.researchgate.net/figure/Max-Principle-Stress-Mild-Steel_fig5_346139825.

[35]https://www.lincolntech.edu/news/skilled-trades/welding-technology/types-of-welding-procedures.

[36] https://www.parallax.com/product/6-5-hub-motor-with-encoder/.

[37] https://grabcad.com/library/brushless-motor-bldc_2212-6t-2200kv-1

[38] 6.5" Hub Motor With Encoder - Parallax