Report

Implementation and Design of a Leaf Collection Robot

By

Mr. Lucky Agarwal	67011698
Ms. Pitchaporn Wattanaprapawit	67011268
Mr. Pannawish Siriwaleewiah	67011229
Mr. Joon Niramitsuphachet	67011602
Mr. Sriharach Srihawanit	67011313
Ms. Sitapat Tippaphanat	67011306
Mr. Theechutha Suphachitkulchai	67011351
Mr. Poonnachote Kaewsuwan	67011649
Mr. Rachavin Vitayakovit	67110139
Ms. Papimon Leelamali	67011635

Course ID: 01416304

Course Name: Feedback Control

Semester: 1/2568

Department of Robotics and AI Engineering

School of Engineering

King Mongkut's Institute of Technology Ladkrabang

ABSTRACT IN ENGLISH

This project presents the design and development of an autonomous Leaf Collection Robot aimed at efficiently collecting fallen leaves in outdoor environments such as lawns and campus areas. The system addresses the labor-intensive and inefficient process of manual leaf collection by integrating LiDAR-based SLAM navigation, Micro-ROS communication, and a mechanized roller and storage unit. The robot utilizes a Jetson Nano for mapping and real-time navigation while an ESP32 microcontroller handles motor control and odometry feedback through encoders. The system can operate autonomously or manually, avoiding obstacles and navigating through predefined routes. The custom-designed chassis, roller mechanism, and control circuit ensure reliable leaf collection and storage with minimal human intervention. The results demonstrate the robot's capability to map environments, navigate effectively, and collect leaves efficiently—providing a sustainable and practical solution for automated leaf management.

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to all our team members for their hard work, dedication, and invaluable contributions throughout the development of this project. Each member's effort, creativity, and commitment have been fundamental to the success of our work.

Our deepest appreciation goes to our advisors, Mr. Pitikhate Sooraksa and Miss Wandee Petchmaneelumka, for their continued guidance, encouragement, and insightful feedback. Their expertise and constructive advice have greatly helped us refine our ideas, overcome challenges, and achieve our project objectives.

We would also like to extend our heartfelt thanks to the Department of Robotics and AI Engineering at King Mongkut's Institute of Technology Ladkrabang (KMITL) for providing us with the essential facilities, learning resources, and financial support necessary for the completion of this project. The department's commitment to fostering innovation and research created an environment that allowed us to explore, experiment, and grow academically.

In particular, we wish to acknowledge Mr. Poonnachote Kaewsuwan for his supervision of the software design and implementation, and Mr. Pannawish Siriwaleewiah for his expert guidance in hardware design. Their technical insight and mentorship were instrumental in ensuring both the robustness and reliability of our system.

TABLE OF CONTENTS

Page

Abstract i

ACKNOWLEDGEMENT ii

TABLE OF CONTENTS iv

LIST OF TABLES v

LIST OF FIGURES vi

CHAPTER 1 INTRODUCTION **1**

1.1 Background and Problem 1

1.2 Objectives 1

1.3 Scope 2

1.4 Expected Benefits and Outcomes 2

CHAPTER 2 LITERATURE REVIEW AND RELATED THEORIES **3**

2.1 Concepts and Theories 3

2.2 Related Works 3

CHAPTER 3 METHODOLOGY AND DESIGN **5**

3.1 Hardware Design 5

3.1.1 Assembly 5

3.1.2 Cover & Lid 5

3.1.3 Base 6

3.1.4 Roller 7

3.1.5 Box 8

3.1.6 Motor Holders 10

3.1.7 Wheel & Roller Covers 10

3.1.8 Component Holder 11

3.1.9 Wheel 12

3.1.10 Unused Parts 12

3.2 Component Selection Rationale 13

3.2.1 Motor Selection 13

3.2.1.1 Encoder DC Motor – JGB37-520 13

3.2.1.2 Roller Motor – 25GA-370 14

3.2.2 LiDAR Sensor 14

3.2.3 Microcontrollers and Processing Units 14

3.2.3.1 ESP32 Microcontroller 14

3.2.3.2 Jetson Nano 15
3.2.4 Power Components and Safety Elements 15
3.2.4.1 LM2576 Voltage Regulator 15
3.3 Circuit Design 15
3.3.1 Circuit Schematic 16
3.3.2 Power Regulation and Distribution 16
3.3.3 Motor Control Circuit 17
3.3.3.1 MDD20A Dual H-Bridge Driver (Wheel Motors) 17
3.3.3.2 L298N Motor Driver (Roller Motor) 17
3.3.4 Battery Runtime and Efficiency 18
3.3.4.1 Battery 1 18
3.3.4.2 Battery 2 18
3.4 Software Design 19
3.4.1 Overview 19
3.4.2 ESP32 Control 19
3.4.3 Manual Control 20
3.4.4 Autonomous Control 20
CHAPTER 4 RESULTS AND CONCLUSION 22
4.1 Results and Conclusion 22
4.2 Limitations 23
4.3 Future Work 23
BIBLIOGRAPHY 24
APPENDIX A Cost Analysis 25
APPENDIX B Detailed Technical Drawings 26

LIST OF TABLES

Table **Page**

3.1 Components Connected to First Battery 18

3.2 Components Connected to Second Battery 18

A.1 Total Project Costs 25

LIST OF FIGURES

Figure **Page**

3.1 Assembled robot with and without the cover 5
3.2 Robot cover and lid 5
3.3 Camera Holder 6
3.4 LiDAR Holder 6
3.5 Base 6
3.6 Roller Leaf Collector 7
3.7 Side view of the roller showing nut placement 7
3.8 Opposite side of the roller 7
3.9 Roller Holder 8
3.10 Leaf collection box 8
3.11 Flexible box tip 9
3.12 Box handle 9
3.13 Box holder 9
3.14 Roller motor holder 10
3.15 Wheel motor holder 10
3.16 Wheel and Roller covers. 10
3.17 Component holder 11
3.18 Battery hatch 11
3.19 Wheel 12
3.20 Unused Component holder 12
3.21 Unused Wheel design 13
3.22 Unused Battery holder design 13
3.23 Circuit Schematic of the Robot 16
3.24 LM2576 Voltage Regulator Schematic 17
3.25 ESP32 Control Flowchart 19
3.26 Manual Control Flowchart 20
3.27 Mapping Process 20

4.1 Example of a map generated by the robot 22

B.1 Robot Cover Drawing 26
B.2 Robot Base Drawing 26
B.3 Robot Lid Drawing 27
B.4 Wheel Motor Holder Drawing 27

B.5 Roller Motor Holder Drawing 27
B.6 Roller Holder Drawing 28
B.7 Roller Drawing 28
B.8 Box Holder Drawing 29
B.9 Box Drawing 29
B.10 Roller and Wheel Cover Drawings 29

CHAPTER 1

INTRODUCTION

1.1 Background and Problem

Urban environments generate substantial amounts of organic waste in the form of fallen leaves and tree debris. In the United States alone, approximately 40.2 million tons of leaf biomass are produced annually, contributing to the broader issue of urban waste management. Of this, an estimated 28 million tons of leaves drop each year, representing a significant ecological and economic concern if not properly managed. Leaves are rich in essential nutrients such as nitrogen, phosphorus, and potassium—worth an estimated 551 million USD per year—and improper disposal can lead to nutrient leaching, which negatively impacts soil and water systems [4].

When accumulated on urban surfaces, leaf litter can substantially elevate concentrations of dissolved phosphorus and nitrogen in stormwater runoff, accelerating eutrophication and causing excessive algae growth in nearby water bodies. Studies have demonstrated that without proper leaf removal, as much as 56% of the annual total phosphorus yield originated from fallen leaves, whereas active collection programs reduced this value to 16%. Implementation of municipal leaf collection and street cleaning programs achieved reductions of 83–84% in phosphorus and 71–74% in nitrogen loads, underscoring the environmental importance of effective leaf management [5].

Despite their high nutrient value, leaves are often treated as waste rather than as a recoverable resource. Traditional disposal methods, such as open dumping or burning, not only waste valuable nutrients but also contribute to air pollution and soil degradation. According to Nowak et al. [4], approximately 13.5% of total municipal solid waste in the U.S. consists of yard trimmings, nearly a quarter of which are fallen leaves. While 22 states have established regulations for yard trimming disposal, many municipalities still lack efficient collection systems or sustainable reuse methods.

Given these environmental and economic implications, efficient leaf management is essential not only at the municipal level but also within local institutions. At King Mongkut's Institute of Technology Ladkrabang (KMITL), large quantities of leaves accumulate daily around campus lawns and pathways, especially near the HM building. The manual cleaning process requires significant labor and time. Uncollected leaves can obstruct drainage systems and affect the aesthetics of public areas. Existing commercial leaf-collecting robots are typically expensive and lack adaptability to specific environments, such as campus lawns. Therefore, an affordable, autonomous, and effective robotic system capable of navigating flat terrain, and collecting leaves efficiently is needed. To address this, the project proposes the design and development of a leaf collection robot with autonomous and manual control, integrating electronics, sensors, and LiDAR based SLAM to automate cleaning tasks and optimize environmental management.

1.2 Objectives

- **Automated Leaf Collection**: Develop a system to automate the leaf collection process, reducing manual labor.
- **Navigation and Control**: Develop an autonomous navigation system using LiDAR and SLAM for mapping and localization while also providing a manually controlled option.
- **Mechanical Design**: Design and implement an efficient leaf roller and storage system capable of col-

lecting dry or slightly moist leaves.

- **Circuit Design**: Design and build an electronic circuit system for motor control, power regulation, and communication between hardware components.

1.3 Scope

The system operates under the following constraints:

- **Terrain Limitation:** Designed for operation on flat and even surfaces such as grass lawns or smooth outdoor pavements.
- **Leaf Condition:** Optimized for collecting dry or slightly moist leaves.
- **Navigation Range:** Autonomous navigation is based on pre-defined LiDAR-generated maps with an effective sensing range of up to 12 meters.
- **Obstacle Detection:** Employs real-time LiDAR feedback for obstacle detection and avoidance.
- **Environmental Conditions:** Designed to operate under standard weather conditions.

1.4 Expected Benefits and Outcomes

- **Automation Efficiency:** Reduces manual labor and time required for leaf collection through autonomous operation and LiDAR-based navigation.
- **Environmental Impact:** Promotes proper disposal of organic waste, reducing nutrient runoff and mitigating eutrophication in nearby water bodies.
- **Cost-Effectiveness:** Provides a low-cost, locally developed alternative to expensive commercial autonomous leaf collectors.
- **System Scalability:** Modular hardware and ROS2-based architecture allow future upgrades such as AI-based vision or multi-robot collaboration.
- **Educational Value:** Enhances practical learning in robotics, control systems, and embedded software.
- **Practical Outcome:** A fully functional prototype capable of autonomous mapping, navigation, and leaf collection under standard environmental conditions.

CHAPTER 2

LITERATURE REVIEW AND RELATED THEORIES

2.1 Concepts and Theories

- **Autonomous Navigation and SLAM**

  Simultaneous Localization and Mapping (SLAM) is a method that allows robots to create maps of unknown environments while keeping track of their position within that map. LiDAR (Light Detection and Ranging) sensors are commonly used for SLAM due to their high accuracy and resistance to lighting variations. The LiDAR sensor emits laser pulses that measure the distance to surrounding objects, enabling real-time mapping and obstacle avoidance.

  In this project, the Jetson Nano processes LiDAR data through ROS2 (Robot Operating System 2) to perform mapping and navigation, while an ESP32 microcontroller handles motor control and odometry feedback.

- **Motor Control and Circuit Design**

  The motion of the robot depends on the effective operation of DC motors and their corresponding control circuits. The MDD20A dual H-bridge motor driver is employed for the encoder motors, while an L298N driver operates the roller motor. The H-bridge configuration enables bidirectional motor control using Pulse Width Modulation (PWM) for speed regulation.

  Additionally, the LM2576 step-down regulator is implemented to convert the 12V LiPo battery supply into a stable 5V DC output for powering the Jetson Nano and microcontrollers. This ensures voltage stability, protects the system components, and enhances energy efficiency.

- **Power Management and Efficiency**

  Power management plays a crucial role in ensuring stable operations across all subsystems. The use of a LiPo battery allows for compact and high-capacity power storage. The LM2576 regulator helps maintain a consistent output, reducing power loss, and protecting sensitive electronics. The battery life is carefully managed through current optimization, reducing idle power consumption by turning off unused components when the robot is stationary.

- **Leaf Collection Mechanism**

  Several leaf collection mechanisms exist, with the three most common being vacuum, blower, and roller systems. A vacuum mechanism sucks leaves into a collection container, while a blower directs them toward a designated area. A roller mechanism, on the other hand, uses a brush to sweep leaves into a box, similar to human cleaning. After analyzing each design, we chose a roller-based system, as vacuum mechanisms can clog with dust and debris, and blower systems did not meet our requirements.

2.2 Related Works

Several studies have explored the development of autonomous cleaning robots using LiDAR and motor control circuits:

- Natsu and Hisashi [3] developed an autonomous blower robot that used a three-wheel omnidirectional caster system, allowing it to move smoothly in any direction. The robot used LiDAR-based mapping and localization (LeGO-LOAM and HDL-Localization) for real-time position tracking and automatic path following, achieving over 98% cleaning efficiency in outdoor tests.
- Hsia et al. [1] designed a smart leaf-blowing robot that used a ResNet50-based CNN model to detect and classify leaves in real time. It relied on ultra-wideband (UWB) localization with three anchors to get precise positioning. The robot's controller automatically adjusted the blower and motors based on the camera's input, helping it clean more effectively.
- Miao, Zhang, and Wang [2] proposed a Non-Maximum Fusion (NMF) algorithm to improve how robots detect leaves. Instead of removing overlapping detection boxes like in traditional methods, NMF merges nearby boxes to better detect dense or overlapping leaves. This makes path planning easier and improves overall cleaning efficiency.
- A recent study in Processes (2024) [6] presented a small leaf collection and recycling vehicle that uses a dual-motor drive system and a mechanical leaf-crushing and storage unit. Its intelligent control system allows smooth gear shifting, regenerative braking, and energy-efficient operation.

CHAPTER 3

METHODOLOGY AND DESIGN

3.1 Hardware Design

Our robot was designed using Computer-Aided Design (CAD) to create efficient and well-optimized components that work seamlessly together to achieve the best possible performance. The design is intended to be compact and mobile, while also being easy to repair and upgrade. Detailed technical drawings of each component can be found in Appendix B.

3.1.1 Assembly

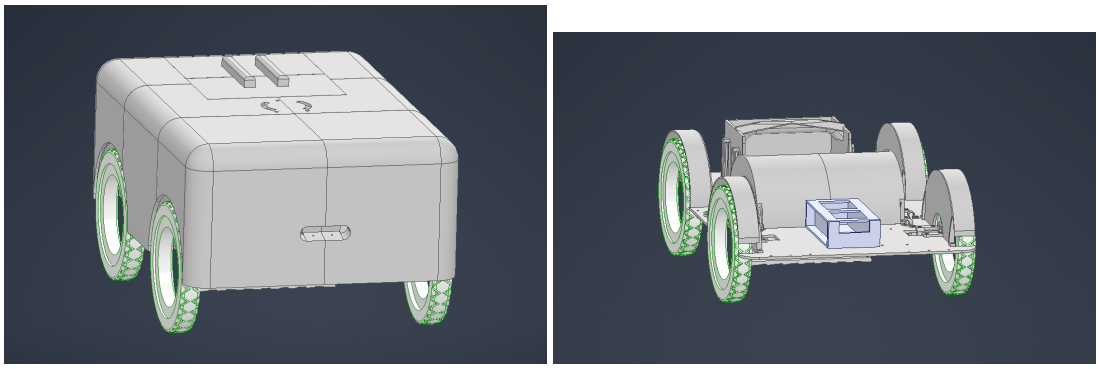


Figure 3.1: Assembled robot with and without the cover

This image shows the final CAD design after all components have been assembled, both with and without the cover.

3.1.2 Cover & Lid

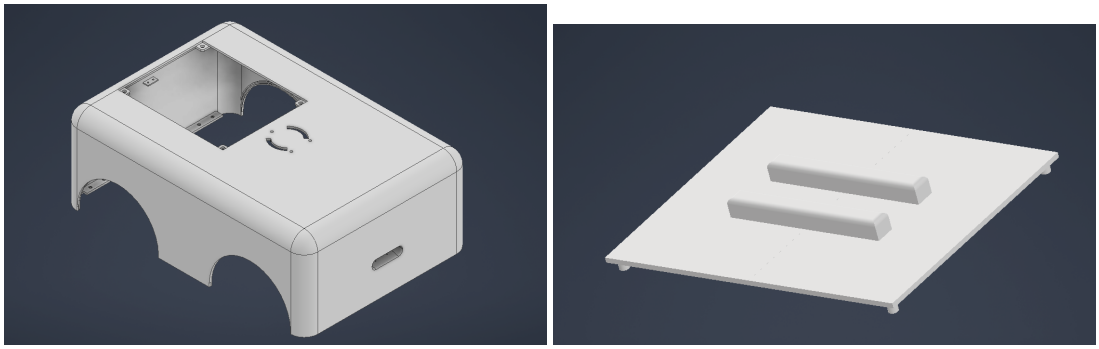


Figure 3.2: Robot cover and lid

These images show the CAD models of the cover and lid, which protect the robot from dust and debris. Silicone and an additional top layer can be applied to make the cover rainproof.

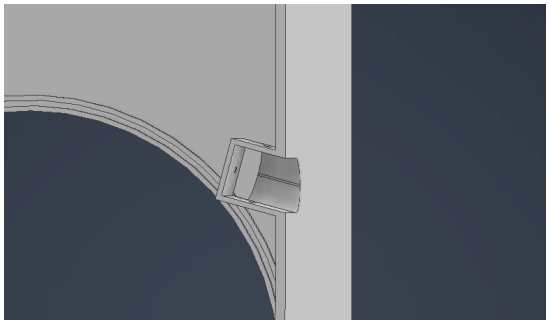


Figure 3.3: Camera Holder

This image shows the design of the camera holder on the cover. It is positioned at a —15° angle to achieve the optimal view for the 3D camera.

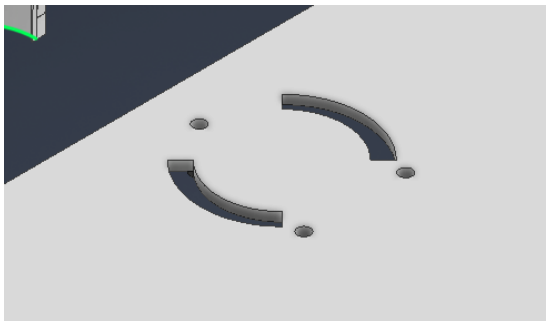


Figure 3.4: LiDAR Holder

This image shows the LiDAR holder design. The LiDAR will be mounted here, and an additional cover can be added to protect it from water, making the setup weather-resistant.

3.1.3 Base



Figure 3.5: Base

This image shows the CAD model of the base, which is made of aluminum. It supports all other components and provides structural stability.

3.1.4 Roller

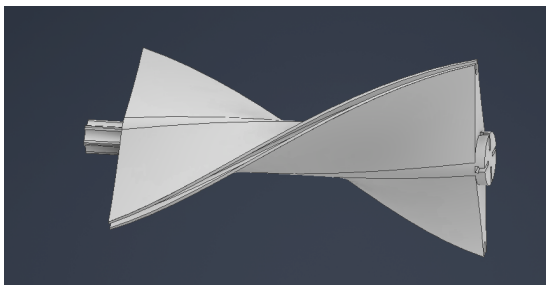


Figure 3.6: Roller Leaf Collector

The roller is designed to collect leaves using attached plastic strings that sweep them into the collection box. It was split into 2 components for easier printing.

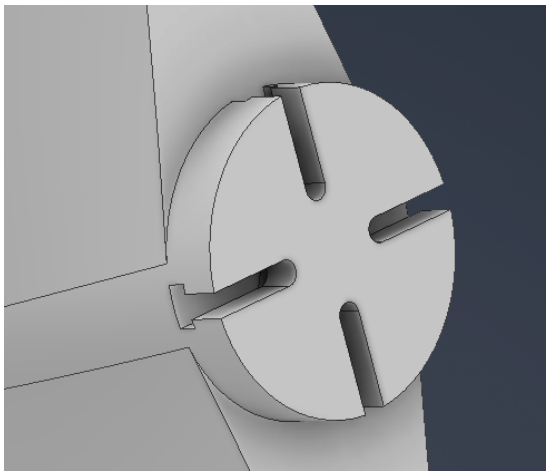


Figure 3.7: Side view of the roller showing nut placement

This side of the roller features four nut holes instead of a central screw to securely connect it to the shaft.

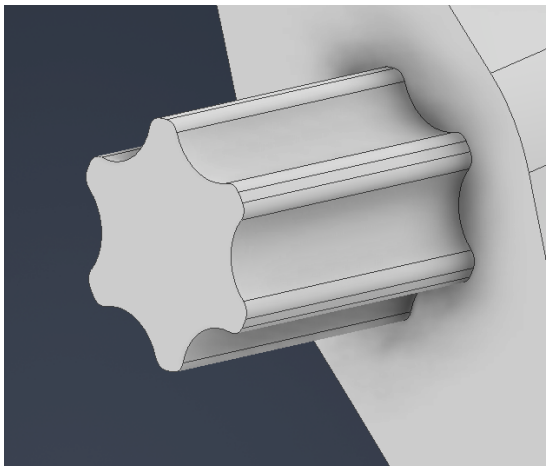


Figure 3.8: Opposite side of the roller

The opposite side of the roller is designed to minimize friction by reducing the surface area in contact with the roller holder.

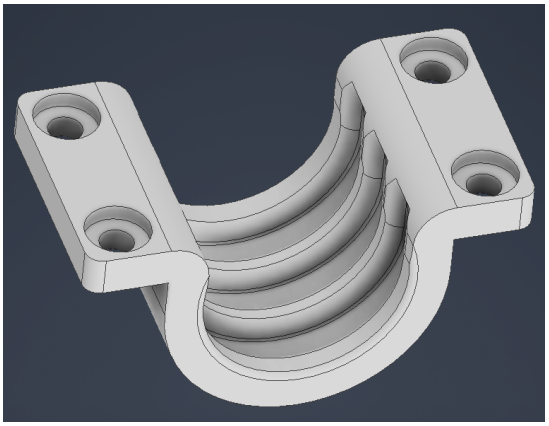


Figure 3.9: Roller Holder

This component holds the roller in place and ensures smooth rotation during operation.

3.1.5 Box

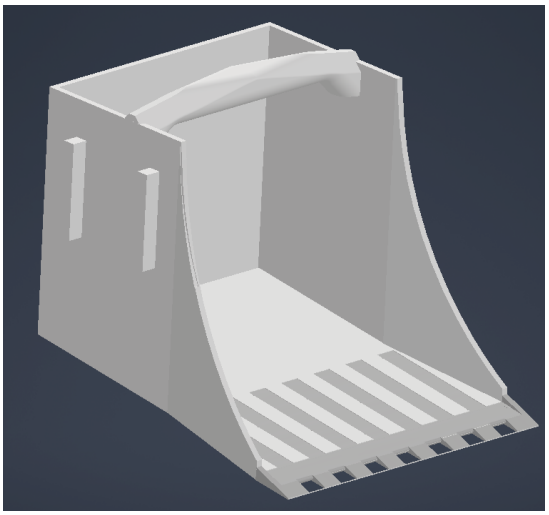


Figure 3.10: Leaf collection box

This image shows the CAD model of the collection box. It stores the collected leaves and can be easily detached for cleaning and disposal.

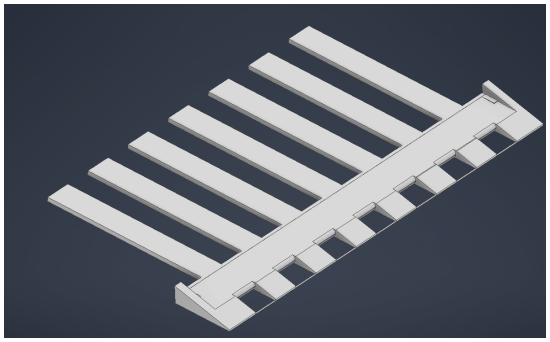


Figure 3.11: Flexible box tip

The flexible tip is made from a softer plastic material to prevent breakage when close to the ground. It can also be redesigned to adapt to different terrain conditions.

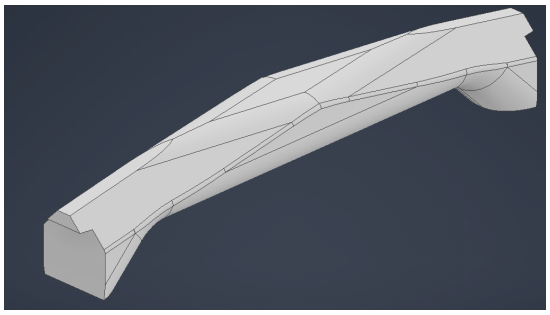


Figure 3.12: Box handle

This handle is attached to the box to make it easier to pull out from the holder.

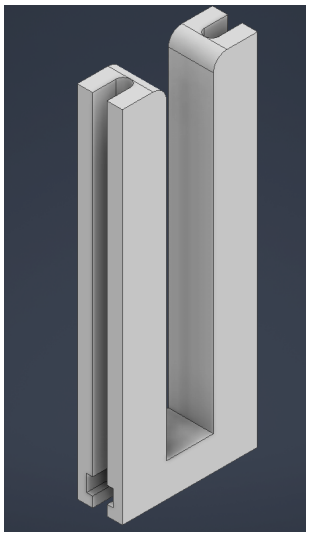


Figure 3.13: Box holder

The box holder secures the collection box to the base. It offers three adjustable height levels, each differing by 5 cm, to fine-tune the collection position.

3.1.6 Motor Holders

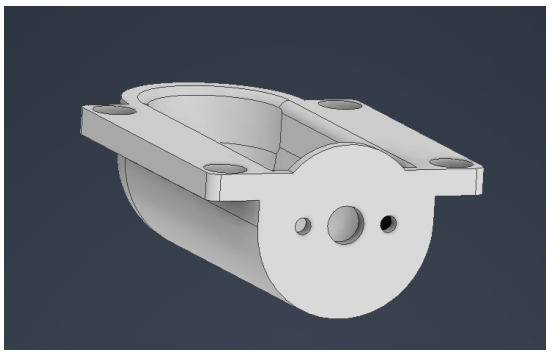


Figure 3.14: Roller motor holder

This image shows the CAD design for the roller motor holder.

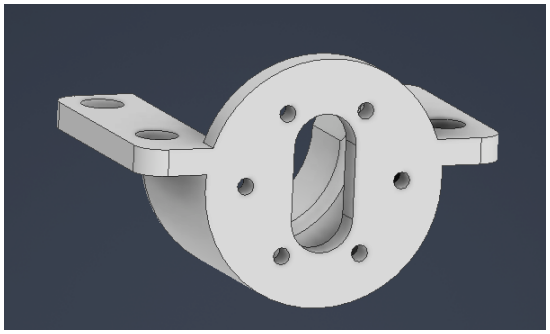


Figure 3.15: Wheel motor holder

This image shows the CAD design for the wheel motor holder.

3.1.7 Wheel & Roller Covers

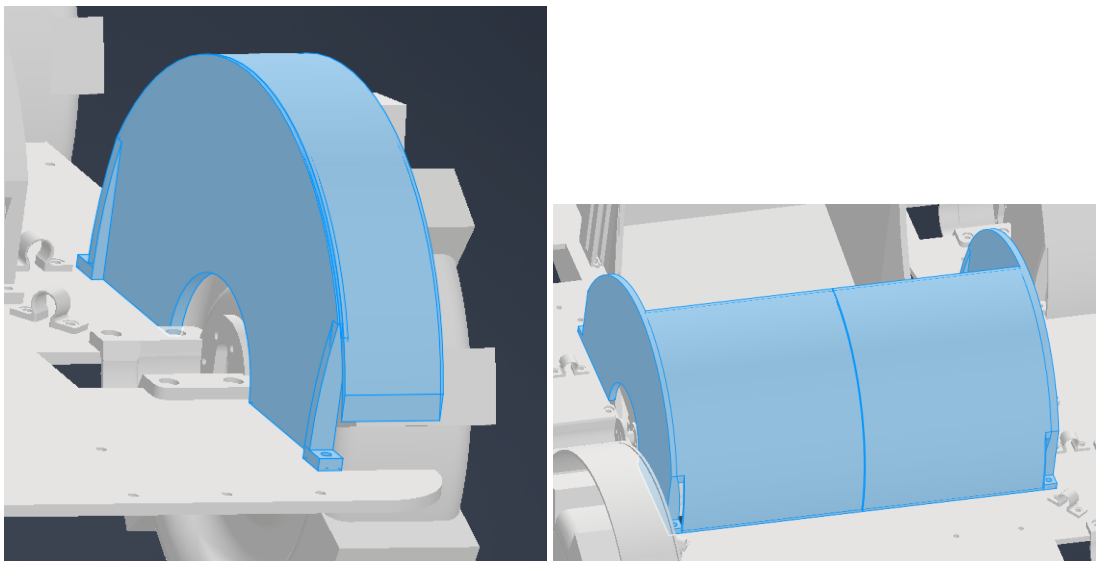


Figure 3.16: Wheel and Roller covers.

These covers protect the wheel and roller from dirt and debris, keeping the interior components clean and functioning properly.

3.1.8 Component Holder

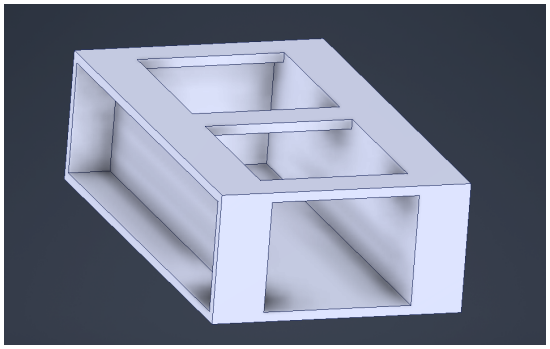


Figure 3.17: Component holder

The component holder is used to mount electronic boards and the battery securely within the robot.

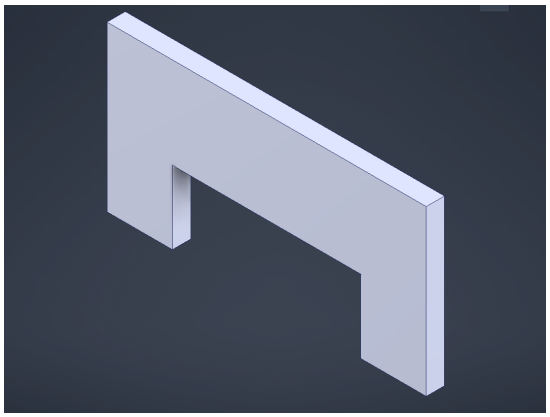


Figure 3.18: Battery hatch

The hatch covers the battery compartment to prevent it from falling out during movement or vibration.

3.1.9 Wheel

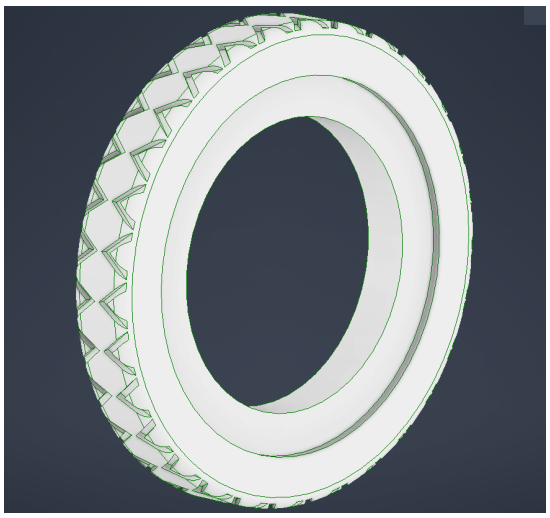


Figure 3.19: Wheel

The wheel has a large diameter to provide stability and smooth movement on rough or uneven terrain, while supporting the full weight of the robot. It uses an aluminum rim for added strength and durability.

3.1.10 Unused Parts

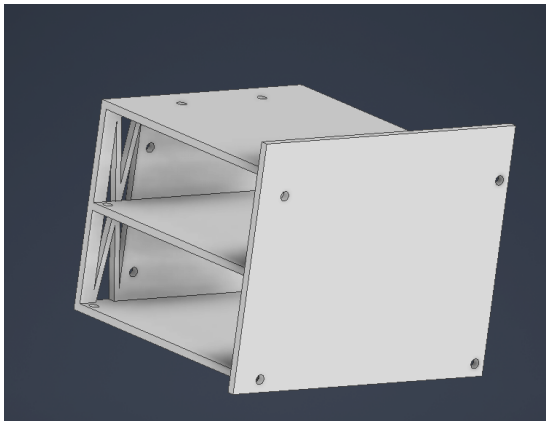


Figure 3.20: Unused Component holder

This component holder was designed to organize all electronic boards in a single location. However, it was not used due to the complex and time-consuming assembly process.

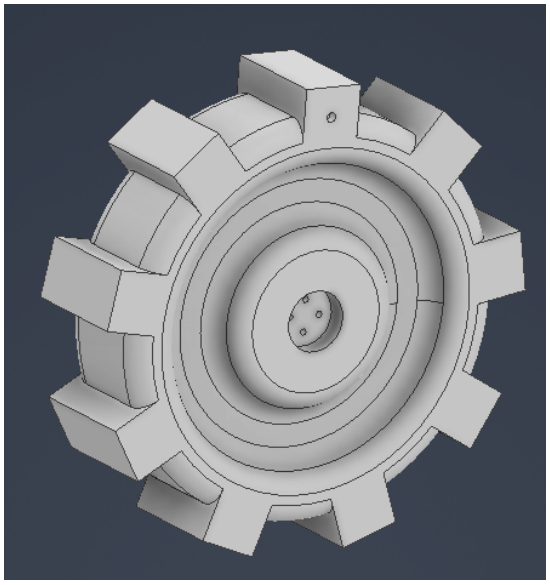


Figure 3.21: Unused Wheel design

This wheel was designed with deep grooves to improve traction on both firm and soft surfaces. However, the groove edges were not rounded, causing excessive resistance, and the motor lacked sufficient torque to rotate it effectively.

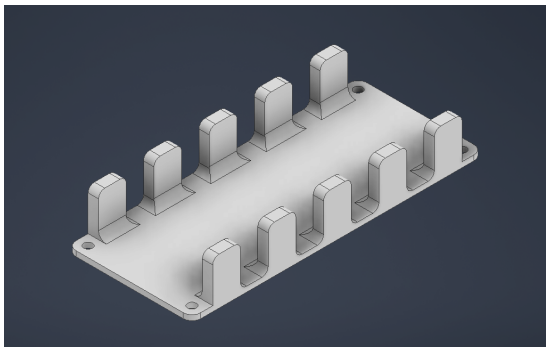


Figure 3.22: Unused Battery holder design

The original battery holder design was excluded because the final build required additional batteries, making the holder incompatible with the new configuration.

3.2 Component Selection Rationale

In designing the circuit and system for the autonomous leaf collection robot, each hardware component was selected based on its electrical characteristics, compatibility, efficiency, and cost-effectiveness. The system required a balance between performance, power management, and ease of implementation. A list of all components used and their costs is provided in Appendix A.

3.2.1 Motor Selection

3.2.1.1 Encoder DC Motor – JGB37-520

The encoder DC motors are responsible for the robot's wheel movement and navigation. We chose the 12V, 37 RPM, 30 kg·cm torque variant because it offers:

- High torque (30 kg·cm) at a relatively low speed (37 RPM), suitable for moving a medium-weight robot chassis without slippage.
- Integrated encoder for precise rotation measurement, enabling accurate odometry feedback used by the control system for SLAM-based navigation.
- 12V operating voltage, aligning perfectly with the robot's Li-Po battery system and compatible with the MDD20A MOSFET driver.
- Low current draw, allowing extended runtime without overloading the battery.

The low RPM ensures smooth and controllable motion, which is crucial for mapping and obstacle avoidance tasks. Higher-speed motors were avoided as they could cause instability and increased power consumption.

3.2.1.2 Roller Motor – 25GA-370

The roller motor powers the leaf collection mechanism that sweeps and stores leaves into the container. The variant with 12V, 100 RPM, 5 kg·cm Torque was selected for the following reasons:

- Moderate speed (100 RPM) provides sufficient sweeping action without dispersing leaves.
- 12V operation simplifies power distribution with the same supply line as the drive motors.
- Simple control requirement — only ON/OFF switching via the L298N driver, reducing circuit complexity.
- Small current requirement (0.06 A), ensuring minimal impact on the total power budget.

This motor's torque–speed balance ensures reliable collection on grass surfaces while maintaining energy efficiency.

3.2.2 LiDAR Sensor

The RPLiDAR A1 was chosen as the main distance sensor for environmental mapping due to its:

- 360° scanning coverage with up to 12-meter detection range.
- High scan frequency (5–10 Hz) providing dense data points for accurate mapping.
- Compact and low-power design (5V, 500 mA), making it easy to integrate with the existing power system.
- Full compatibility with ROS2 and Jetson Nano, allowing direct integration with the SLAM algorithm.

Compared to ultrasonic or infrared sensors, LiDAR provides superior accuracy, faster response, and reliable distance readings even in varying lighting conditions. This ensures robust obstacle detection and path planning without needing external vision systems.

3.2.3 Microcontrollers and Processing Units

3.2.3.1 ESP32 Microcontroller

The ESP32 was selected as the main motor controller for several key reasons:

- Built-in PWM generators and interrupt handling, suitable for real-time motor speed control.
- Low power consumption (average 160 mA active mode), preserving battery life.
- UART communication interface allows reliable serial data exchange with the Jetson Nano.

Additionally, the ESP32 supports Micro-ROS, enabling smooth message passing with the ROS2 system, making it ideal for embedded real-time control within the robot.

3.2.3.2 Jetson Nano

The Jetson Nano was selected as the main processing unit for mapping, navigation, and sensor data handling due to its balance of performance, power efficiency, and compact design.

- Sufficient power for real-time LiDAR mapping and control.
- Supports ROS2 and AI libraries (OpenCV, TensorFlow) for easy software integration.
- Operates efficiently within 5V–10W, suitable for the 12V Li-Po power system.
- Compact 69×45 mm design fits easily within the robot chassis.

The Jetson Nano serves as the robot's main computational unit, processing sensor data and coordinating navigation with the ESP32 controller.

3.2.4 Power Components and Safety Elements

3.2.4.1 LM2576 Voltage Regulator

The LM2576 was chosen over linear regulators such as the LM7805 due to its switching efficiency (≈80–90%) and ability to handle up to 3A output current. Key selection reasons:

- Efficient power conversion, minimizing battery heat loss.
- Compact design with minimal external components.
- Proven reliability and stability for continuous load operation.

The inclusion of Schottky diodes, smoothing capacitors, and inductors ensures clean output voltage for sensitive logic components like the Jetson Nano and ESP32.

3.3 Circuit Design

The electronic circuit of the Robot is designed to integrate all major subsystems including power supply, motor control, and communication interfaces. The circuit ensures stable power delivery, efficient motor operation, and seamless data transfer between the Jetson Nano and ESP32 microcontroller.

The design is modular, consisting of three main parts:

1. Power Regulation and Distribution Circuit
2. Motor Driver and Control Circuit
3. Microcontroller Interface Circuit

Each module was carefully selected and tested to maintain optimal performance under continuous operation while ensuring safety and energy efficiency.

3.3.1 Circuit Schematic

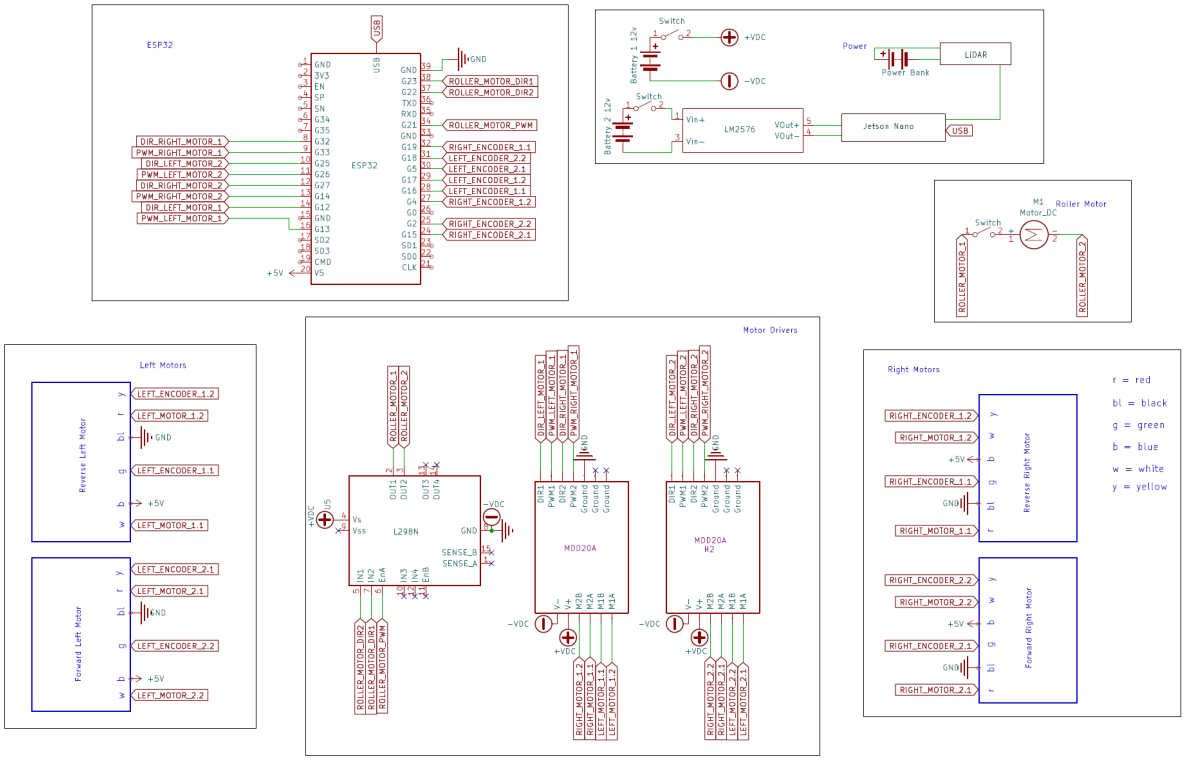


Figure 3.23: Circuit Schematic of the Robot

3.3.2 Power Regulation and Distribution

The robot is powered by two 12V Lithium Polymer (Li-Po) batteries, chosen for their high energy density and ability to provide stable current under variable load conditions. Two batteries were chosen due to the high current demand of the components. One battery is used for the motor drivers while the other one is used for the Jetson Nano and ESP32. The LiDAR module was powered by an additional power bank. The roller motor, Jetson Nano and Motor Drivers each have their own switch allowing for independent control.

Since different components of the robot require different voltage levels, a DC-DC step-down - LM2576 regulator was implemented.

1. Input Voltage: 12V (from Li-Po)
2. Output Voltage: 5V (for Jetson Nano and logic-level electronics)
3. Switching Frequency: 52 kHz

The LM2576 regulates voltage using a switching principle:

- During the ON cycle, the internal MOSFET conducts, allowing current to flow from the input through the inductor to the load.
- During the OFF cycle, the inductor maintains current flow via the Schottky diode.
- The feedback voltage divider adjusts the duty cycle dynamically to maintain a constant 5V output.

Capacitors are added at both input and output terminals to smooth voltage fluctuations, while the inductor

ensures stable current delivery. This configuration minimizes power loss and heat generation, allowing for prolonged battery operation. The schematic of LM2576 can be seen below:

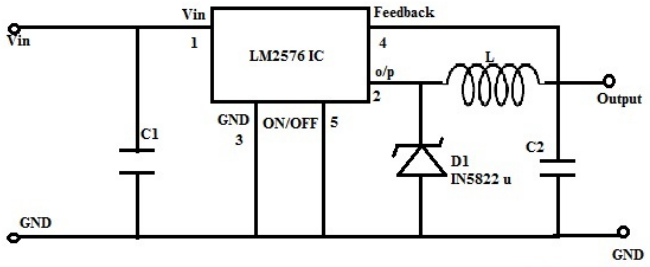


Figure 3.24: LM2576 Voltage Regulator Schematic

3.3.3 Motor Control Circuit

The robot uses two distinct motor driver circuits to handle movement and collection mechanisms.

3.3.3.1 MDD20A Dual H-Bridge Driver (Wheel Motors)

The main wheel motors are controlled using the MDD20A dual H-Bridge driver, which supports two encoder DC motors. This motor driver uses 2 H-Bridges consisting of 8 N-channel MOSFETs each: 2 MOSFETs are used in parallel for each switch in the H-Bridge to reduce the heat as it can supply a high current of 20A. In addition, a Schottky diode is connected in parallel to each pair of MOSFETs to provide a path for the inductive current to flow.

Operation Principle:

- The H-Bridge allows current to flow in both directions through the motor, enabling forward and reverse motion.
- The PWM signals from the ESP32 control motor speed by adjusting the duty cycle of the input DC signal which controls the frequency of the MOSFETs.
- Encoder feedback from each motor is read by the ESP32 to calculate rotational speed and position, enabling precise odometry for navigation.

A MOSFET-based motor driver was chosen over a BJT-based one as it offers the following advantages:

- Low $R_{DS(on)}$, reducing heat generation.
- High switching speed suitable for high-frequency PWM control.
- Greater efficiency than traditional BJT-based drivers.

This circuit forms the foundation of the robot's movement system, allowing smooth speed transitions and accurate path following.

3.3.3.2 L298N Motor Driver (Roller Motor)

For the roller mechanism that collects leaves, a simpler L298N motor driver is used. This module controls a single 12V DC motor (25GA-370) that operates at 100 RPM, providing sufficient torque to sweep and collect leaves effectively. Unlike the wheel motors, the roller motor does not require precise speed or directional control—only ON/OFF operation is needed.

The L298N is ideal for this purpose due to its:

- Integrated H-Bridge configuration,

- Built-in protection diodes, and
- Easy-to-use logic control interface.

Although the L298N has higher power losses than MOSFET-based drivers, it remains cost-effective and suitable for low-current applications like the roller motor (draws approximately 0.06 A).

3.3.4 Battery Runtime and Efficiency

3.3.4.1 Battery 1

Table 3.1: Components Connected to First Battery

Component	Voltage (V)	Current (A)	Power (W)	Quantity	Total Power (W)
Encoder Motor Drivers	5	1	5	2	10
Encoder Motors	12	1	12	4	48
DC Motor Driver	12	0.083	1	1	1
DC Motor	12	0.185	2.22	1	2.22
Total					**61.22**

$$\text{Power Efficiency} = 95\%$$

$$P_{in} = \frac{61.22}{0.95} = 64.44\ \text{W}$$

$$I_{draw} = \frac{P_{in}}{V_{battery}} = \frac{64.44}{11.1} = 5.81\ \text{A}$$

$$\text{Battery Capacity} = 2.2\ \text{Ah}$$

$$\text{Battery Run Time} = \frac{2.2}{5.81} \times 60 = 22.74\ \text{min.}$$

3.3.4.2 Battery 2

Table 3.2: Components Connected to Second Battery

Component	Voltage (V)	Current (A)	Power (W)	Quantity	Total Power (W)
Jetson Nano	10	1	10	1	10
ESP32	3.3	0.15	0.495	1	0.495
Total					**10.495**

$$\text{Power Efficiency} = 95\%$$

$$P_{in} = \frac{10.495}{0.95} = 11.047\ \text{W}$$

$$I_{draw} = \frac{P_{in}}{V_{battery}} = \frac{11.047}{11.1} = 0.995\ \text{A}$$

$$\text{Battery Capacity} = 2.2\ \text{Ah}$$

$$\text{Battery Run Time} = \frac{2.2}{0.995} \times 60 = 132.66\ \text{min.}$$

3.4 Software Design

3.4.1 Overview

The software of our robot uses the ROS2 framework to control the robot which enables efficient navigation and communication between components. It provides real-time and reliable communication and works well across different hardware such as the ESP32 and Jetson Nano. It consists of three main parts: the ESP32, which controls the motors; the Jetson Nano, which provides manual or autonomous control; and the communication layer between them.

3.4.2 ESP32 Control

The ESP32 is responsible for controlling the speed and direction of the motors ensuring the robot moves as intended. The Micro-ROS library is used to allow the ESP32 to integrate within the ROS framework and act as a node. It connects to the motor drivers and sends velocity and direction data in the form of PWM signals. A flowchart of the system can be seen below:

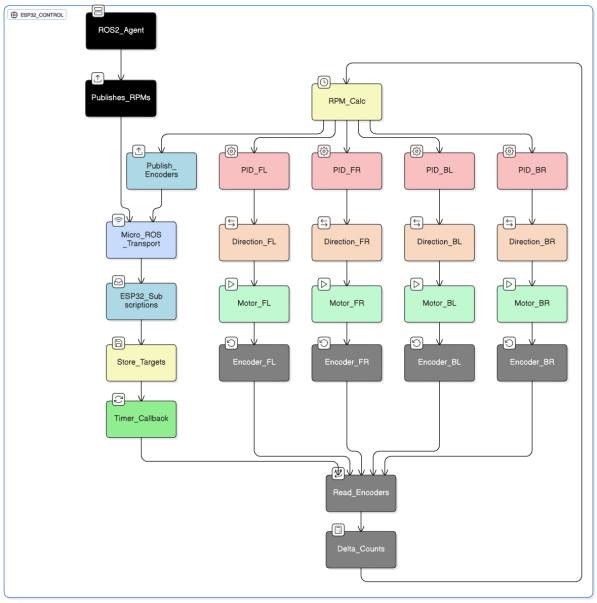


Figure 3.25: ESP32 Control Flowchart

First, the ROS2 agent running on the Jetson Nano publishes the RPM values to the respective topics for each motor. The ESP32 then subscribes to these topics and receives the RPM values over a UART connection via the Micro-ROS framework. Then, it stores these target values in local variables and calls the timer callback for each motor. After this, the values from the encoder are read and converted to RPM values using the following formula:

$$\text{RPM} = \frac{\text{P}}{\text{PPR} \times \text{t}} \times 60 \tag{3.1}$$

PPR is the pulses per revolution of the encoder, t is the time interval between readings and P is the change in encoder counts. The time interval is set to 20 ms. Then, these RPM values are published back to the ROS2 topics for each motor and used for the PID controller to adjust the PWM and direction signal ensuring the motor moves as intended.

A PID controller is used to maintain the desired RPM by adjusting the PWM signal sent to the motor driver. The proportional term provides an immediate correction proportional to the error and increases the motor's responsiveness.

The larger the difference in RPM, the stronger the correction will be. The integral term is used to eliminate any steady-state errors, ensuring the RPM value stays close to the target value. After conducting a series of tests by tuning the PID values and comparing the RPM values, a K_p value of 0.1 and a K_i value of 0.02 were chosen to achieve the target values and a derivative term was not needed. The PID controller generated a PWM value between -255 and 255, which was then sent to the motor driver to control the speed of the motors. This feedback loop continues until the target RPM value is achieved.

3.4.3 Manual Control

The robot is controlled manually by a keyboard interface using ROS2. A series of ROS2 nodes and the Micro-ROS communication convert the user inputs into commands for the motors to move. An overview of the system can be seen below.

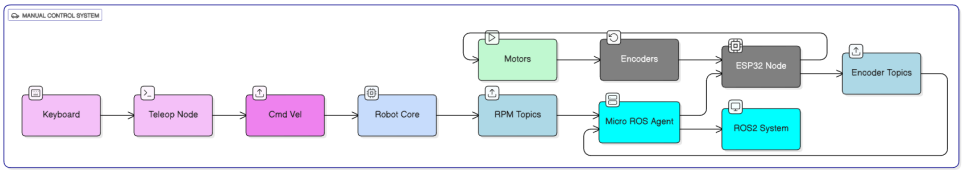


Figure 3.26: Manual Control Flowchart

When the user presses a key on the keyboard, the *teleop twist keyboard* node reads the input and publishes velocity and direction commands on the *cmd vel* topic, which contain the desired linear and angular velocity for the robot. The *Robot Core* node subscribes to the *cmd vel* topic and converts these commands into RPM values for each motor and publishes them to the respective motor nodes. Finally, the Micro-ROS agent sends these commands to the ESP32 via a serial UART interface and the ESP32 controls the motors accordingly.

3.4.4 Autonomous Control

For the autonomous control, two main steps are carried out. First, a map of the area in which the robot will move is generated using the LiDAR sensor data. This map is then used for path planning and obstacle avoidance. Then, we create a path for the robot to move around in the map.

During the mapping process, a 2D map of the surroundings is created using data from both the LiDAR sensor and the wheel encoders. Laser-based odometry is combined with Simultaneous Localization and Mapping (SLAM) to build the map. An example of the map generated can be seen in Figure 4.1.

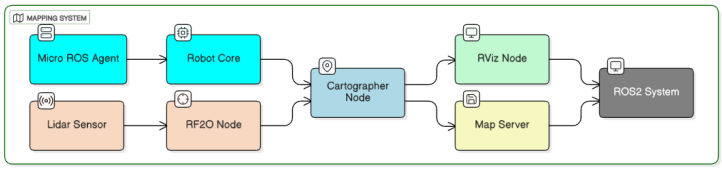


Figure 3.27: Mapping Process

The Micro-ROS agent sends the encoder data to the *Robot Core* node, which processes the data to determine how far the robot has moved and in which direction. In addition, the LiDAR sensor collects distance measurements from the surroundings and publishes them to the *scan* topic. The *RF2O* node processes these measurements to estimate the robot's motion and get a sense of the surroundings. Both the wheel odometry and the LiDAR odometry data are then

sent to the *Cartographer* node, which performs SLAM to build a 2D occupancy grid showing free areas and obstacles in the environment. RViz is used to visualize the map being generated in real time as well as the position of the robot. Once the map is fully built and satisfactory, the *map server* CLI tool is used to save the map for later use.

After mapping is completed and the environment map has been saved, the robot enters the autonomous navigation phase. In this stage, the *Nav2* stack is launched to enable path planning, localization, and motion control. The system uses the previously built map to determine the robot's position, calculate global and local paths, and send velocity commands to move toward a given goal. The *Robot Core* node subscribes to these velocity commands from Nav2 via the *cmd vel* topic and converts them into wheel speed targets for motion execution.

The navigation stack continuously monitors the robot's pose, dynamically updates the path when new obstacles are detected, and ensures smooth trajectory tracking. Through RViz, goal points can be set on the map and the robot's real-time movement, planned paths, and obstacle avoidance behavior can be visualized. Once a goal is reached, Nav2 maintains localization and waits for the next destination command.

CHAPTER 4

RESULTS AND CONCLUSION

4.1 Results and Conclusion

Our robot was able to complete all of its main objectives successfully. It could map the designated area using LiDAR and navigate through the environment with moderate accuracy. While the mapping process worked in controlled indoor conditions, some inconsistencies were observed in outdoor environments due to uneven surfaces and varying light conditions. An example of the map generated is shown below:

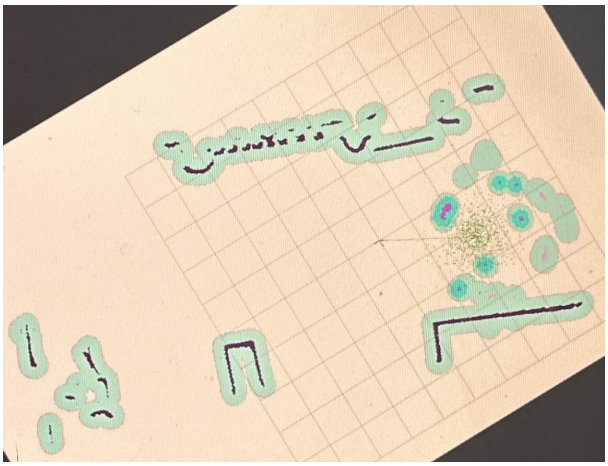


Figure 4.1: Example of a map generated by the robot

Manual control of the robot, on the other hand, worked exceptionally well. The robot responded promptly to keyboard inputs, maintained good stability during movement, and was easy to operate in different areas of the KMITL campus. The system allowed precise maneuvering, making it highly effective for collecting leaves in both narrow and open spaces. The robot was also able to pick up and store leaves efficiently in the collection box, demonstrating proper coordination between the mechanical and control systems. A demonstration video of the robot can be seen at: *here*.

Communication between the ESP32, Jetson Nano, and sensors worked reliably, with the ROS2 framework ensuring smooth coordination between navigation, control, and motor response. The manual control mode showed immediate responsiveness with minimal delay, and the overall hardware and software integration proved stable and efficient during testing.

In conclusion, the leaf collecting robot successfully addressed the problem of fallen leaves around campus. Although the LiDAR system showed limitations in outdoor mapping, the robot's manual control mode provided consistent and dependable operation. Through the integration of electronics, sensors, and control theory, an affordable and effective solution was realized. Overall, the robot demonstrated strong potential for practical maintenance applications that can reduce manual labor and promote cleaner, greener environments.

4.2 Limitations

The robot had several limitations that affected its performance. Currently, it can only operate on flat or slightly sloped terrain. In addition, the LiDAR sensor used was limited in both accuracy and range, making it unsuitable for outdoor environments and reducing the robot's autonomous navigation capabilities. The Jetson Nano also proved to be somewhat unstable and laggy, which further impacted overall performance. Moreover, at times, leaves could fall out of the collection box, and the roller mechanism was not sufficiently adaptive to uneven surfaces.

4.3 Future Work

To address the current limitations and enhance system performance, several improvements can be considered for future development:

- Firstly, a more robust LiDAR system with higher resolution and longer detection range can be integrated to improve environmental perception and accuracy in mapping. This can be complemented by incorporating a camera module for visual feedback, enabling the robot to distinguish between terrain types, identify obstacles more precisely, and verify successful leaf collection.
- Additionally, implementing an AI-assisted navigation algorithm can enhance the robot's decision making process, allowing dynamic path planning in more complex environments. A mini-PC or Raspberry Pi 5 could replace or complement the Jetson Nano to achieve higher processing efficiency and improved thermal stability while maintaining compact form factors.
- Mechanical enhancements can also increase operational efficiency. A redesigned roller system with flexible or adaptive bristles would allow better contact on uneven surfaces such as grass or small pits. Likewise, an improved storage compartment with a sealing or compression mechanism would prevent collected leaves from falling out during movement.
- From a power perspective, upgrading to higher-capacity lithium-ion batteries or modular battery packs could extend the operational time per charge. Introducing smart battery monitoring (for voltage, temperature, and discharge rate) would further ensure safe and reliable operation. A self-charging docking station could also be developed to enable automatic recharging, reducing human intervention.
- For navigation safety, adding height and tilt sensors, such as ultrasonic or infrared range finders facing downward would allow the robot to detect curbs, potholes, or steep slopes, preventing it from getting stuck or tipping over. Integrating bump or proximity sensors around the chassis could provide additional collision protection in tight environments.

By incorporating these future enhancements, the autonomous leaf collection robot can evolve into a more intelligent, energy efficient, and reliable platform suitable for larger-scale deployment and research applications in outdoor environmental maintenance.

Bibliography

[1] Shih-Chang Hsia, Szu-Hong Wang, Jen-Yu Yeh, and Chuan-Yu Chang. A smart leaf blow robot based on deep learning model. *IEEE Access*, 11:111956–111962, 2023.

[2] Yanzi Miao, Zongwei Zhang, and Hesheng Wang. Nmf: an efficient method for detecting the fallen leaves using cleaning robots on the road. In *2021 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 986–991, 2021.

[3] Nakagawa Natsu and Date Hisashi. Development of an autonomous blower robot for cleaning up and collecting fallen leaves. In *2023 IEEE 19th International Conference on Automation Science and Engineering (CASE)*, pages 1–6, 2023.

[4] David J. Nowak, Eric J. Greenfield, and Ryan M. Ash. Annual biomass loss and potential value of urban tree waste in the united states. *Urban Forestry & Urban Greening*, 46:126469, 2019.

[5] William R. Selbig. Evaluation of leaf removal as a means to reduce nutrient concentrations and loads in urban stormwater. *Science of The Total Environment*, 571:124–133, 2016.

[6] Xiaowen Zhang, Haibin Liu, Chang Liu, Zhengyu Zhao, Wenchao Han, Lvfa Yin, and Guo Anfu. Design and analysis of small fallen leaf collection, crushing, and recycling vehicle. *Processes*, 12:2011, 09 2024.

APPENDIX A

Cost Analysis

Table A.1: Total Project Costs

Component	**Part Number**	**Quantity**	**Unit Price**	**Total Price (฿)**
Jetson Nano		1	3306	3306
Roller Motor Driver	L298N	1	60	60
Encoder Motor Driver	MDD20A	2	1180	2360
Encoder Motors	JGB37-520	4	485	1940
Roller Motor	25GA-370	1	86	86
ESP32		1	240	240
ESP32 Extender		1	56	56
Voltage Step Down	LM2576	1	95	95
Lidar	RPLidar A1	1	2500	2500
Switch		3	12	36
12V LiPo Battery		2	536	1072
Aluminum Base		1	952	952
Wheels		4	556	2225
Total				14928

APPENDIX B

Detailed Technical Drawings

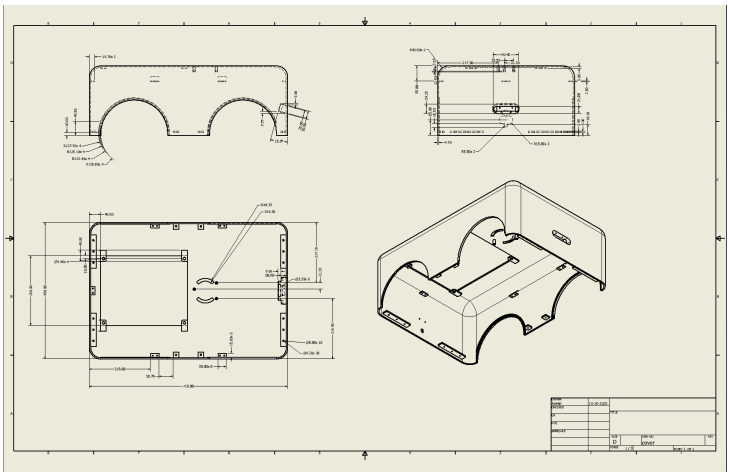


Figure B.1: Robot Cover Drawing

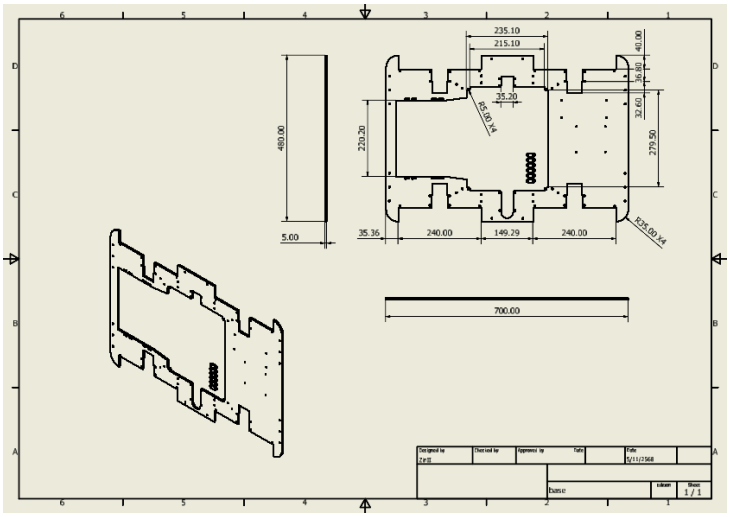


Figure B.2: Robot Base Drawing

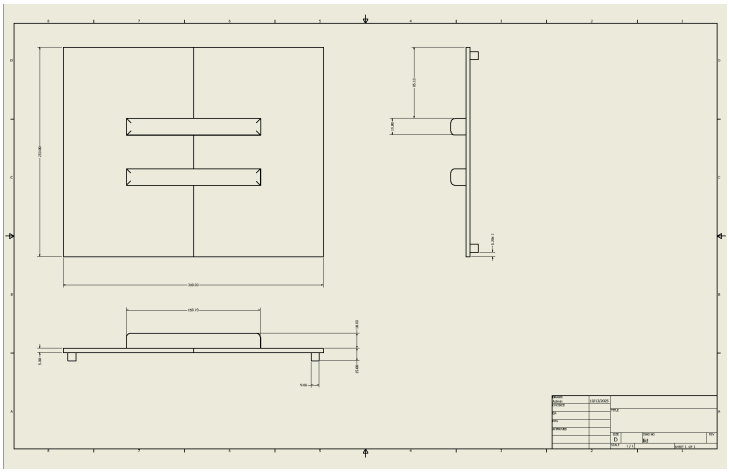


Figure B.3: Robot Lid Drawing

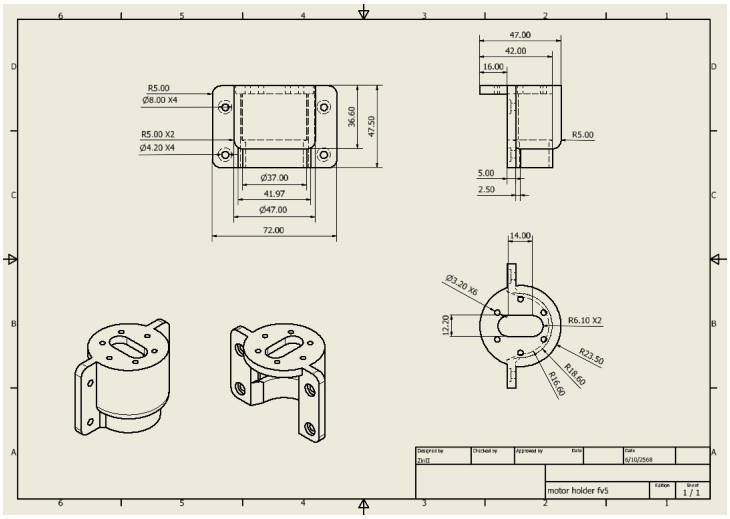


Figure B.4: Wheel Motor Holder Drawing

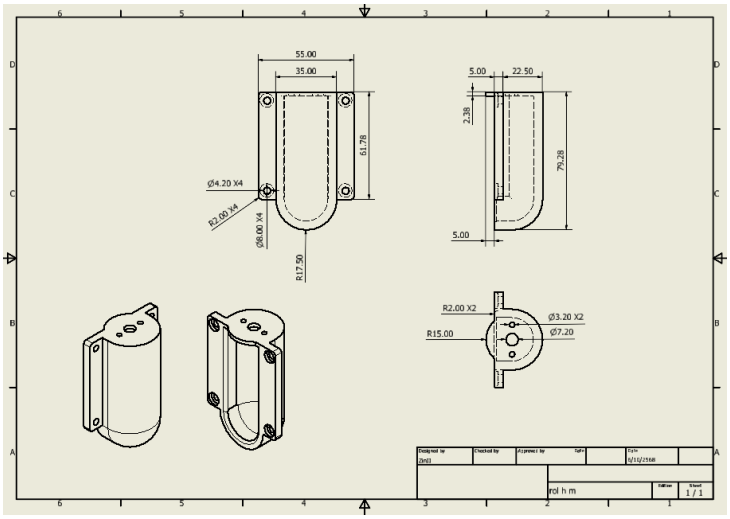


Figure B.5: Roller Motor Holder Drawing

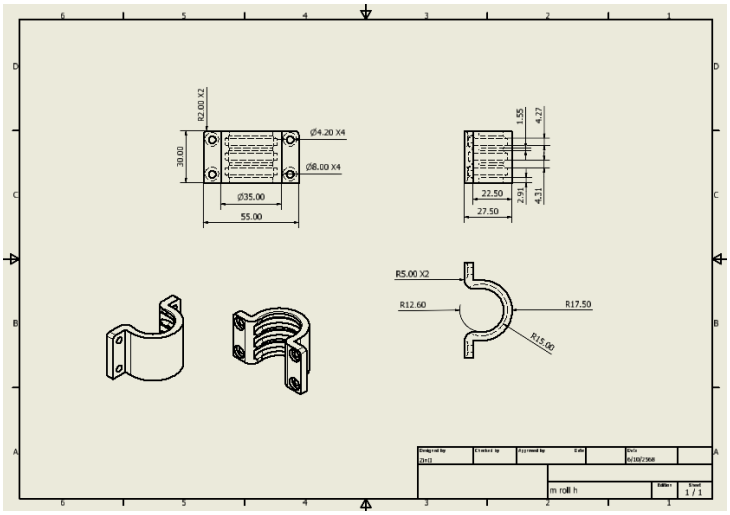


Figure B.6: Roller Holder Drawing

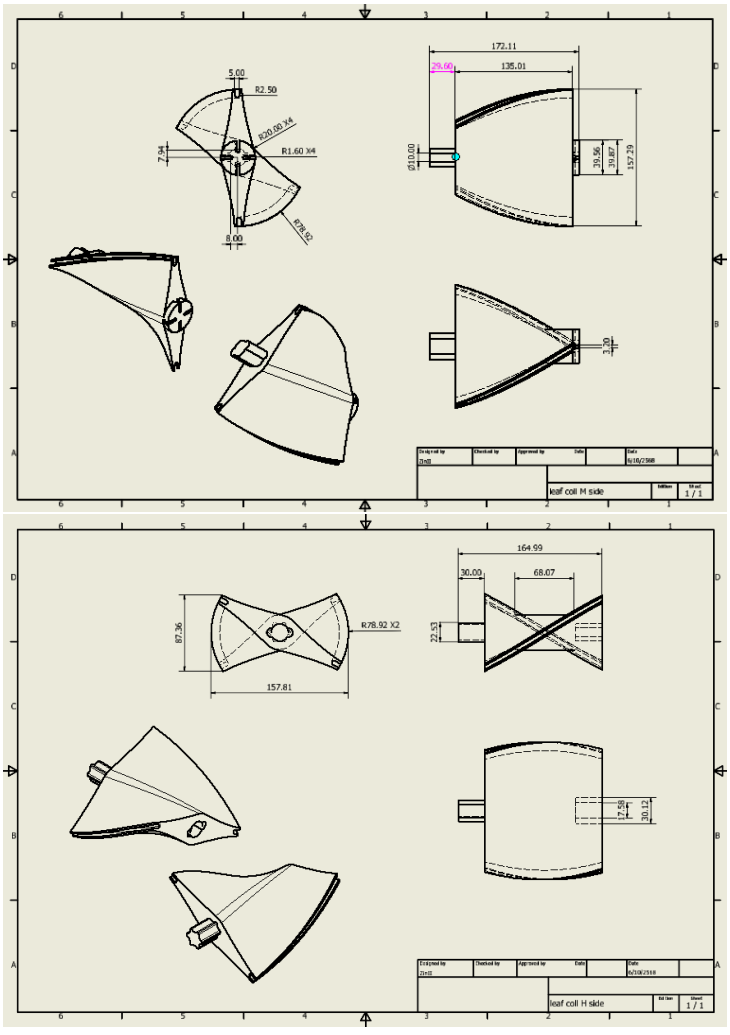


Figure B.7: Roller Drawing

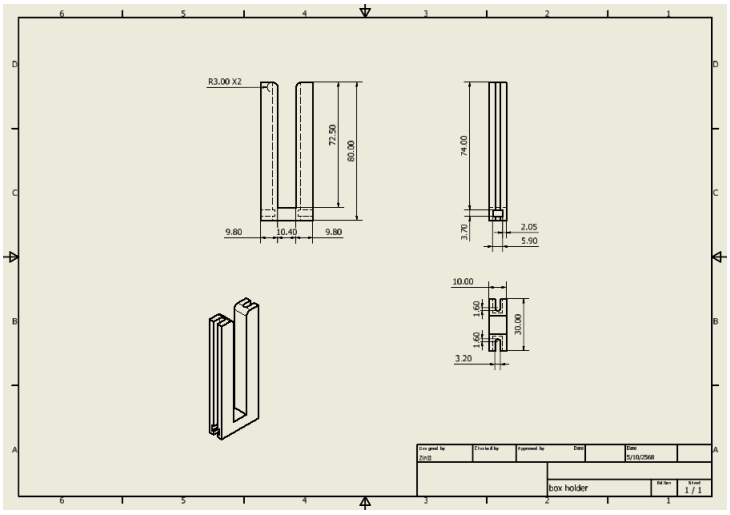


Figure B.8: Box Holder Drawing

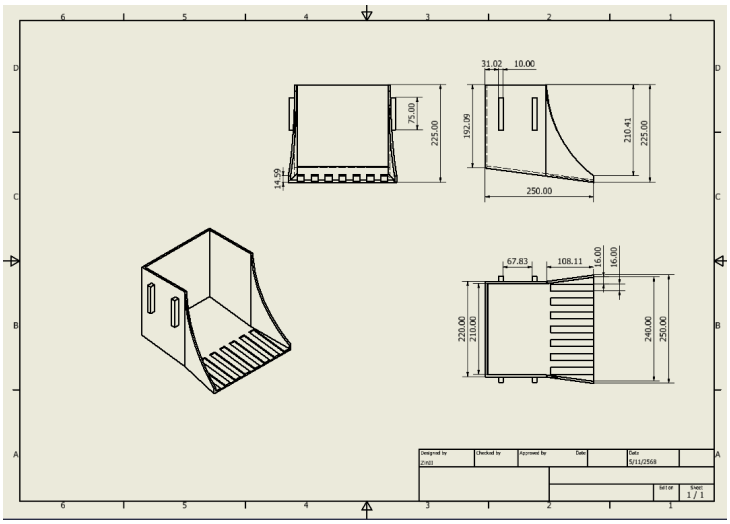


Figure B.9: Box Drawing

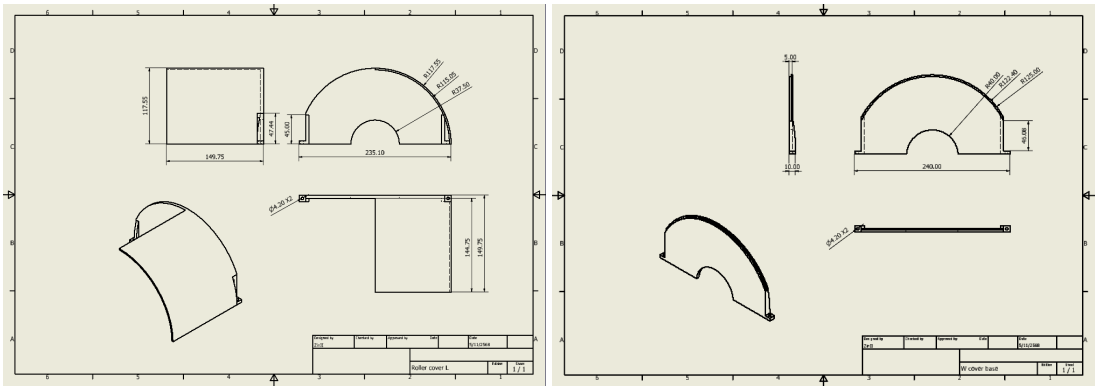


Figure B.10: Roller and Wheel Cover Drawings