



# Autonomous Vacuum-Cleaning Robot: A Comprehensive Approach to Smart Home Cleaning Solutions

Folasade Durodola<sup>1</sup>, Samuel Owwoye<sup>1</sup>, Ayanlowo Joshua<sup>1</sup>, Afolabi Zacchaeus<sup>1</sup>, Paul Wisdom<sup>1</sup>, Idowu Kayode<sup>1</sup>

<sup>1</sup>Department of Mechatronics Engineering, Federal University of Agriculture, Abeokuta, Nigeria

Corresponding Author: [durodolafo@funaab.edu.ng](mailto:durodolafo@funaab.edu.ng)

Received 09 February 2026; revised 02 March 2026; accepted 15 April 2026

## Abstract

The pervasive challenge of maintaining clean living spaces is significantly addressed by the transformative role of autonomous vacuum cleaners. This paper details the design and development of an autonomous vacuum cleaner, leveraging Light Detection and Ranging (LiDAR) sensor technology and Robot Operating System 2 (ROS2) to implement efficient path planning and navigation algorithms. The methodology encompasses a comprehensive system architecture design, meticulous mechanical assembly, advanced software integration including Simultaneous Localization and Mapping (SLAM) and the Nav2 stack. Key results demonstrate robust motor control, accurate real-time data acquisition from the LiDAR sensor, reliable communication between microcontroller units, and effective dry waste collection across various indoor surfaces. The system's capabilities in autonomous navigation and cleaning represent a substantial contribution to the evolving landscape of smart home technologies, marking a strategic departure from conventional, less efficient navigation paradigms towards intelligent, adaptive, and resource-optimized autonomous solutions.

**Keywords:** Algorithms, LiDAR sensor, Path planning, ROS2, SLAM

## 1. Introduction

The advancement of robotics and automation technologies has significantly transformed not only industrial processes, but also revolutionized various aspects of our society and civilization (Blessing, 2024). One of these aspects is how we manage household chores with autonomous vacuum cleaners emerging as a popular solution, offering convenience and efficiency in maintaining clean living spaces. This paper explores the design, functionality, and implementation of an autonomous vacuum cleaner, highlighting its significance in the realm of smart home technologies. Traditionally, cleaning methods such as mopping, sweeping, and dusting have been time-consuming and could lead to discomfort or injuries, particularly for individuals with mobility issues (Arciniega-Rocha et al., 2023; Muhammad et al., 2025). The introduction of vacuum cleaners revolutionized the cleaning industry, making the process of cleaning faster and less demanding (Harish et al., 2020). Despite their effectiveness, the use of the vacuum cleaners still requires human control thus consuming time and energy and cannot be easily utilized by the elderly (Manasa et al., 2021; Vijayalaxmi et al., 2021).

Autonomous vacuum cleaners utilize a combination of sensors, algorithms, and actuators to navigate and clean spaces without human intervention. These devices utilize various technologies, such as Light Detection and Ranging (LiDAR), infrared sensors, ultrasonic sensors and sometimes computer vision, to map their environment, avoid obstacles, and optimize cleaning paths (Anıl & Hatice, 2022). The increasing demand for smart home devices, the growing awareness of the importance of cleanliness in promoting

health and well-being as well as the difficulties associated with making time for these cleanings have driven interest in developing more sophisticated and user-friendly cleaning solutions (Yusof et al., 2023).

The advancement of robotics and automation technologies has profoundly transformed various aspects of society, extending its influence from industrial processes to the management of household chores. Autonomous vacuum cleaners have emerged as a popular solution, offering convenience and efficiency in maintaining clean living spaces. A clean living-space is crucial for preventing the spread of germs and allergens, with the World Health Organization (WHO) claiming in 2019 that approximately 23% of all global deaths could be prevented through healthier environments. Studies have shown that maintaining cleanliness can lower the incidence of infections, contributing to overall health and reducing healthcare costs (Browne & Mitchell, 2023). Moreover, a clean environment not only promotes physical health but also enhances mental well-being, leading to a more productive and fulfilling life (Brass, 2019).

The evolution from manual tools to automated devices has further progressed with the emergence of autonomous systems. An autonomous system is capable of perceiving its environment, making independent judgments, and selecting actions to achieve goals, critically possessing the ability to learn and evolve. This capacity for independent decision-making and adaptation has necessitated the development of autonomous vacuum cleaning robots, addressing the limitations of traditional methods and the increasing demand for convenience in modern lifestyles. Early autonomous vacuum cleaning robots explored various navigation algorithms. Anıl and Hatice (2022) designed and implemented a vacuum cleaning robot based on the random-walk and snake or 'S' shaped pathway algorithm. Ong and Ku Azir (2020) also implemented the random walk alongside the wall-following algorithm. While these algorithms provided basic movement patterns, they often lacked advanced mapping capabilities, limiting their efficiency in complex environments and leading to redundant cleaning.

Sensor technologies for obstacle detection in the early models frequently relied on ultrasonic sensors (Anıl & Hatice, 2022; Bagaye et al., 2020; Harish et al., 2020; Manasa et al., 2021; Muhammad et al., 2025; Nabamita et al., 2019; Ong & Ku Azir, 2020). However, ultrasonic sensors have inherent limitations in accuracy, range, and directionality, often leading to basic reactive behaviors where the robot might simply stop or turn upon encountering an obstacle, rather than intelligently navigating around it. Infrared sensors have also been used but majorly serve as auxiliary sensors due to their dependence on lighting condition and narrow field of view. Krishnan et al. (2022) utilized a Raspberry Pi Camera and the OpenCV software for detecting obstacles in order to create a map of its environment. The control and processing units in these robots typically involved microcontrollers like NodeMCU (Sumit, 2025), Arduino (Anshu & Pawan, 2020; Ejiko et al., 2022; Manasa et al., 2021; Muhammad et al., 2025; Ong & Ku Azir, 2020) alone, such microcontrollers serving as slave device to a single-board computers such as Raspberry Pi (Rizuwana et al., 2020) or the Raspberry Pi alone (Ajay et al., 2020; Anıl & Hatice, 2022; Krishnan et al., 2022). Early attempts at indoor mapping and localization included GPS modules, as seen in Shakhawat et al. (2019), but these proved challenging due to signal attenuation by solid objects.

A significant progression in the field has been the shift towards LiDAR-based Simultaneous Localization and Mapping (SLAM). This technology enables real-time environment mapping, optimal path planning, and dynamic obstacle avoidance (Ajay et al., 2020; Anıl & Hatice, 2022). LiDAR offers substantial advantages over basic sensors by providing complete 3-D information of obstacles and a 360-degree real-time view of the environment, which is crucial for intelligent navigation (Ajay et al., 2020; Rizuwana et al., 2020). This superior sensing capability is fundamental to implementing complex SLAM algorithms, allowing for real-time mapping, optimal path planning, and dynamic obstacle avoidance. This progression moves the robot from simply reacting to obstacles to intelligently navigating around them while simultaneously building and continuously updating its internal map of the environment (Rizuwana et al., 2020). This represents a fundamental transition in robot behavior from simple reactive responses to proactive, environmentally aware decision-making.

Cleaning mechanisms in autonomous vacuum cleaners typically involve brush mechanisms for agitating dirt and powerful suction fans (Anil & Hatice, 2022; Ejiko et al., 2022). A prominent trend is the incorporation of integrated mopping functions, transforming these devices into versatile 2-in-1 cleaning solutions (Anil & Hatice, 2022). User interaction and autonomy features have also evolved. Some studies incorporated Internet of Things (IoT) technology to remotely control the vacuum cleaner's operation often through the use of a mobile application (Joon & Kowalczyk, 2021; Nabamita et al., 2019; Sumit, 2025; Vijayalaxmi et al., 2021). Furthermore, features that enhance autonomy, such as self-emptying dustbins that automatically transfer collected debris to a larger receptacle in the docking station, and automatic recharging capabilities, are becoming increasingly prevalent. More advanced concepts include the integration of robotic arms for cleaning uneven areas or object manipulation, as seen in Joon and Kowalczyk (2021). Performance evaluation frameworks, as proposed by Shivansh and Shraddha (2023), provide metrics for assessing cleaning efficiency, operational efficiency, productivity, durability, safety, and adaptability.

This research aims to develop a robot capable of navigating diverse home environments, detecting obstacles, and adapting to various floor types while optimizing cleaning paths. By employing sophisticated algorithms for mapping and localization, the robot will ensure thorough coverage of cleaning areas, thereby reducing the time and effort required for maintaining cleanliness in homes.

## 2. Methodology

### 2.1 Hardware Components

- (a) Raspberry Pi 4B was chosen as the single-board computer (SBC) responsible for processing sensor data and executing complex algorithms because of its sufficient processing power for real-time navigation tasks and its robust support for the Robot Operating System 2 (ROS2) on Ubuntu 22.04, which is essential for implementing sophisticated mapping and navigation algorithms.
- (b) Arduino Nano was used as the slave microcontroller receiving signals from the Raspberry Pi and providing low-level control to the motors.
- (c) RPLiDAR A1 M8 was chosen as the primary sensor as its ability to scan in a 360-degree radius and produce real-time detailed distance measurements which can be used to create accurate maps makes it an effective choice for navigation and obstacle avoidance.
- (d) BNO055 Inertial Measuring Unit (IMU) sensor is also used to provide odometry information.
- (e) Two Encoder motors are used for locomotion, driving the robot's wheels and providing sufficient torque to navigate various indoor surfaces.
- (f) A DC motor extracted from a Twin Torque geared motor is used in the vacuum's suction system.
- (g) L298N Motor driver is used to provide power and control signals to the encoder-motors. A single L298N motor driver IC can control two motors; therefore, the speed and direction of each motor can be controlled independently.
- (h) A 5V lithium-ion battery pack with a capacity of 10,000 mAh was chosen to provide a balance between functionality and weight, supplying power to the Raspberry Pi which in turn powers both the RPLidar and the Arduino Nano. A separate combination of Lithium-ion battery cells in 3S2P configuration is used to power all motors.

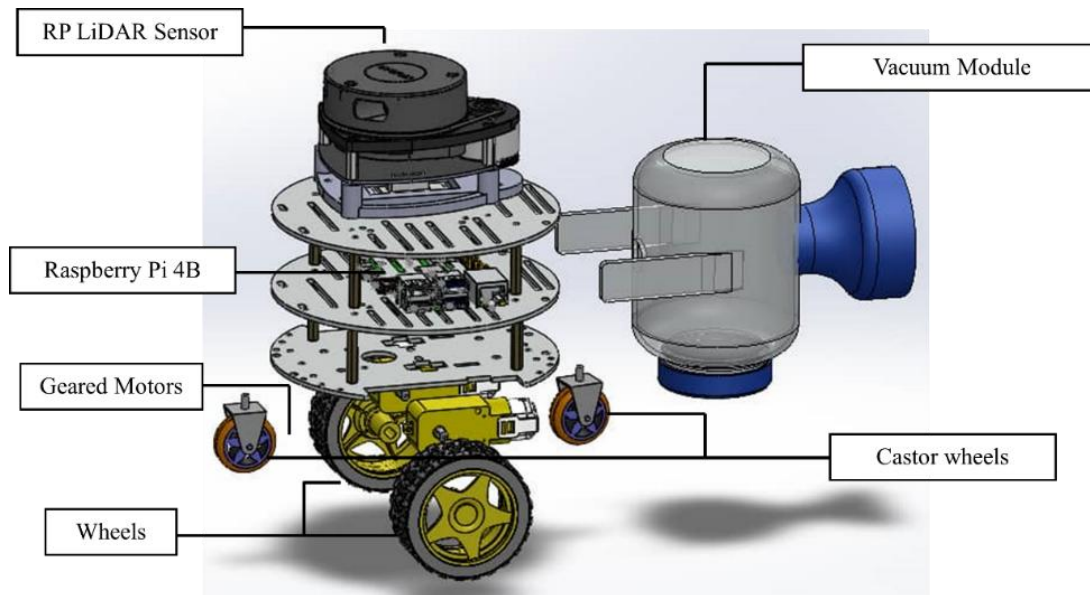
The specifications of the key components are summarized in Table 1.

**Table 1: Components Specifications**

| S/N | Components     | Specifications                                                                                                                                                            |
|-----|----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1.  | Raspberry Pi 4 | RAM: 4 GB, CPU: ARM Cortex-A72 quad-core, Support: ROS2 on Ubuntu OS                                                                                                      |
| 2.  | RP LiDAR       | Detection range: 12 meters, Sample rate: 8000 sample/secs, Angular resolution: 1°, Distance resolution: 0.2% of actual distance, Power supply: 5V, Scan frequency: 5.5 Hz |
| 3.  | Battery pack   | Voltage: 12 V, Capacity: 10000 mAh                                                                                                                                        |
| 4.  | Motor Driver   | Supply voltage: 12 V                                                                                                                                                      |
| 5.  | Arduino Nano   | Operating Voltage: 5V                                                                                                                                                     |
| 6.  | Encoder Motors | Supply Voltage: 9V to 12V                                                                                                                                                 |
| 7.  | DC Motor       | Supply Voltage: 5V to 12V                                                                                                                                                 |

## 2.2 Mechanical Assembly

The mechanical structure of the robot was designed using SolidWorks to ensure stability, ease of movement, and proper positioning of all sensors and components. The design and the exploded view image of the autonomous vacuum cleaner are shown in Figure 1 and Figure 2.

**Fig. 1: 3D Design of the Vacuum Cleaner**

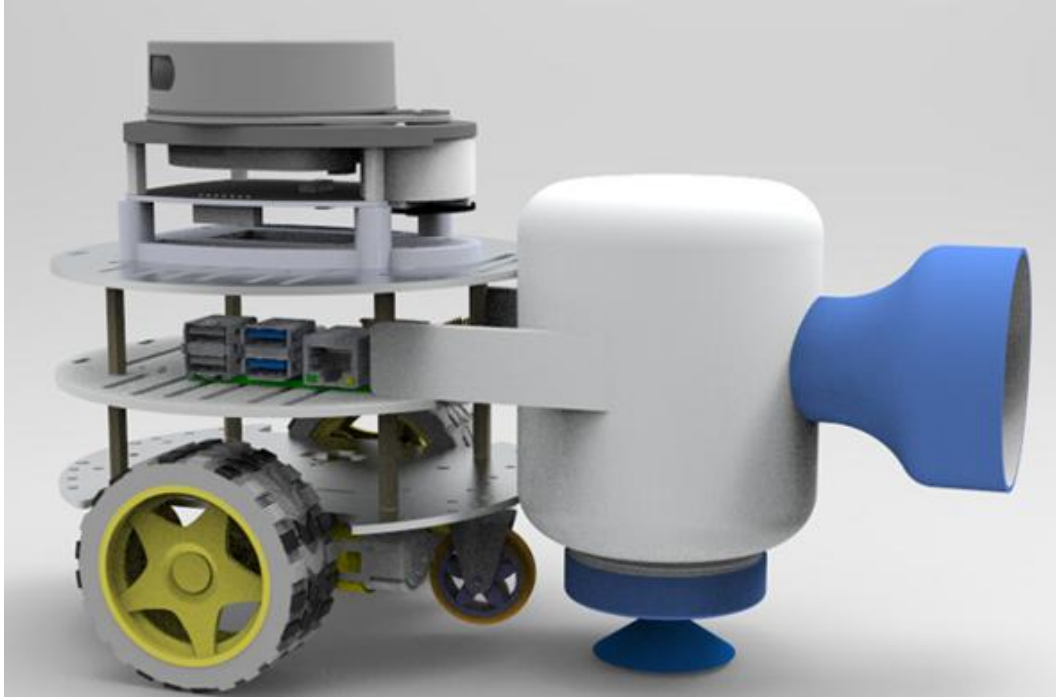


**Fig. 2:** Exploded View of the Vacuum Cleaner

The autonomous vacuum cleaning robot consists of two sub-assemblies which are as follows:

### **2.2.1 Chassis Assembly**

The robot's body features a light-weight, low-profile chassis commercially purchased. This design allows the robot to move effectively under furniture and in tight spaces. The chassis has three layers with the RPLidar placed on the top layer for efficient operation, the middle layer houses and protects the Raspberry Pi, Arduino Nano and the 5V Li-ion battery system. The motor driver is placed on the bottom layer due to proximity to the encoder motors.

### **2.2.2 Vacuum Mechanism Assembly**

The vacuum system was made from locally sourced materials such as a recycled plastic container which serves as the bin, a funnel which works as an inlet and aluminium sheet from which the impeller was made. The impeller and a DC motor supplied with 12V creates the necessary suction, collecting the dirt into a removable bin through the inlet that can be easily emptied and cleaned.

The sub-assemblies were then combined through the use of a zip-tie. The block diagram of the system, illustrated in Figure 3, depicts the comprehensive integration of all system components and their interactions. The circuit diagram is also provided in Figure 4.

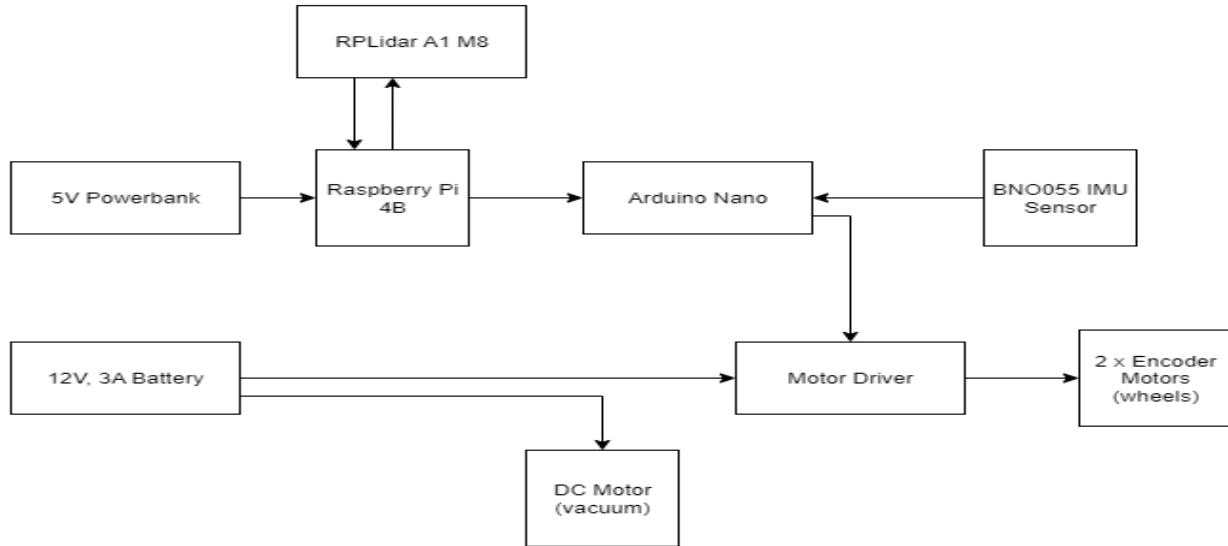


**Fig. 3:** Block Diagram of the Vacuum Cleaner

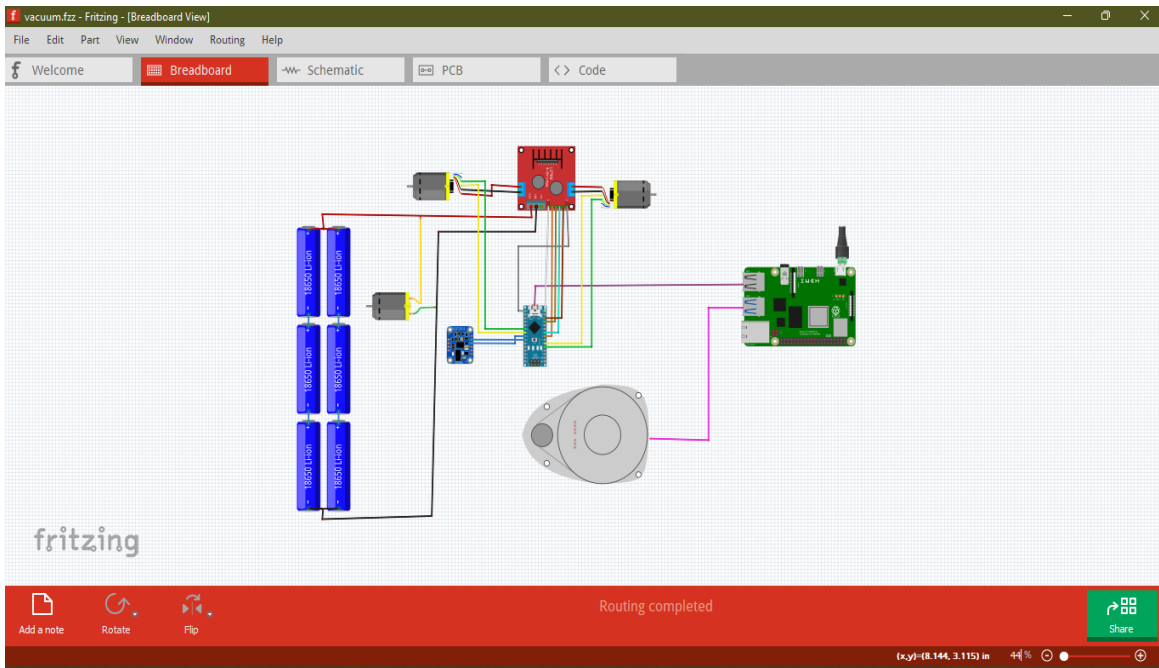


**Fig. 4:** Circuit Diagram of the Vacuum Cleaner

## 2.3 Software Development

The ROS2 environment was installed on the Raspberry Pi 4, utilizing Ubuntu 22.04 as the operating system. ROS2 packages were configured to facilitate seamless data flow between the LiDAR sensor, the central processing unit, and the control modules, enabling real-time navigation. Specific ROS2 nodes were developed to handle distinct tasks, such as data collection from the LiDAR sensor, SLAM processing, and motor control. Corresponding ROS2 topics were established for publishing and subscribing to various data

streams, including LiDAR scans and motor control signals, ensuring robust inter-component communication. Navigation and Simultaneous Localization and Mapping (SLAM) were implemented using specialized ROS2 packages. The gmapping algorithm was configured to construct a real-time map of the environment based on LiDAR scans while concurrently tracking the robot's precise location within that map. The flowchart shown in Figure 5 illustrates the navigation and cleaning algorithm using LiDAR-based environmental mapping.

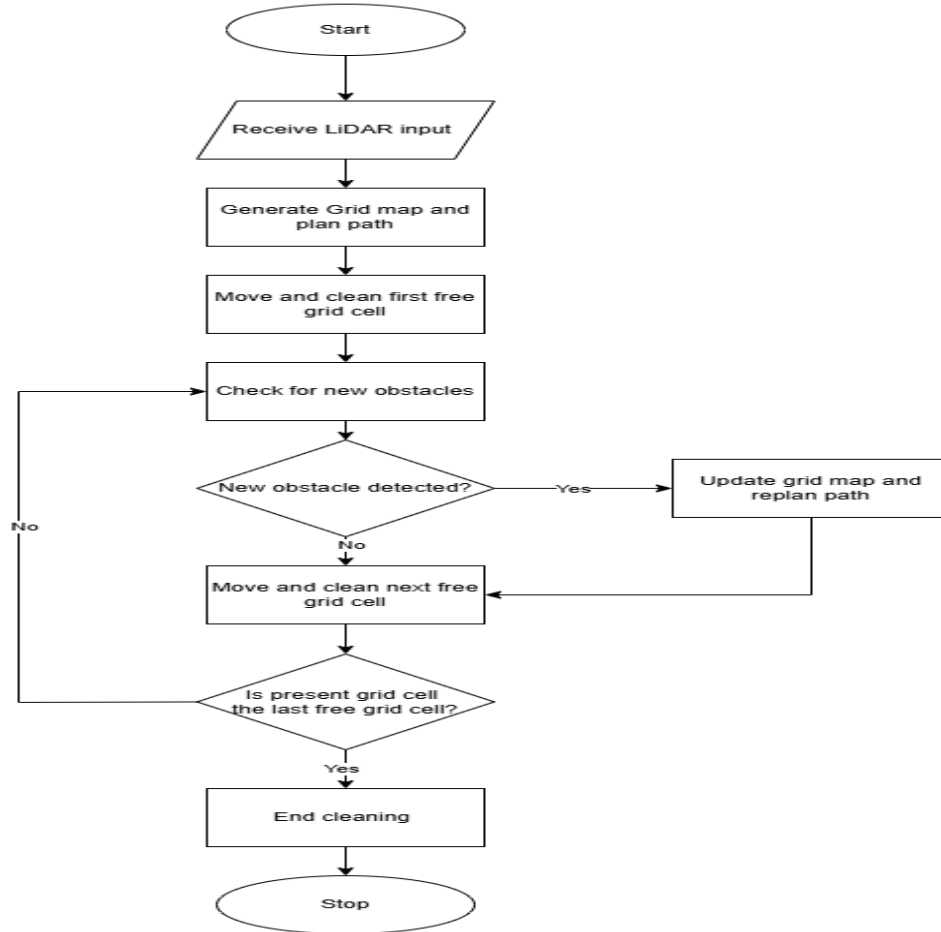


Fig. 5: Flow chart of the System

## 2.4 Design Calculations

The electrical ratings of the components are based on their voltage and current ratings which are provided by their manufacturer as well as their power consumption.

The power  $P$  utilized by a component with current  $I$  and voltage  $V$  is given by equation 1.

$$P = I \times V \quad (1)$$

The electrical components are split into two units:

### 2.4.1 Control And Sensory Unit

The control and sensory unit are made up of the Raspberry Pi which serves as the central processing unit, the Arduino Nano used to control the L298N motor driver and the RP LiDAR A1 M8 which receives environmental data and sends it to the Raspberry Pi for processing.

The Raspberry Pi 4B is rated 5v, 3A. It sets asides 1.2A for its peripherals using up to 1.8A to run itself. The Raspberry Pi uses 0.6A when operating at idle state and about 1.28A at overclocked state. The Arduino nano and RP LiDAR are connected to the Raspberry Pi via 5v, 0.5A USB ports each.

From equation (1), the Power consumed by each of these components are given below:

Power consumed by Raspberry Pi in Idle state ( $P_{ri}$ ):  $5v \times 0.6A = 3W$

Raspberry Pi in overclocked state ( $P_{ro}$ ):  $5v \times 1.28A = 6.4W$

Therefore, the Raspberry Pi consumes 3W to 6.4W

Power consumed by Arduino Nano ( $P_n$ ):  $5v \times 0.5A = 2.5W$

Power consumed RP LiDAR ( $P_l$ ):  $5v \times 0.5A = 2.5W$

Assuming the Raspberry Pi is overclocked, the maximum total power utilized by the control and sensory unit is given by equation 2.

$$Total\ Power = P_{ro} + P_n + P_l \quad (2)$$

Therefore,

$$Total\ Power = 6.4 + 2.5 + 2.5$$

$$Total\ Power = 11.4W$$

Since the Raspberry Pi receives a 5v input voltage, the current drawn by the Raspberry Pi and thus the control and sensory unit from the battery can be obtained from the expression in equation 3.

$$Current\ (A) = \frac{Total\ Power\ (W)}{Voltage\ Rating\ (V)} \quad (3)$$

Hence,

$$Current\ (A) = \frac{11.4W}{5V}$$

$$Current = 2.28A$$

The Raspberry Pi draws 2.28A of current from the battery or power supply, therefore the supply must be able to provide at least 2.3A. It is for this reason and also because it is a delicate component requiring a constant supply of 5V, that the raspberry Pi is connected to a dedicated Power source capable of providing an output of 3A and a constant voltage of 5V.

## 2.4.2 Motor Unit

The motor unit comprises of a L298N motor driver, two encoder-motors and a DC motor (extracted from a geared motor) for the vacuum unit, and the power source. The power supply is capable of supplying 12V and at least 2.7A for effective operation.

Each of the encoder motors is to be operated with a voltage 12V and a stall current of 1A (1000mA). Therefore, from equation (1), the power consumed by each motor ( $P_m$ ) is:

$$P_m = 12 \times 1$$

$$P_m = 12W$$

The Total Power consumed by both motors is:

$$Total\ Power = 2 \times P_m$$

$$Total\ Power = 2 \times 12W$$

$$Total\ Power = 24W$$

The vacuum motor is operated close to no-load and thus can be used with about 0.7A (700mA). The power consumed by this motor ( $P_v$ ) is:

$$P_v = 12 \times 0.7$$

$$P_v = 8.4W$$

The total power consumed by all three motors is thus:

$$Total\ Power = 24W + 8.4W$$

$$Total\ Power = 32.4W$$

The power source consists of six 18650 Li-ion cells, connected in 3 Series, 2 Parallel (3S2P) configuration. Each cell has a rating of 3.7V to 4.2V, 1.5A and a capacity of 2.5Ah.

Using the 3S2P configuration, the total voltage is 3(3.7) to 3(4.2), current is 2(1.5) A and capacity is 2(2.5) Ah which gives a voltage range of 11.1V to 12.6V, current of 3A and a total capacity of 5Ah.

Therefore, when the battery is fully charged, it is capable of supplying the 12V and 3A required for the motor unit.

## 2.5 Battery Runtime

The battery runtime is the time the battery takes to completely discharge after being fully charged while operating continuously at maximum load.

This is mathematically expressed in equation 4.

$$\text{Battery runtime} = \frac{\text{Battery supply voltage (V)} \times \text{Battery capacity (Ah)}}{\text{Total power consumption (W)}} \quad (4)$$

The power source for the control unit has ratings of 5v, 3A, 10,000mAh. And the total power consumed is calculated as 11.4W

Thus, from equation (4),

$$\text{Battery runtime for the control unit} = \frac{5 \times 10}{11.4}$$

Battery runtime for control unit = 4.39 hours or (4 hours 23 minutes 6 seconds)

The power source for the motor unit has ratings of 12v, 3A, 5,000mAh. And the total power consumed is calculated as 32.4W

Thus, from equation (4),

$$\text{Battery runtime for the motor unit} = \frac{12 \times 5}{32.4} = 1.85$$

Battery runtime for motor unit = 1.85 hours or (1 hour 51 minutes)

Therefore, the effective Battery runtime for the vacuum cleaning robot is estimated as 1.85 hours (1hour, 51minutes).

## 2.6 LiDAR Coverage and Field of View

The LiDAR works by rotating and taking distance measurements at specific angular intervals. The number of distance points captured by the sensor per full rotations depends on the angular resolution.

Total points per full 360° scan is given by equation 5.

$$N = \frac{360^\circ}{\text{Angular resolution}} \quad (5)$$

Since the Angular Resolution for the LiDAR sensor as specified by the manufacturer is 1°, Total points per full 360° scan is given by:

$$N = \frac{360^\circ}{1}$$

$N = 360 \text{ points per full revolution.}$

### 2.6.1 Distance Measurement Accuracy

The distance resolution of the LiDAR specified by the manufacturer is 0.2% of the actual measured distance.

Therefore, for any detected object at distance, d, the error in measurement is given by:

$$\text{Error} = 0.002 \times d$$

### 2.6.2 Data Processing

Having shown that each scan consists of 360 distance points, and happens at 5.5Hz (scans per second) according to the data sheet, therefore the total number of distance points processed per second is given by:

$$\text{Data Rate} = 360 \times 5.5$$

$$\text{Data Rate} = 1980 \text{ distance points per second}$$

### 2.6.3 Scan Time per Revolution

The LiDAR sensor has average scan rate (frequency) of 5.5Hz, meaning it completes 5.5 full scans per second.

The time taken for one complete scan is given by equation 6

$$T_{scan} = \frac{1}{\text{Scan Rate (Hz)}} \quad (6)$$

$$\text{Therefore, } T_{scan} = \frac{1}{5.5(\text{Hz})} = 0.182 \text{ seconds.}$$

### 2.6.4 Robot Movement Synchronization with LiDAR

The maximum speed at which the robot can move without missing obstacles depends on the LiDAR scan rate. If an obstacle is detected, the robot should be able to react before the next scan cycle completes.

To avoid missing obstacles, the robot must not move further than the LiDAR's resolution per scan cycle. If the robot moves as velocity,  $v$ , the maximum distance travelled per scan cycle is given by equation 7.

$$d_{max} = v \times T_{scan} \quad (7)$$

For safe operation, this distance should be small enough so that the LiDAR detects obstacles before the robot collides with them.

A good assumption is that the robot should not travel more than a fraction (e.g. 10%) of the detection range per scan cycle to allow sufficient time for reaction.

Assuming a safe travel distance per scan of 10% of the detection range:

$$d_{safe} = 0.1 \times 12$$

$$d_{safe} = 1.2\text{m}$$

using this as  $d_{max}$ ,

$$V_{max} \times 0.182 = 1.2$$

The maximum safe speed is thus:

$$V_{max} = \frac{1.2}{0.182}$$
$$V_{max} = 6.59\text{m/s}$$

## 3. Results and Discussion

The implemented autonomous vacuum cleaning robot is shown in Figure 6. A number of unit tests were carried out on the autonomous vacuum cleaning robot. This was to ensure that all individual components or sub-units of the robot work correctly before an integration test of the entire system was carried out. The results are given below:

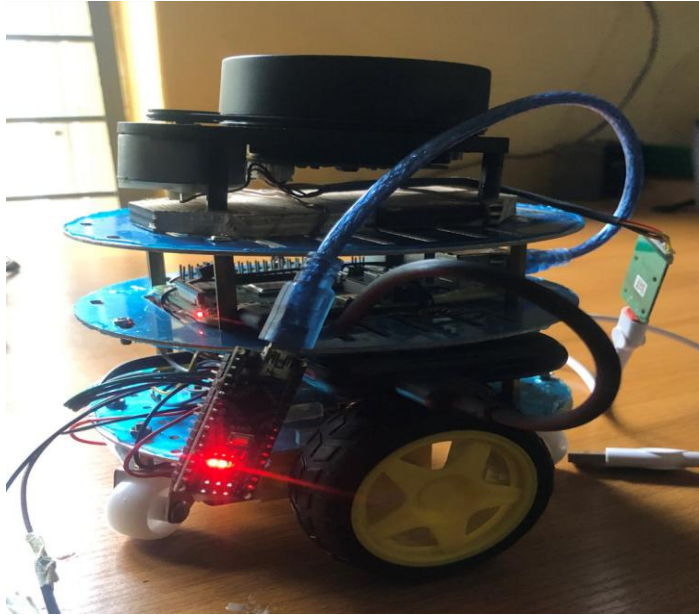


**Fig. 6:** Implementation of the Autonomous Vacuum Cleaner Robot

### 3.1 Motor Unit Testing

The objective of this test was to assess the individual and collective responsiveness of the geared DC motors and the L298N motor driver in controlling the robot's movement through Pulse Width Modulation (PWM) signals issued from the Arduino. Each wheel was tested separately to evaluate its response to varying duty cycles. When powered by a 12V DC supply, both left and right motors responded accurately to the PWM signals, demonstrating smooth acceleration and deceleration. Directional control was also verified by reversing motor polarity via digital signals from the Arduino, confirming precise control over both forward and reverse motion on each wheel independently.

Subsequently, both motors were driven simultaneously to assess coordinated locomotion. The robot executed forward, reverse, and rotational (pivot/turn) motions with minimal latency, indicating proper synchronization and sufficient torque delivery. The motors were also tested under varying loading speeds. It was observed that the wheels did not rotate when the motors were loaded at speeds less than 90 RPM. At 90 RPM, the wheels began to move slowly, with increasing loading speed further increasing the rotation speed up to 250 RPM. Any further increase in loading beyond 250 RPM resulted in no rotation of the wheels, indicating that the power supply was insufficient to drive the motors under such high loads. The results of the effect of loading speeds on wheel rotation are presented in Table 2. Figure 7 shows the variation of the loading speed with the wheel rotation.

**Table 2: Test results of motors at varying loading speeds**

| S/N | LOADING SPEED (RPM) | WHEEL MOVEMENT (mm/s)                                                      |
|-----|---------------------|----------------------------------------------------------------------------|
| 1.  | 0 - 89              | No Rotation                                                                |
| 2.  | 90 -250             | Rotating speed is proportional to the loading speed as shown in figure 4.2 |
| 3.  | above 250           | No Rotation                                                                |

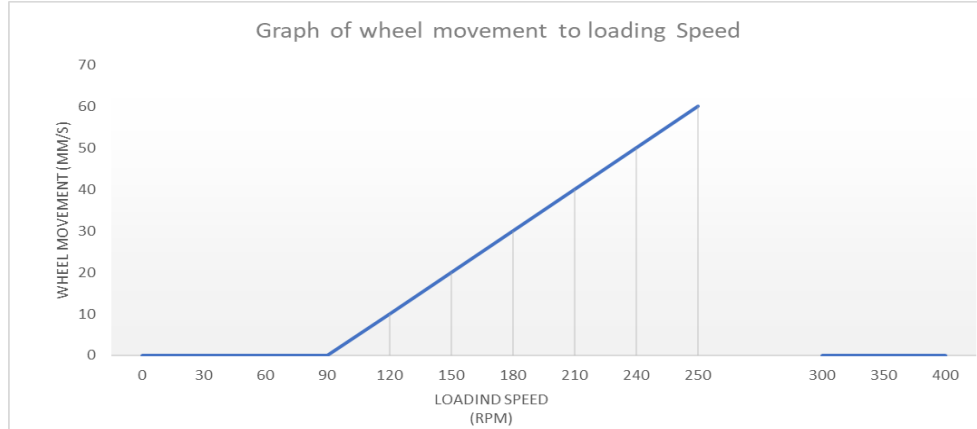


**Fig. 7:** Graph of Loading Speed Vs Wheel Rotation

### 3.2 RP LiDAR Unit Testing

The objective of this test was to evaluate the responsiveness and data acquisition capability of the RP LiDAR sensor when interfaced with the Raspberry Pi. The RP LiDAR A1 was connected to the Raspberry Pi via a USB interface, and the "rplidar" Python package was used to initiate scans and retrieve real-time distance data. Initial tests confirmed that the LiDAR motor spun correctly and emitted laser pulses upon initialization. The sensor was then allowed to perform continuous 360° scans, during which angular and distance data were collected and logged. Latency between scan commands and data receipt was minimal, and the sensor returned consistent readings across multiple trials.

Further testing focused on validating the accuracy and stability of the scan data under stationary and dynamic conditions. In a static setup, distance measurements to known obstacles were found to be within  $\pm 2$  cm of actual values, confirming sensor calibration and reliable distance resolution. The Raspberry Pi was capable of processing incoming data streams in real-time without performance degradation, demonstrating its suitability for handling LiDAR data at full scan rates. The RP LiDAR and Raspberry Pi together formed a stable sensing subsystem, capable of providing precise environmental perception for navigation tasks.

### 3.3 Arduino Unit Testing

The objective of this test was to validate the communication link and interoperability between the Raspberry Pi and Arduino Nano, ensuring reliable data exchange for coordinated robot control. The Raspberry Pi, functioning as the central processing unit, was responsible for higher-level decision-making, while the Arduino handled low-level tasks such as PWM signal generation and motor control. Serial communication (UART) was established via USB or GPIO Tx/Rx pins, using a standard baud rate of 9600 bps. Initial tests involved sending simple command strings (e.g., 'F' for forward, 'B' for backward) from the Raspberry Pi to the Arduino and observing the execution of corresponding motor actions.

The results demonstrated that the Arduino accurately interpreted and responded to commands sent from the Raspberry Pi with minimal latency. Bidirectional communication was also tested by having the Arduino send sensor or acknowledgment data back to the Pi, confirming the stability of the serial interface. Packet loss or communication errors were negligible during prolonged tests, and command-response timing was consistent. This confirmed that the communication pipeline between the two microcontrollers was robust and suitable for real-time control, enabling synchronized operation across the processing layers of the robot system.

### 3.4 Vacuum Module Testing

The objective of this test was to evaluate the effectiveness of the vacuum module in removing various types of household debris from different floor surfaces. The module was powered using a 12V supply delivered through a boost converter, ensuring adequate suction performance. Tests were conducted on both wooden and tiled surfaces to account for surface variation. Debris types included common particles such as dust, paper shreds, food crumbs, and small plastic fragments to simulate realistic cleaning scenarios.

The results, summarized in Table 3, indicate that the vacuum module effectively cleaned dust, dirt particles, human and pet hair, bread crumbs, fabric fibers, and paper scraps from both wooden and tiled floors. However, the module was not effective in cleaning spilled liquids or glass shards.

**Table 3:** Vacuum module test results

| S/N | DEBRIS TYPE        | WOODEN FLOOR | TILED FLOOR |
|-----|--------------------|--------------|-------------|
| 1.  | Dusts              | Cleaned      | Cleaned     |
| 2.  | Dirt particles     | Cleaned      | Cleaned     |
| 3.  | Human and Pet Hair | Cleaned      | Cleaned     |
| 4.  | Bread Crumbs       | Cleaned      | Cleaned     |
| 5.  | Spilled liquids    | Not cleaned  | Not cleaned |
| 6.  | Fabric fibers      | Cleaned      | Cleaned     |
| 7.  | Paper Scraps       | Cleaned      | Cleaned     |
| 8.  | Glass shards       | Not cleaned  | Not cleaned |

### 3.5 Analysis of Overall System Performance

The integrated system demonstrated robust performance across its core functionalities. The motor unit exhibited precise and responsive movement control, essential for effective navigation. The LiDAR sensor consistently provided accurate, real-time environmental data, which the Raspberry Pi processed efficiently without performance degradation. This strong performance in LiDAR-based mapping and obstacle avoidance directly addresses the limitations of older models that lacked advanced mapping capabilities and relied on less precise sensors. The reliable communication between the Raspberry Pi and Arduino Nano ensured synchronized operation of high-level decision-making and low-level motor control.

However, a critical functional limitation was revealed during the vacuum module testing: the inability to handle wet waste or glass shards. This observation highlights that despite significant advancements in robotic navigation and intelligence, the practical utility and widespread adoption of autonomous cleaning robots remain constrained by the need for parallel innovation in their core cleaning mechanisms. Even with highly sophisticated navigation, the robot cannot autonomously address a full spectrum of typical household cleaning scenarios involving spills or broken items. This means that the advanced navigation, while impressive, cannot compensate for this fundamental cleaning deficiency, reducing its overall utility for the end-user and necessitating manual intervention for specific types of messes. This underscores that for autonomous cleaning robots to achieve true "smart home" integration, their cleaning capabilities must evolve to be as robust and versatile as their navigation systems, addressing a wider range of debris types

and states.

#### 4. Conclusion

The design and development of the autonomous vacuum cleaning robot have been successfully implemented, demonstrating effective locomotion, advanced mapping, and intelligent navigation capabilities within indoor environments. Key findings from the comprehensive testing include the stable and responsive movement control provided by the selected 12V DC motors and L298N motor driver, the efficient handling of motor commands and sensor integration by the Arduino firmware, and the reliable transmission of real-time mapping data by the RP-LiDAR sensor without communication failures. Furthermore, the vacuum module proved effective in collecting various types of dry waste. These results collectively indicate that the robot successfully meets the functional requirements outlined in the research scope, representing a significant step towards a more interconnected and technologically advanced domestic environment.

Looking forward, improvements can be made to the autonomous vacuum cleaning robot to enhance its performance. A critical enhancement involves developing and integrating an automatic charging system to facilitate truly autonomous battery management, thereby minimizing human intervention. This should be complemented by a sophisticated battery monitoring system that enables adaptive charging scheduling based on the robot's usage patterns and remaining power. Another improvement that can be made is in terms of the vacuum mechanism to enable the robot clean up wet waste as well. A self-emptying dustbin mechanism would also significantly improve operational efficiency by reducing the frequency of manual maintenance.

#### References

- Ajay, G., Mayur, B., Pranay, D., Soumitra, A., & Medha, A. (2020). Autonomous floor cleaning robot with obstacle avoidance technology. *International Journal of Research in Engineering, Science and Management*, 3(6).
- Anil, E., & Hatice, D. (2022). Design and implementation of a cost-effective vacuum cleaner robot. *Turkish Journal of Engineering*, 166–177.
- Anshu, P., & Pawan, K. (2020). A smart autonomous floor cleaner with an Android-based controller. *IEEE*, 235–239.
- Arciniega-Rocha, R. P., Erazo-Chamorro, V. C., Phimmavong, P., & Gyula, S. (2023). Revolutionizing cleaning: The future of broomstick and dustpan design. *Journal of Civil Engineering and Environmental Sciences*, 9(2), 73–78. DOI: 10.17352/2455-488X.000072
- Bagaye, U. Y., Abubilal, K. A., & Usman, A. D. (2020). Development of an autonomous vacuum cleaner. *Zaria Journal of Electrical Engineering Technology*, 97–106.
- Blessing, E. (2024). The impact of robotics on society and civilization. *Robotics Engineering*.
- Brass, L. (2019). Good housekeeping: How the cleanliness (or messiness) of your home affects your mental, spiritual, and physical health. *Vibrant Life*, 35(2), 22–26.
- Browne, K., & Mitchell, B. G. (2023). Multimodal environmental cleaning strategies to prevent healthcare-associated infections. *Antimicrobial Resistance & Infection Control*, 12, Article 83. DOI: 10.1186/s13756-023-01274-4
- Ejiko, S. O., Erinle, T. J., Onibon, G. L., Ayeni, O. M., & Aliu, O. L. (2022). Design and construction of an autonomous motorized (robotic) vacuum cleaner for student's standard room. *Journal of Engineering Technology*.
- Harish, S., Prajith, K. P., Siddharth, B., & Prashanth, B. N. (2020). Design and analysis of automatic robotic vacuum cleaner. *International Journal of Advanced Research in Engineering and Technology*, 11(11), 728–736.
- Joon, A., & Kowalczyk, W. (2021). Design of autonomous mobile robot for cleaning in the environment with obstacles. *Applied Sciences*, 11, 8076.

- Krishnan, R. H., Naik, B. A., Patil, G. G., Pal, P., & Singh, S. K. (2022). AI-based autonomous room cleaning bot. In International Conference on Futuristic Technologies (INCOFT) (pp. 1–4).
- Manasa, M., Vidyashree, T. S., Bindushree, V., Sanjana, R., & Gowra, P. S. (2021). Smart vacuum cleaner. *Global Transitions Proceedings*, 2, 553–558.
- Muhammad, Z. H., Jamaludin, J., Muhammad, H. P., Sujana, M. R., & Mohamad, K. I. (2025). Smart robot cleaner using Internet of Things. *Journal of Advanced Research in Applied Sciences and Engineering Technology*, 46(1), 175–186.
- Nabamita, R. D., Rashmi, D., Sneha, A., & Kavita, M. (2019). Robotic automated floor cleaner. *International Research Journal of Engineering and Technology*, 6(3).
- Ong, R. J., & Ku Azir, K. N. F. (2020). Low-cost autonomous robot cleaner using mapping algorithm based on Internet of Things (IoT). *IOP Conference Series: Materials Science and Engineering*, 767, 012071.
- Rizuwana, P., Manuel, V. H., Madan, M. R., Raihan, E. A., & Mohan, R. E. (2020). Autonomous reconfigurable floor cleaning robot. *IEEE Access*, 8.
- Shakhawat, H. A., Md. Ragib, S., Md. Farhanul, I., Saiful, I., Noushad, H., & Shamik, D. (2019). Designing and optimization of an autonomous vacuum floor cleaning robot. *IEEE*, 25–30.
- Shivansh, K., & Shraddha, S. (2023). An empirical evaluation framework for autonomous vacuum cleaners in industrial and commercial settings: A multi-metric approach. *Research Berg*, 1–21.
- Sujana, K., Kumar, M. A., Karishma, M., Salman, S. K., & Raju, V. (2024). Vacuum cleaner using IoT. *Journal of Nonlinear Analysis and Optimization*, 15(1).
- Sumit, K. (2025). Development of an IoT-based smart vacuum cleaner controlled via NodeMCU for autonomous home cleaning solutions. *Journal of Cross-Disciplinary Sustainability Innovations*, 1(2), 1–8.
- Vijayalaxmi, S. K., Dnyaneshwari, J., Mansi, K., & Salim, L. (2021). Autonomous floor cleaning robot. *International Research Journal of Engineering and Technology*, 8(5).
- Yusof, A., Safiuddin, A., Ruslan, M., Khairiunizzam, A. M., & Fikry, A. (2023). Smart vacuum: A demand rise. DOI: 10.13140/RG.2.2.24879