DEVELOPMENT OF AN AUTONOMOUS SOLAR GRASS CUTTING ROBOT WITH PATH MEMORIZING MECHANISM

A PROJECT REPORT

Submitted by

VENUGOPAL. R	**(2002246)**
VIJAYAKRISHNA. R	**(2002248)**
VISHNUPRASAD. Y	**(2002252)**

In partial fulfilment for the award of the degree of

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

SRI RAMAKRISHNA ENGINEERING COLLEGE

(An Autonomous Institution)

COIMBATORE-641022

ANNA UNIVERSITY: CHENNAI-600025

MAY - 2024

SRI RAMAKRISHNA ENGINEERING COLLEGE

BONAFIDE CERTIFICATE

Department of Electronics and Communication Engineering

PROJECT WORK – MAY 2024

This is to certify that the project entitled

DEVELOPMENT OF AN AUTONOMOUS SOLAR GRASS CUTTING ROBOT WITH PATH MEMORIZING MECHANISM

is the bonafide record of project work done by

VENUGOPAL R

Register No: 2002246

VIJAYAKRISHNA R

Register No: 2002248

VISHNUPRASAD Y

Register No: 2002252

who carried out the project work under my supervision, certified further that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate of B.E. Electronics and Communication Engineering during the year 2020-2024.

Dr. M. Jagadeeswari, ME., Ph.D.	**Dr. C. S. Manikandababu, ME., Ph.D.**
Head of the Department	Guide

Submitted for the Project Viva-Voce examination held on _____________

Internal Examiner

External Examiner

ACKNOWLEDGEMENT

We put forth our heart and soul to thank **ALMIGHTY** for being with us all through this technical adventure.

We express our heartfelt thanks to our esteemed Managing Trustee, **Sri.D.LAKSHMINARAYANASWAMY, B.Tech, MBA,** and Joint Managing Trustee **Sri. R.SUNDAR, B.A, MBA,** for their invaluable support.

We thank our Principal, **Dr. N.R.ALAMELU**, **B.E. (Hons.), M.E., Ph.D.,** for her benevolent patronage in carrying out this project.

We express our sincere respect and gratitude to our Professor and head of the department, Electronics and Communication Engineering **Dr.M.JAGADEESWARI, M.E., Ph.D.,** for her continuous encouragement and motivation throughout the project.

We are obliged to our Project coordinator, **Dr.S.JAYANTHY, M.E., Ph.D., Professor** for giving us essential technical expertise and support.

We are indebted to our Project Guide, **Dr.C.S.MANIKANDABABU, M.E., Ph.D., Associate Professor** for providing encouragement and guiding us throughout this project with enthusiasm and enabling us to complete it within stipulated time.

We extend our thanks to all the teaching, non-teaching faculty members of our department and other departments, friends and family members for their support in carrying out the project successfully.

ABSTRACT

Grasses give beauty to the environment especially when properly managed. There are challenges of and not limited to time consuming, labor intensiveness in carrying out this maintenance thus, the need for the system which can autonomously perform such task. In an era marked by a growing emphasis on sustainability and automation, the integration of solar power and autonomous navigation technologies holds significant promise for eco-friendly outdoor tasks. The robot's design incorporates solar panels for renewable energy harvesting, addressing environmental concerns while reducing reliance on traditional power sources. Utilizing IR sensors and Arduino uno kit, the robot navigates predefined cutting paths autonomously, ensuring efficient grass cutting without human intervention. The path memorization mechanism allows the robot to learn and adapt its route over time, enhancing its effectiveness in diverse outdoor environments. Through systematic methodology encompassing component selection, assembly, and programming, the project culminates in a functional prototype capable of sustainable grass cutting operations. Results demonstrate the effectiveness of the autonomous grass-cutting robot, highlighting its efficiency in maintaining designated cutting paths while minimizing power consumption. Discussion delves into the implications of the project's findings, including its contributions to sustainable robotics and avenues for future improvement.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATION

EEPROM	Electrically Erasable Programmable Read-only Memory
PWM	Pulse Width Modulation
IDE	Integrated Development Environment
IR	Infrared Radiation
IoT	Internet of Things
RISC	Reduced Instruction Set Computer
SPI	Serial Peripheral Interface
UART	Universal Asynchronous Receiver/Transmitter
USB	Universal Serial Bus

LIST OF EQUATIONS

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Autonomous robots have become increasingly prevalent in various fields, offering efficient and precise solutions for tasks ranging from manufacturing to agriculture. In the realm of outdoor maintenance, the development of autonomous grass-cutting robots presents a promising avenue for sustainable landscaping practices. This project focuses on creating an autonomous solar grass-cutting robot equipped with an IR sensor-based path memorizing mechanism, controlled by an Arduino microcontroller. By harnessing solar energy for power and utilizing advanced sensing and control technologies, this robot aims to provide an environmentally friendly and efficient solution for maintaining lawns and green spaces with minimal human intervention.

1.2 PROJECT OVERVIEW

The project entails the design and construction of an autonomous solar grass-cutting robot equipped with IR sensors and an Arduino microcontroller for path memorization. The robot's main components include solar panels for energy harvesting, DC motors for propulsion, cutting blades for grass trimming, IR sensors for path detection, and an Arduino board for control and decision-making. The robot navigates autonomously within predefined boundaries, utilizing IR sensors to detect and memorize the cutting path. The Arduino board processes sensor data, enabling the robot to adjust its direction and cutting pattern accordingly. Through iterative testing and refinement, the project aims to optimize the robot's performance in terms of cutting efficiency, energy consumption, and navigational accuracy. The utilization of renewable solar energy enhances the robot's sustainability and autonomy, reducing reliance on external power sources. Overall, this project contributes to the advancement of autonomous outdoor maintenance systems while promoting environmental responsibility and technological innovation.

1.3 SUMMARY

This chapter deals with the basic introduction an autonomous solar grass-cutting robot with path memorization using IR sensors and Arduino. By harnessing renewable energy, the robot navigates autonomously and efficiently cuts grass within predefined boundaries. The integration of sustainable technology and robotic automation offers a promising solution for outdoor maintenance tasks while promoting environmental responsibility.

CHAPTER 2

LITERATURE SURVEY

2.1 INTRODUCTION

A literature survey is the section of a project report that highlights the numerous studies and research done in the issue of your interest, as well as the results that have already been published, while considering the project's varied aspects and scope. It is the most important part and guides the investigation in the appropriate direction. Inconsistencies and inequalities in study methodology are discovered after a careful examination of the available data. Only content that is relevant and directly related to the research is picked for the survey. A critical approach is required when selecting literature

2.2 BACKGROUND WORKS

Adeodu A.O. et al (2020) proposed a likely draws from existing literature on autonomous robots, obstacle avoidance techniques, path planning algorithms, and solar-powered systems. Commonly cited approaches may include sensor-based navigation methods, such as using ultrasonic or infrared sensors for obstacle detection, and path planning algorithms like A* or potential fields. Additionally, the paper might discuss prior research on solar-powered robots, emphasizing the importance of renewable energy sources for sustainable robotic applications. This literature survey serves as a foundation for understanding the context and motivation behind the development of the proposed autonomous grass-cutting robot and highlights the potential contributions of the current study to the field of robotics, particularly in semi-structured outdoor environments.

M. Manimegalai et al (2020) proposed a unique integration of functionalities into a solar-powered grass cutter. Commonly cited approaches may include techniques for autonomous navigation, such as GPS-based or sensor-based methods, as well as mechanisms for pesticide spraying and printing functionality. Additionally, this paper may discuss prior research on solar-powered agricultural robots, emphasizing the importance of renewable energy sources for sustainable farming practices. This paper has a foundation for understanding the context and motivation behind the development of the proposed multi-functional grass cutter and highlights the potential contributions of the current study to the fields of robotics and agriculture, particularly in the context of small-scale farming operations.

Dr. Khan Sohelrana. et al (2020) proposed a comprehensive overview of IoT-based gesture-controlled grass cutting vehicles with IoT applications in robotics, gesture recognition technologies, and autonomous vehicles. Commonly cited approaches may include IoT architectures for remote control and monitoring, machine learning algorithms for gesture recognition, and navigation strategies for autonomous vehicles. Additionally, this paper may discuss prior research on smart agriculture and precision farming, emphasizing the potential of IoT technologies to improve efficiency and productivity in agricultural operations. This paper serves as a foundation for understanding the state-of-the-art in gesture-controlled grass cutting vehicles and highlights the potential contributions of the current study to the field of agricultural robotics and IoT-based automation.

Dr. J. G. Chaudhari et al (2022) proposed an innovative solution for grass cutting machinery, utilizing solar power and advanced control systems. While specific citations are not provided, the paper is likely inspired by existing literature on solar-powered equipment and automation technologies. It draws from research on solar energy applications in agriculture and smart control systems for optimizing machinery performance. Additionally, it references studies on noise and pollution reduction in traditional lawn maintenance tools, highlighting the growing need for eco-friendly alternatives. By examining this related works, the paper establishes the context for its unique approach, emphasizing its potential contributions to sustainable agriculture and technological advancement.

Md. Rawshan Habib et al, (2019) proposed a novel grass cutting machine powered by solar energy and controlled using PID controllers. While specific citations are not provided, the paper likely builds upon existing research in solar-powered machinery and control systems. It may draw from literature on solar energy applications and PID control strategies for motorized systems. Additionally, it may emphasize the importance of lightweight and portable design in agricultural equipment and highlight the potential for autonomous operation to improve efficiency. Through this exploration of related works, the paper sets the stage for its unique contribution to sustainable agriculture and technological advancement.

Ms. Lanka Priyanka et al (2018) developed a solar-powered grass-cutting device with reinforced blades, detailed in their paper from 2015. This grass cutter can operate both manually and automatically. The materials commonly used in its construction include GI sheet, motor, wheel, aluminum sheet, transfer cord, rectangular pipe, and insulating material. Additionally, they incorporated components such as a comparator, rechargeable battery, relay, temperature sensor, and DC motor. The voltage produced by the solar panel is shown on Dipin's LCD display device. They introduced a fully automated, solar-powered garden mower capable of manual control with minimal effort. Through MATLAB programming and a digital camera mounted above the robot, software is fed into the system to guide its movements based on predefined patterns. These robots, designed to alleviate human effort, can detect humans and other objects in their path, thus protecting both themselves and humans from harm. The grass is cut in distinct patterns, offering a unique layout. The author's intention is to reduce human labor and dependence on non-renewable energy sources by creating a solar grass cutter machine. Solar energy is harnessed, stored in batteries, and utilized as required. With proper monitoring, these functions operate smoothly according to schedule. A special system prevents overcharging of the batteries, thereby extending their lifespan. This innovation has potential applications in small-scale gardening.

2.3 PROPOSED WORK

The proposed system entails developing an autonomous solar grass-cutting robot integrated with a path memorizing mechanism using Arduino and IR sensors. Solar panels harness renewable energy, powering the robot's movements and cutting operations. Equipped with IR sensors, the robot navigates autonomously, avoiding obstacles while efficiently trimming grass. The Arduino microcontroller orchestrates the robot's actions, processing sensor data and implementing path memorization algorithms. As it traverses the lawn, the robot memorizes key landmarks, ensuring consistent cutting paths for future sessions. This sustainable solution offers precise grass cutting without manual intervention, promoting eco-friendly lawn maintenance practices with minimal energy consumption and increased efficiency.

2.4 SUMMARY

This chapter deals with the survey made on various autonomous solar grass cutting machines. This survey provides critical reviews, highlights the concepts and analyses of different ways of cutting grasses without human intervention.

CHAPTER 3

HARDWARE DESCRIPTION

3.1 INTRODUCTION

The chapter deals a comprehensive overview of the hardware components utilized in the development of the autonomous solar grass-cutting robot. Through detailed descriptions and specifications, we elucidate the key elements comprising the robot's structure, power system, sensing capabilities, and cutting mechanism, facilitating a deeper understanding of its functionality and design.

3.2 BLOCK DIAGRAM

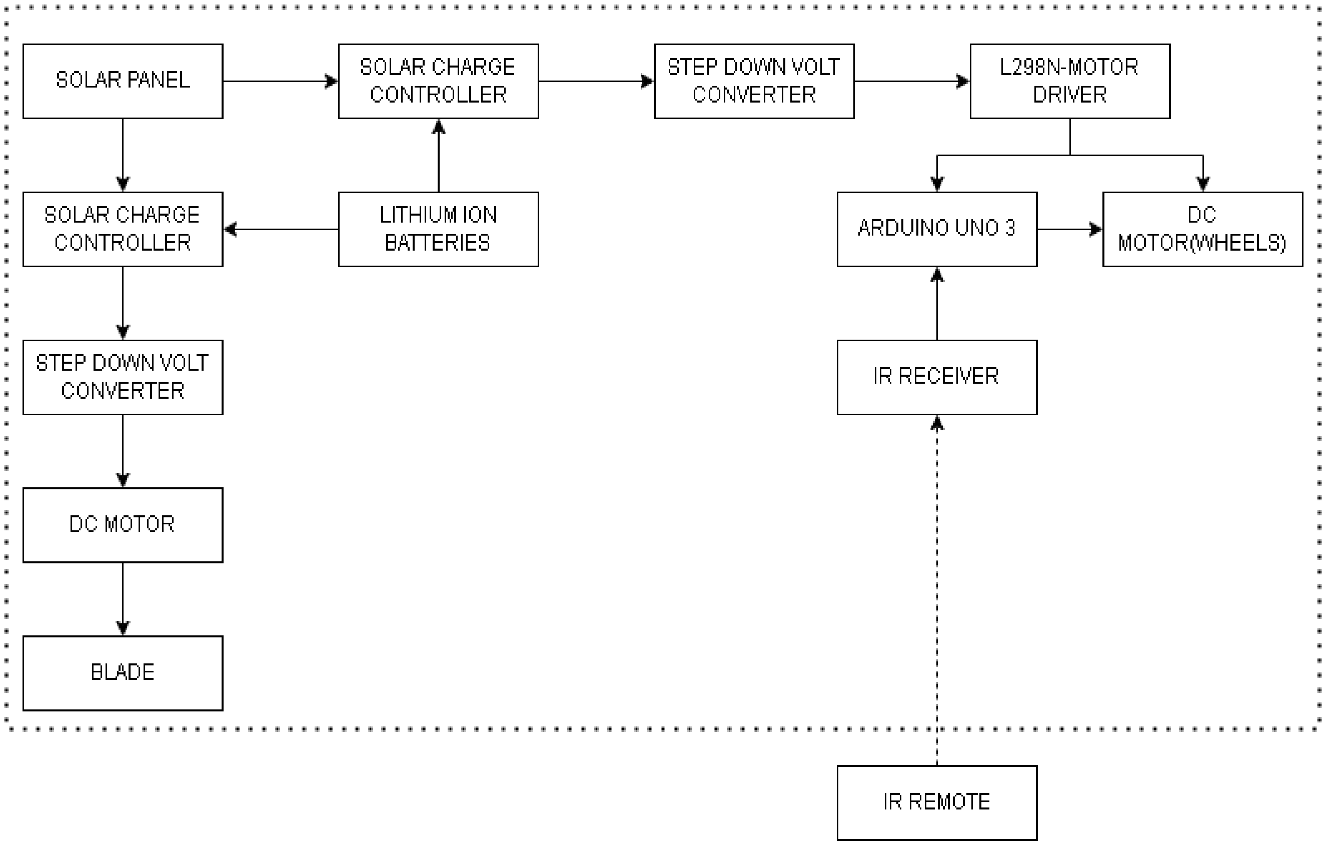


Figure 3.1 Block diagram for the proposed system of autonomous solar grass cutting robot

The Figure 3.1 represent the block diagram of the autonomous solar grass cutting robot depicts the functional modules and their interconnections within the system. At the core of the system is the Arduino Uno microcontroller, which serves as the central processing unit. The Arduino Uno interacts with various peripheral components, including the IR sensor, motor driver (L298N), and PWM solar charge controller, to facilitate autonomous operation. The IR sensor module detects signals from the remote control and relays them to the Arduino Uno for interpretation. Based on the received commands, the Arduino Uno activates the appropriate motor control signals to drive the DC motors connected through the L298N motor driver. This enables the robot to move forward, backward, turn left, or turn right as directed by the user. Additionally, the system incorporates power management components, such as the lithium-ion battery and DC-DC buck converter module, to ensure efficient utilization of solar energy. The solar panel harnesses sunlight to charge the battery, while the PWM solar charge controller regulates the charging process to prevent overcharging or deep discharging.

3.3 SCHEMATIC DIAGRAM

A schematic diagram is a visual representation of a system or process using symbols to represent its components and their interconnections. Schematic diagrams are typically used to depict circuits, systems, or structures, showing the relationships between different elements without necessarily depicting their physical layout or exact proportions.

3.3.1 SCHEMATIC DIAGRAM OF THE ROBOT

The diagram Figure 3.2 and Figure 3.3, depicts the intricate wiring connections and component layout of the autonomous solar grass-cutting robot. Starting with the grass cutting system, it consists of a solar panel, solar charge controller, lithium-ion battery and a DC motor which is connected to a stainless-steel blade. When the switch is on the DC motor starts to rotate the blade at a voltage of 3V with 3000 rpm to cut the grass. Then we have another solar panel which is connected to the solar charge controller, lithium-ion battery and a L298N motor driver. From the motor driver it is connected to the Arduino board. The left motor, pins 5, 6, and 7 are designated for the motor driver module connections. Pin 5 (ena) controls the motor's speed, while pins 6 and 7 (in1 and in2) dictate its direction. Similarly, for the right motor, pins 8, 9, and 10 are utilized, with pin 10 (enb) regulating motor speed. The IR receiver module, crucial for remote control functionality, is connected to pin 11 (RECV_PIN), allowing the Arduino to receive and decode infrared signals from the remote-control unit. Additional functionality includes a reset pin, connected to analog pin A0 (RESET_PIN), enabling system resets when necessary. The Arduino board serves as the central hub, orchestrating the interactions between the various hardware components. Through precise wiring and pin assignments, the diagram illustrates the seamless integration of motors, IR sensors, and remote-control functionality, facilitating a comprehensive understanding of the robot's hardware configuration and functionality. This detailed schematic serves as a valuable reference for both assembly and troubleshooting, ensuring the successful implementation of the autonomous grass cutting system.

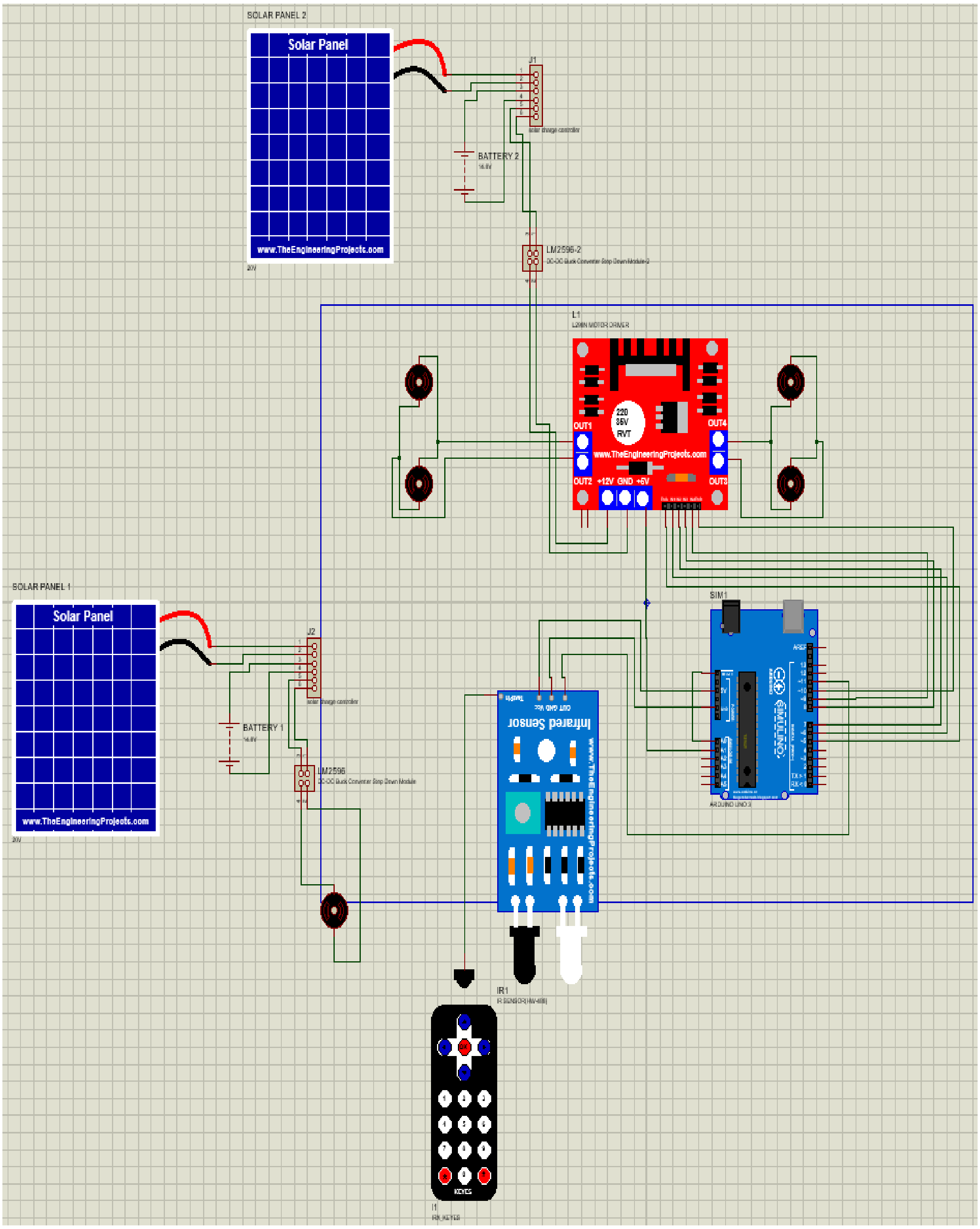


Figure 3.2 Schematic diagram for the autonomous solar grass cutting robot

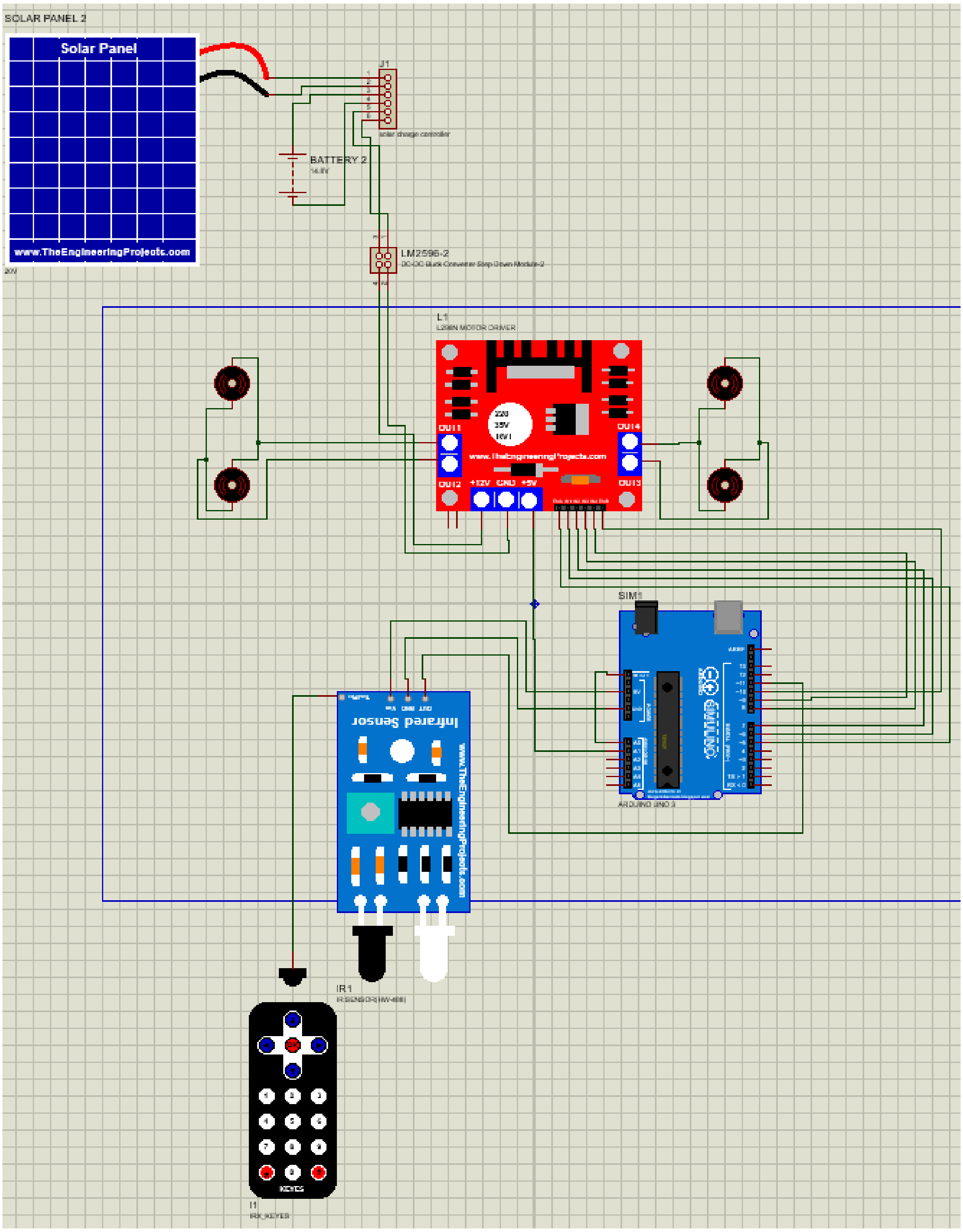


Figure 3.3 Schematic diagram for the main component of autonomous solar grass cutting robot

3.3.2 SCHEMATIC DIAGRAM OF THE GRASS CUTTING SYSTEM

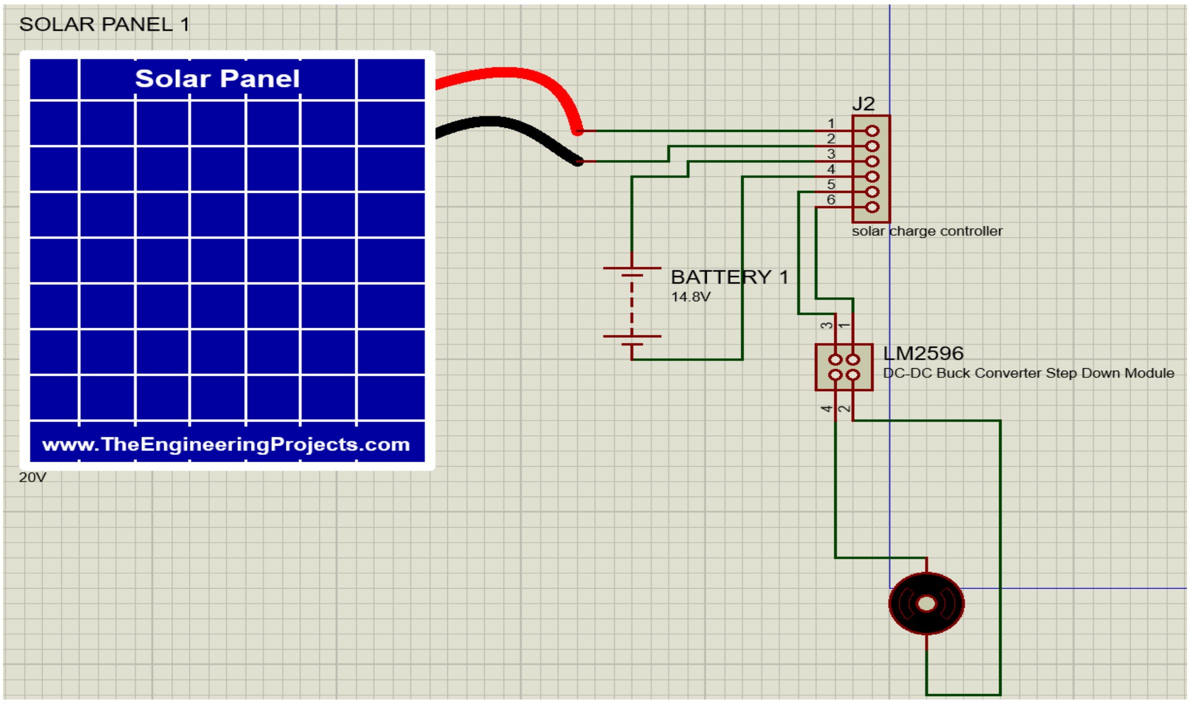


Figure 3.4 Schematic diagram of the grass cutting system of the robot

In the Figure 3.4 the solar panel collects solar energy and converts it into electricity. The solar charge controller regulates the voltage coming from the solar panel and protects the battery from overcharging. The step-down converter reduces the voltage from the battery to a level suitable for the DC motor driver. The lithium-ion batteries store the electricity generated by the solar panel. The DC motor drives the rotating blade that cuts the grass. This process is continued until the battery is drained, once it drained the solar charge controller cuts its connection from the output terminal and connects to the solar panel. From now the solar panel charges the battery with the help of charge controller. Once the battery is fully charged the process is repeated.

3.3.3 SCHEMATIC DIAGRAM OF THE IR SENSOR MODULE

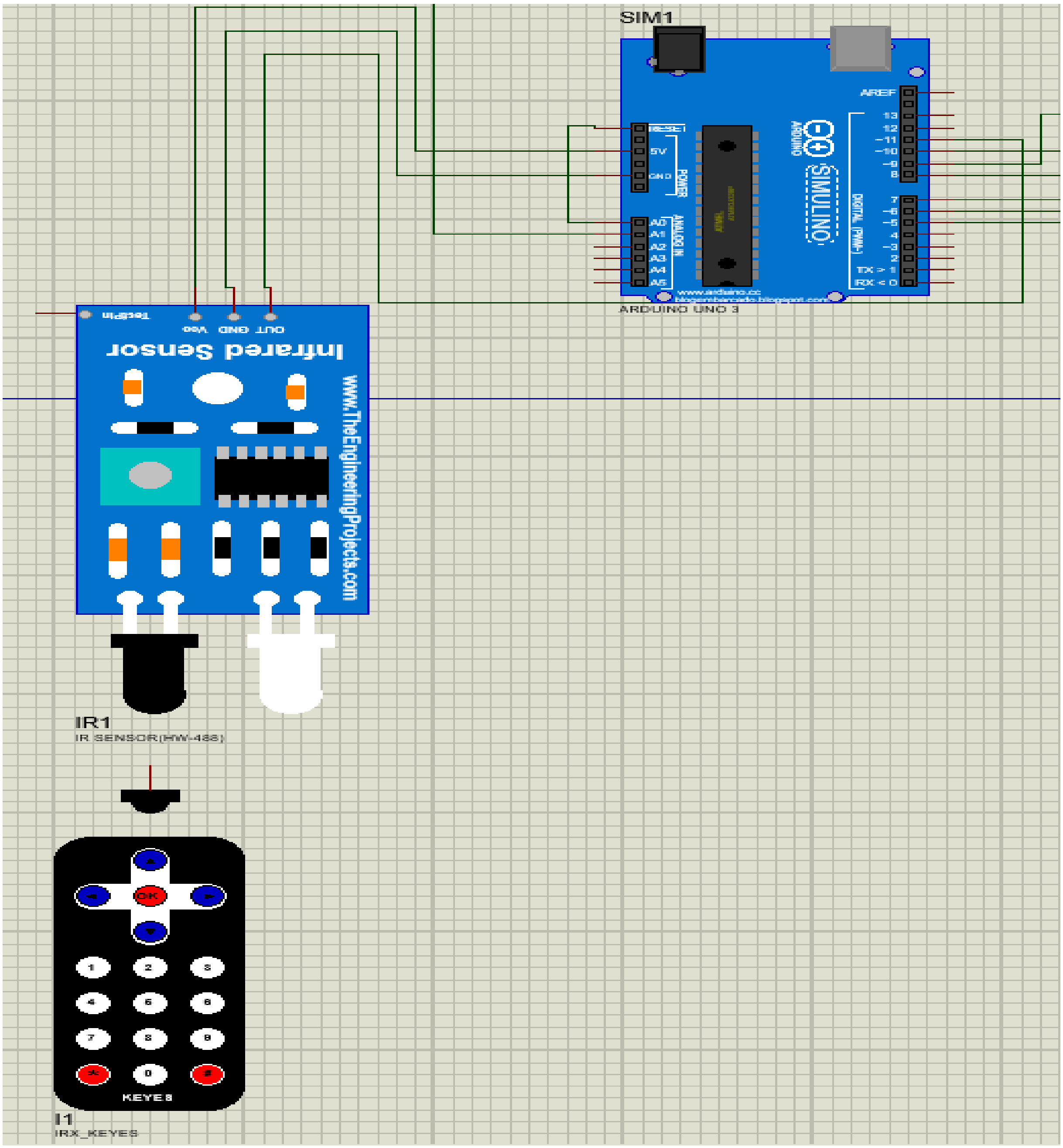


Figure 3.5 Schematic diagram of the connection of IR sensor to the Arduino board.

The Figure 3.5 represents the connection of the IR sensor to the Arduino microcontroller. Here we use a HW-488 IR module which detects an object distance ranging from 0-40 cm. This IR module has three pins which include power supply (Vcc), ground (Gnd) and output pin (Vout). The Vcc pin is connected to the 5V pin of the Arduino controller and the ground pin is connected to the ground of the Arduino controller. The Vout pin is connected to the 11th pin of the Arduino board. We take the hex value from the IR remote and pass the value to the Arduino code. For each value from the IR remote we assign the forward motion, Backward motion, left motion and right motions.

3.3.4 SCHEMATIC DIAGRAM OF THE MOTOR DRIVER MODULE

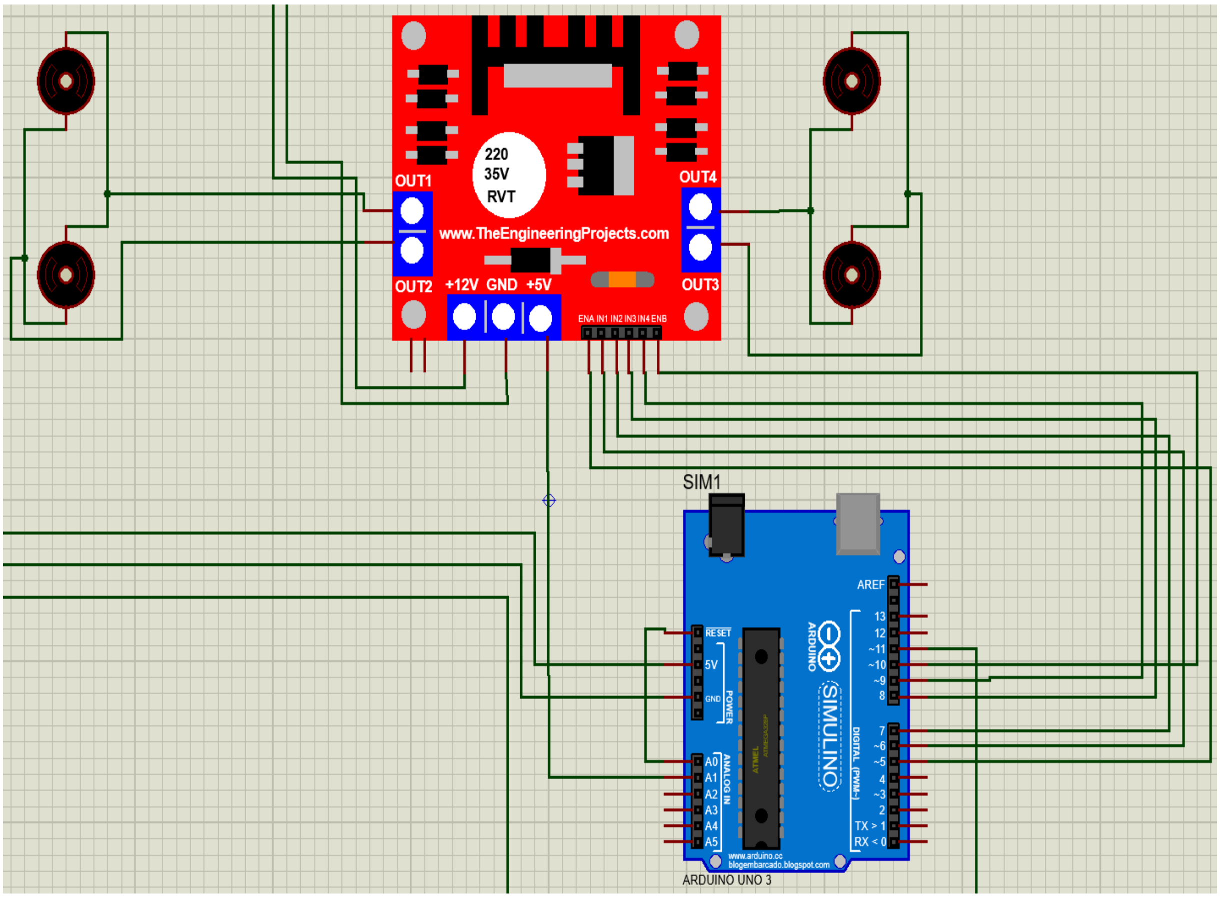


Figure 3.6 Schematic diagram of the connection of L298N motor driver to the Arduino board.

The figure 3.6 represents the connection between the L298N motor driver and Arduino uno board. There are thirteen pins in the motor driver module they are out1, out2, out3, out4, Vs(12V), Vss(5V), gnd, ena, enb, in1, in2, in3 and in4. The out1 and out2 consist of the two DC motor’s positive and negative pins. The out3 and out4 consist of another two DC motors positive and negative pins. The Vs pin (12V) is the power pin of the IC’s internal H-Bridge, which drives the motors. This pin accepts input voltages ranging from 5 to 12V. The logic circuitry of the L298N IC is powered by the Vss pin (5V), which has a 5V to 7V voltage range. You may adjust the motor's forward and backward rotation using the direction control pins. These pins on the L298N chip really regulate the switches of the H-Bridge circuit. Two direction control pins make up this structure. The direction that motor A spins is controlled by the IN1 and IN2 pins, while the direction that motor B spins is controlled by the IN3 and IN4 pins. Applying logic HIGH (5V) or logic LOW (Ground) to these inputs will regulate the motor's spinning direction. The results of several combinations are displayed in table 3.1.

Table 3.1 Direction of the robot for Various inputs

S.NO	INPUT	DIRECTION
1	In1= High In2= Low In3= Low In4= High	Forward
2	In1= High In2= Low In3= Low In4= Low	Left
3	In1= Low In2= Low In3= Low In4= High	Right
4	In1= Low In2= High In3= High In4= Low	Backward
5	In1= Low In2= Low In3= Low In4= Low	Stop

The motors are turned on and off, and their speed is controlled by the speed control pins ENA and ENB. The motors will spin if these pins are pulled HIGH; they will cease spinning if they are pulled LOW. Nevertheless, the motors' speed may be controlled with the use of pulse width modulation, or PWM. Usually, a jumper is attached to these pins on the module. The engine runs at maximum speed when this jumper is in the correct position. Take off the jumpers and connect the motors to the PWM-enabled pins on the Arduino if you wish to control the motors' speed programmatically.

3.4 SOLAR CHARGE CONTROLLER

Figure 3.7 Solar charge controller and its pin diagram

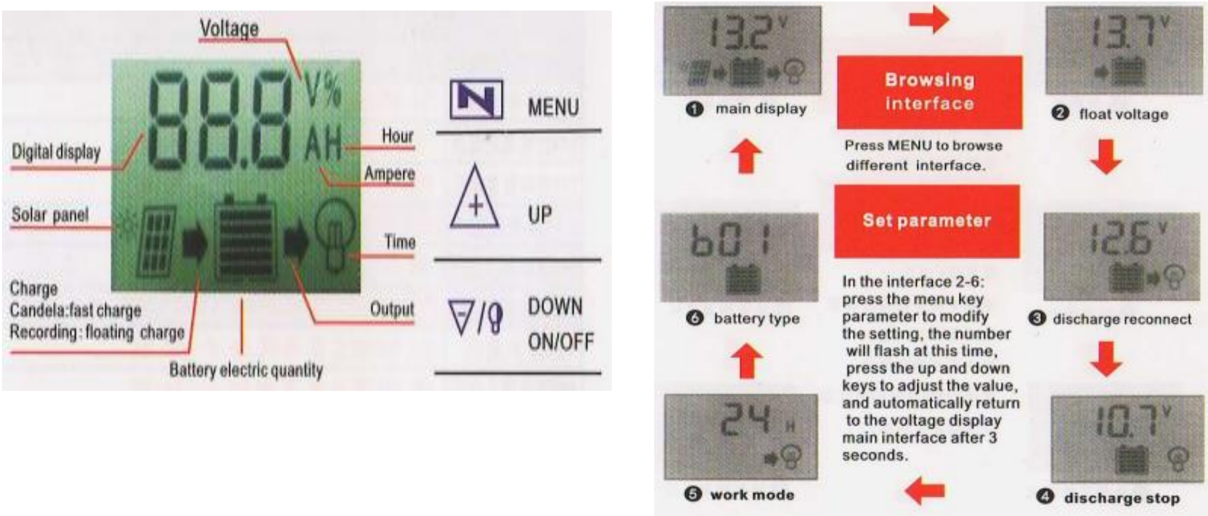


Figure 3.8 Menu button in the solar charge controller display and its options

Solar panels offer a sustainable and environmentally friendly means of generating electricity, yet harnessing this energy presents notable challenges for designers. The output of solar panels is highly contingent on factors like incident angle and light intensity, making power supply parameters volatile. Even minimal shading can significantly diminish efficiency. To address these issues, a comprehensive solution involving a solar panel, battery, and charge controller is optimal. The SCC-30A-PWM-LCD solar charge controller (depicted in Figure 3.7) enables the connection of panels in series or parallel, ensuring consistent power supply parameters and enabling surplus energy storage. A PWM (Pulse Width Modulation) solar charge controller, depicted in Figure 3.8, plays a pivotal role in regulating battery charging and discharging in solar power systems. By modulating voltage and current delivery based on battery charge status, it sustains batteries at optimal voltage levels, preventing overcharging or deep discharging and thereby extending battery lifespan. During charging, the PWM controller intermittently adjusts pulse width to maintain desired battery voltage, reducing duty cycle as the battery approaches capacity to prevent overcharging. Conversely, during discharging, it regulates current flow to ensure stable power supply, adjusting pulse width to prevent deep discharging and safeguard battery integrity. This dynamic control mechanism ensures efficient energy utilization while safeguarding battery health, contributing to the reliability and longevity of solar power systems.

Specifications

- Regulation Type: PWM
- Rated voltage of solar charge controller: 12 (or) 24 V
- Rated current of solar charge controller: 30 A
- Permitted voltage range:
  - ≤ 23 V for 12 V battery - The range of the highest operating voltage of a set of panels connected to one input of the controller
  - ≤ 46 V for 24 V battery - The range of the highest operating voltage of a set of panels connected to one input of the controller
- Output voltage: Equal to the voltage at the battery terminals
- Battery charging current: max. 30 A
- Load Current: max. 10 A
- Main features:
  - 2 x USB power output: 5 V / 2 A,
  - The device is designed to charge only AGM, gel and lead-acid batteries,
  - LCD display,
  - LED diodes indicating the device operation status,
  - Operation modes:
    - 24H - the load is powered all the time
    - 1H ... 23H - the load is powered for the selected

number of hours after sunset

- Dimensions: 134 x 70 x 30 mm

3.13 SUMMARY

This chapter outlines the essential components of the autonomous solar grass-cutting robot. It includes an Arduino Uno microcontroller for control, an L298N motor driver for driving the motors, DC motors for propulsion, a 18650 lithium battery for power, and a PWM solar charge controller for efficient battery charging. Additionally, the system incorporates an IR sensor for obstacle detection and navigation. These components are interconnected and programmed to enable autonomous operation, with the robot powered by solar energy for sustainable and eco-friendly grass cutting.

.

CHAPTER 4
SOFTWARE DESCRIPTION

4.1 INTRODUCTION

This chapter deals with the intricacies of programming the autonomous solar grass-cutting robot. This section elucidates the algorithms, codes, and logic utilized to orchestrate the robot's autonomous functions. It explores how the Arduino Uno microcontroller communicates with sensors, motor drivers, and other hardware components to execute tasks such as path memorization, navigation, and grass-cutting. Through this software, the robot achieves intelligent decision-making and seamless operation in various outdoor environments.

4.2 ALGORITHMS

4.2.1 ALGORITHM FOR STATE-BASED CONTROL

Step 1: Initialize the robot's state to an initial state, such as "Idle" or "Start."

Step 2: Continuously read sensor data to gather information about the robot's environment. This includes data from the IR sensor for obstacle detection and navigation

Step 3: Based on the sensor readings and current state, determine the appropriate state transition. For example, if we pressed IR remote, it will be detected, transition according to the state.

Step 4: Execute the tasks associated with the current state.

Step 5: Continuously evaluate the conditions in the environment and the robot's status to determine if a state transition is needed.

Step 6: Repeat steps 2-5 in a loop to enable continuous monitoring, decision-making, and action execution by the robot.

Step 7: Define termination conditions, such as reaching a designated endpoint or low battery level, to exit the algorithm gracefully when necessary.

4.3 PROCESS FLOW CHART

4.3.1 PROCESS FLOW CHART FOR STATE-BASED CONTROL ALGORITHM

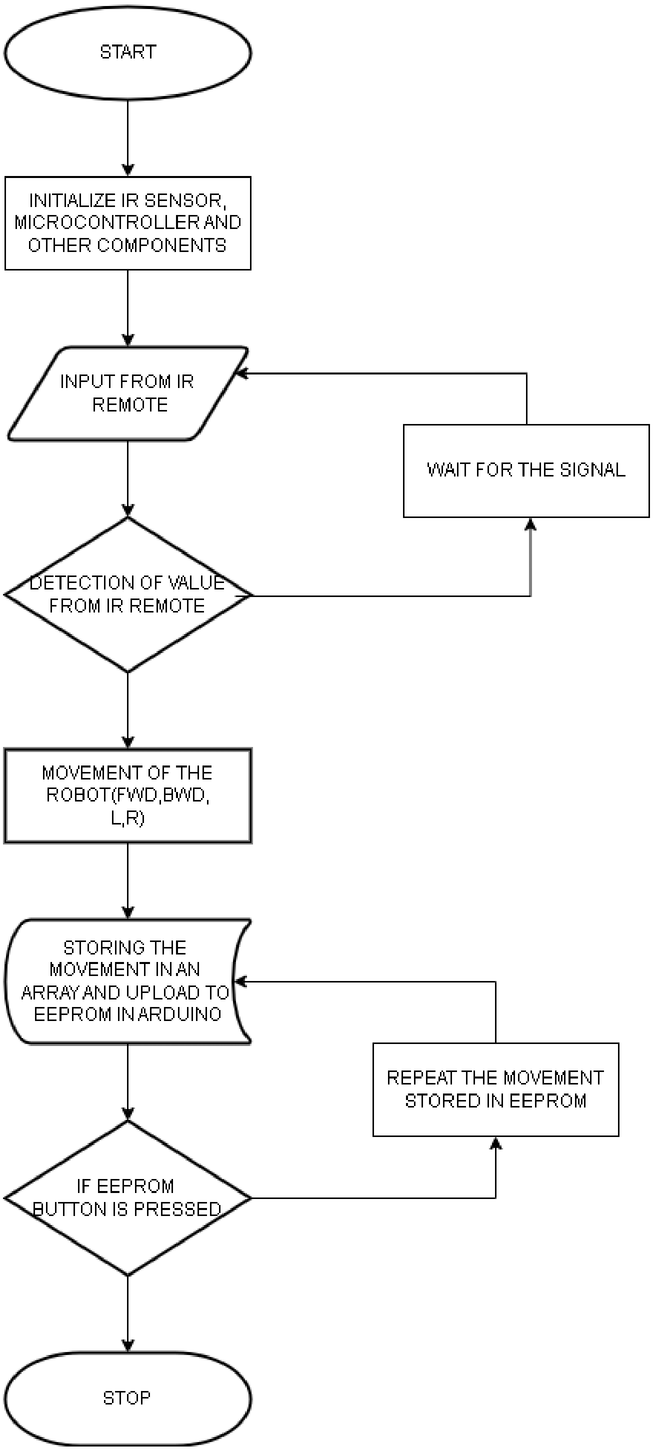


Figure 4.1 Process Flow Chart for State-based control Algorithm

The flowchart shown in Figure 4.1 describes a system for controlling a robot with an IR remote control. The system starts by initializing all the necessary components, getting the robot ready to receive commands. The user then presses a button on the IR remote, sending a signal to the robot. The robot interprets this signal, figuring out which direction (forward, backward, left, or right) the user wants it to move in. The robot then performs that movement and, interestingly, stores that instruction in a special memory that retains data even when powered off. This allows the user to create a sequence of movements for the robot using the remote control. Finally, the robot keeps repeating this stored sequence until the user presses a specific button on the remote, likely a stop command, to halt its movement.

4.4 SUMMARY

This chapter outlines the programming aspects of the autonomous solar grass-cutting robot. The implementation of a State-based control algorithm, enabling the robot to autonomously navigate, detect obstacles, and execute grass-cutting tasks. The algorithm is executed within the Arduino IDE, utilizing the Arduino Uno microcontroller and various sensors and actuators. The flowchart illustrates the system's operation, from initializing components to interpreting user commands via an IR remote control. Through comprehensive programming in the Arduino IDE, the robot achieves intelligent decision-making and seamless operation, making it capable of efficiently maintaining outdoor environments.

CHAPTER 5
EXPERIMENTAL RESULTS

5.1 INTRODUCTION

In this chapter the performance assessment of the autonomous solar grass-cutting robot. This section presents the findings obtained from field tests aimed at evaluating the robot's functionality and effectiveness in outdoor environments. Through rigorous experimentation and data analysis, we aim to provide insights into the robot's navigation accuracy, path memorizing capabilities, grass-cutting efficiency, and overall reliability.

5.2 RESULTS IN ARDUINO IDE

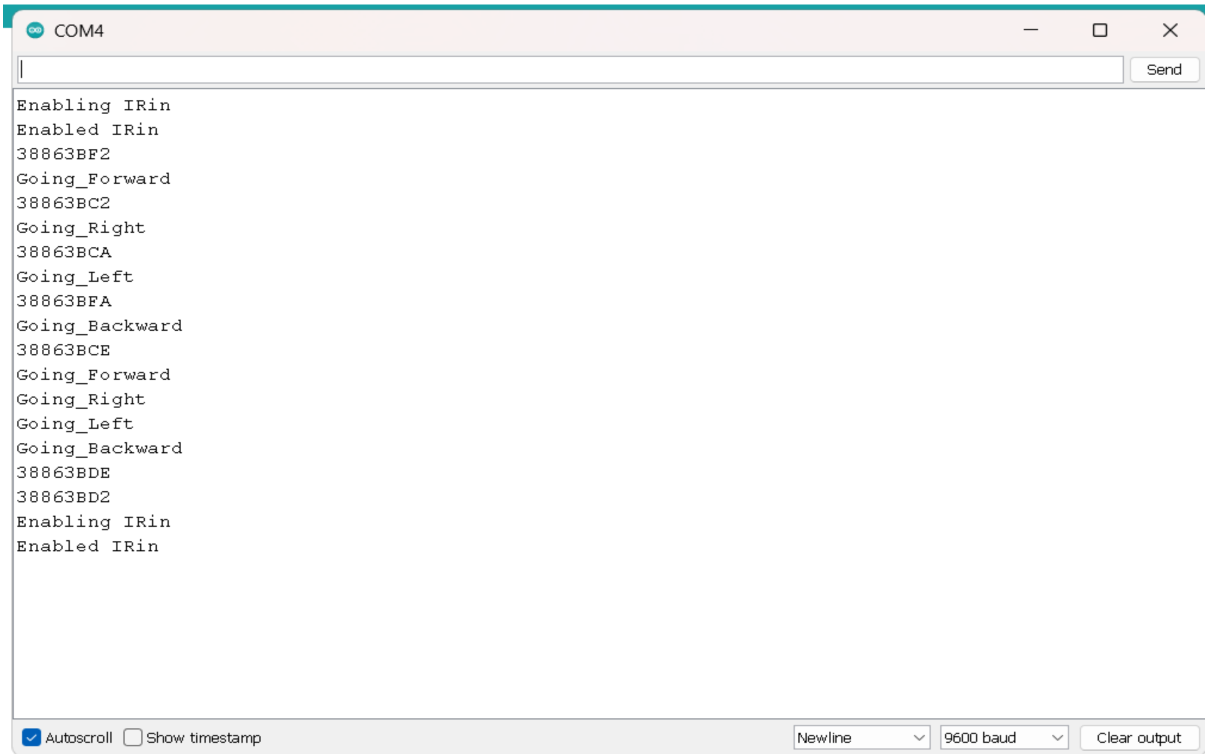


Figure 5.1 Overall simulation output obtained from Arduino ide

```
Enabling IRin
Enabled IRin
```

Figure 5.2 Enabling of IR sensor

```
38863BF2
Going_Forward
38863BC2
Going_Right
38863BCA
Going_Left
38863BFA
Going_Backward
```

Figure 5.3 Represents movement (fwd, bwd, l, r) of the robot

```
38863BCE
Going_Forward
Going_Right
Going_Left
Going_Backward
```

Figure 5.4 The output stored in the temporary storage

```
38863BDE
38863BD2
Enabling IRin
Enabled IRin
```

Figure 5.5 The movements are stored in EEPROM and the Arduino is reset.

The results obtained in the Arduino IDE showcase the overall simulation output shown in the Figure 5.1, including the successful enabling of the IR sensor in the Figure 5.2. This sensor facilitated the interpretation of commands for robot movement (forward, backward, left, right) shown in the Figure 5.3. The output data was stored in temporary memory for immediate processing shown in the Figure 5.4. As shown in the Figure 5.5, the movement instructions were permanently stored in the EEPROM (Electrically Erasable Programmable Read-Only Memory), ensuring data retention even after the Arduino reset for continued autonomous operation

5.3 RESULTS FROM HARDWARE SYSTEM

As shown in the Figure 5.6 and Figure 5.7, the results from autonomous solar grass-cutting robot demonstrated proficient functionality. The motor drivers efficiently regulated motor movements, ensuring precise navigation. The IR sensor accurately detected obstacles, enabling the robot to autonomously navigate around them. The DC motors exhibited robust performance, effectively driving the cutting blades for grass cutting. The lithium battery provided sufficient power for extended operation, while the DC-DC buck converter ensured stable voltage supply. The solar panel effectively charged the battery, enhancing the robot's autonomy.

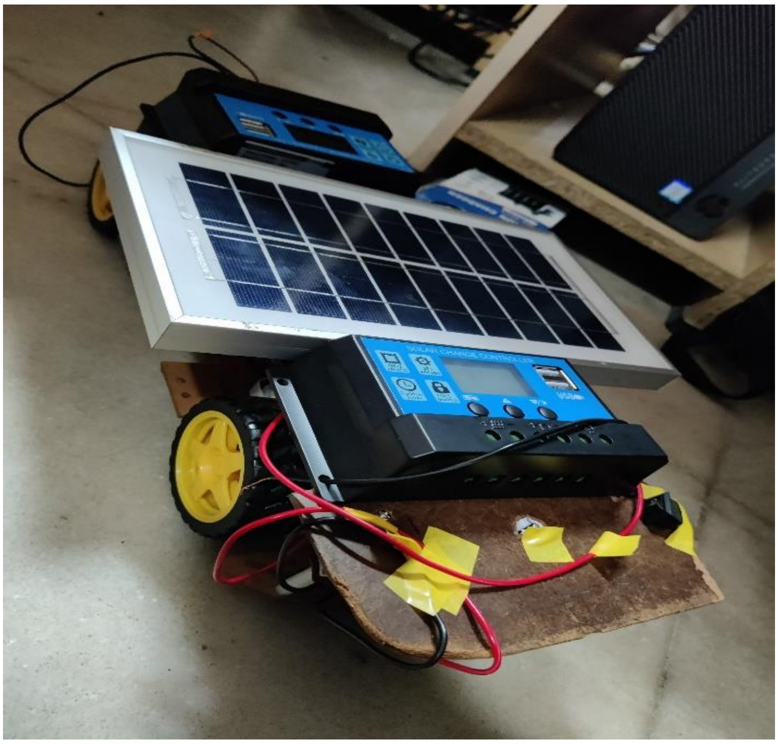


Figure 5.6 The proposed Autonomous solar grass cutting robot

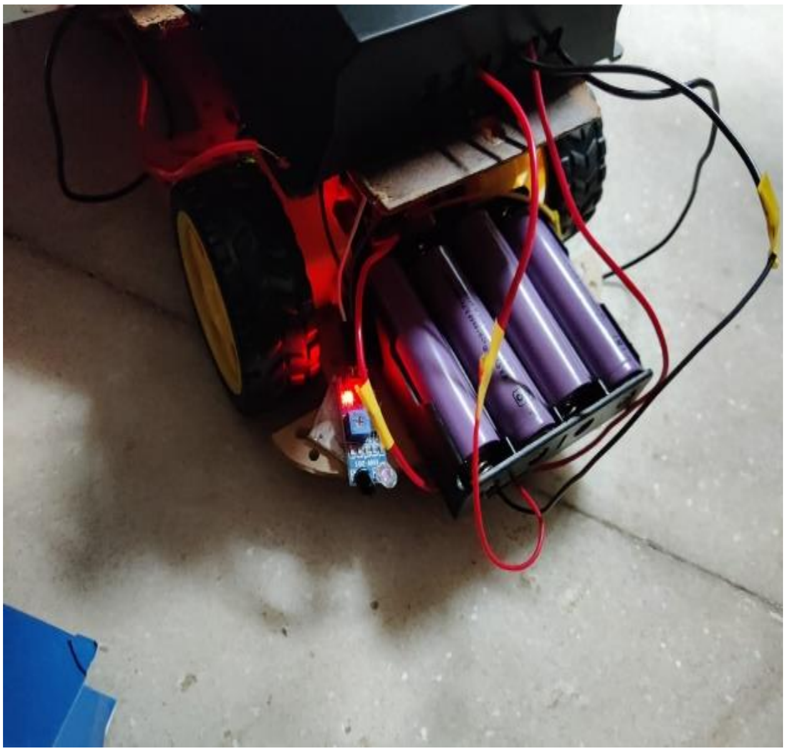


Figure 5.7 Connection of battery to the solar charge controller

Figure 5.8 Connection of solar panel to the charge controller

The figure 5.8 is a visual documentation of the hardware system, showcasing the physical components and their interconnections. The photographs offer a detailed view of the assembled robot, including the Arduino Uno board, motor driver, IR sensor, DC motors, lithium-ion battery, DC-DC buck converter module, solar panel, PWM solar charge controller, and other essential elements. These images provide insight into the arrangement and configuration of the hardware components, highlighting their placement within the robot chassis and their connectivity via wires and connectors. Additionally, close-up shots illustrate specific features of individual components, such as ports, switches, and indicators, contributing to a comprehensive understanding of the hardware setup. Overall, these photo results offer valuable visual insights into the physical implementation of the autonomous solar grass cutting robot, serving as a reference for future development and troubleshooting endeavours.

5.4 SYSTEM PERFORMANCE EVALUATION

The autonomous robot underwent evaluation during daylight hours, starting from 11 am and lasting for 3 hours. Testing occurred on a level lawn with a slight maximum slope of 2 degrees, typical for the lawn growing season. Evaluation criteria included area coverage, lawn availability, energy usage, and machine intervention frequency. Operational settings were optimized to minimize missed spots and ensure continuous operation. All machine configurations were checked before testing, with the battery fully charged. The cutting height was set to 45 mm. If the machine encountered minor issues, it resumed operation immediately; otherwise, any downtime was added to the test duration. The table 5.1 shows the value obtained from the machine performance. We obtained a navigation accuracy of 95% and grass cutting efficiency of 90% while the robot being operated autonomously.

5.1 Table The value obtained from machine performance evaluation

S.NO	METRICS	VALUES
1.	Navigation Accuracy	95%
2.	Grass-Cutting Efficiency	90%
3.	Battery Life (hours)	3
4.	Total Area Covered (m^2)	100

5.5 SOLAR PANEL POWER EVALUATION

Solar Panel Power Evaluation shown in table 5.2 assesses the efficiency and output of solar panels under various conditions. It involves measuring factors such as solar irradiance, panel temperature, and shading effects to determine the panel's performance and energy production potential. This evaluation helps optimize solar panel placement and system design for maximum energy generation. Some of the calculation involves

Solar Irradiance Calculation (Intensity of Solar panel)

Solar irradiance, measured in watts per square meter (W/m^2), represents the amount of solar power received per unit area. It can be calculated using the formula:

- Solar Irradiance = Solar Power / Solar Panel Area eqn.1

Where:

- Solar Power is the total power incident on the solar panel (in watts).
- Solar Panel Area is the surface area of the solar panel exposed to sunlight (in square meters).

Table 5.2 The value obtained from solar panel power evaluation

S NO	Time (hh/mm)	Temperature (approx) in Celsius	Voltage in V	Intensity of Solar panel (W/m^2)
1	09.30	35	18.2	173
2	10.00	40	18.32	179
3	11.00	50	18.50	190
4	12.00	54	18.85	231
5	01.00	58	19.01	240
6	02.00	54	18.80	225
7	03.00	45	18.46	185
8	04.00	36.5	18.05	170

5.6 EVALUATION OF ENERGY CONSUMPTION OF THE MACHINE

Energy consumption of the machine shown in Table 5.3 involves analyzing the amount of electrical power consumed by the machine during operation. This evaluation assesses various factors such as power usage in different modes (idle, active), energy efficiency of components, and overall power consumption patterns. It aims to optimize energy usage, minimize wastage, and improve the sustainability and cost-effectiveness of the machine's operation. Measurements, data logging, and analysis techniques are commonly used to evaluate and optimize energy consumption. The robot's cutting blade and translational motion are powered mostly by mechanical energy that is transferred from the sun to the batteries. Energy consumption (EC) in W/H is therefore represented as follows:

EC = P*T W/h=1940W/h eqn.2

Where,

P = Power Requirement by the robot

T = Total time of operation

Table 5.3 Evaluation of energy consumption of the machine

S.NO	**COMPONE- -NTS**	**ENERGY CONSUMPTIO- -N (V)**	**POWER CONSUMPTIO- N (W)**	**TOTAL ENERGY CONSUMPTIO- -N(EC) (Wh/day)**
1	Motors	12	50	720
2	IR Sensor	5	5	60
3	Arduino Uno	5	10	120
4	DC-DC Buck Converter	5	5	60
5	Other Electronics	5	80	980
			Total power consumption	1940 W/h

.

5.7 SUMMARY

This chapter deals with the experimental results affirm the robust performance of the autonomous solar grass-cutting robot. Comprehensive tests conducted via Arduino IDE and hardware evaluations validated the system's functionality, ensuring precise sensor integration and motor control. Energy consumption assessments underscored efficient power utilization, guiding sustainability considerations and operational optimization.

CHAPTER 6
CONCLUSION AND FUTURE SCOPE

6.1 CONCLUSION & FUTURE SCOPE

The goal of the proposed work is to development the autonomous solar grass-cutting robot with path memorizing mechanism which marks a significant achievement in the realm of automated lawn maintenance. The sole purpose of today's equipment is to eliminate greenhouse gas emissions, which seem to be the principal contributor to climate change. This solar-powered lawn cutter will achieve the project's environmental aims while also being low-cost to administer thanks to the shortage of a fuel source The integration of Arduino with IR sensors facilitated precise control, enabling the robot to navigate and cut grass autonomously. Furthermore, the utilization of solar power underscores sustainability, reducing reliance on traditional energy sources. The experiment results showcased the system's efficiency and reliability, demonstrating its potential for real-world applications in agriculture and landscaping industries.

Firstly, integrating advanced sensors such as GPS and LiDAR could improve navigation accuracy and obstacle detection, enhancing the robot's efficiency in complex environments. Additionally, incorporating machine learning algorithms can enable the robot to adapt to varying terrain and optimize cutting patterns based on grass growth patterns. Furthermore, enhancing the system's energy storage capacity and implementing smart charging algorithms can enhance its autonomy and operational flexibility. Collaborations with industry stakeholders can facilitate field trials and real-world deployment, validating the system's performance under diverse conditions and paving the way for commercialization.

REFERENCES

[1] Adeodu.A.O, Daniyan S. Ebimoghan.S. ,Akinola.O "Development of an Embedded Obstacle Avoidance and Path Planning Autonomous Solar Grass Cutting Robot for Semi-Structured Outdoor Environment" 2018 IEEE 7th International Conference on Adaptive Science & Technology (ICAST)

[2] Balaganesan S.M, Abishek.R, Aravinth.R, Maignanamoorthy A S N, "Solar Based Grass Cutter Robot", 2023 Second International Conference on Augmented Intelligence and Sustainable Systems (ICAISS).

[3] Dr. Chaudhari J G, Akash S Ingole, Aakash Z Patel, Kunal R Bhagat, Ashwini S Gaurkhede "Smart Solar Based Grass Cutter" International Journal of Advanced Research in Science, Communication and Technology (IJARSCT) Volume 2, Issue 1, May 2022

[4] Dutta P.P, Baruah A, Konwar A and Kumar V. "A Technical Review of Lawn Mower Technology. AUBU-Journal of Engineering Technology. 1(1):1-7, 2016

[5] Dr. Khan Sohelrana, Ayesha Sultana, Shireen Fatima "A Review on Smart IoT based Gesture Controlled Grass Cutting Vehicle", Proceedings of the Fourth International Conference on Trends in Electronics and Informatics (ICOEI 2020)

[6] Ms. Lanka Priyanka, Mr. Prof. J. Nagaraju, Mr. Vinod Kumar Reddy, "Fabrication of Solar Powered Grass Cutting Machine", International Journal and Magazine of Engineering, Technology, Management and Research, Vol. 2, 2015, 386- 390

[7] N. Nouorami-Vatami, M. Bosse, J. Roberts, M. Dunbabin. "Practical Path Planning and Obstacle Avoidance for Autonomous Mowing" International Conference on Robotics and Automation. Pages 1899-1906.

[8] Okarfor B, “Simple Design of Self Powered Lawn Mower” International Journal of Engineering and Technology. 3(10):933-938, 2013

[9] Pratik Patil, Ashwini Bhosale, Prof. Sheetal Jagtap, “Design and Implementation of Automatic Lawn Cutter”, International Journal of Emerging Technology and Advanced Engineering, 2014.

[10] Md. Rawshan Habib, Koushik Ahmed, Naureen Khan, Mahbubur Rahman Kiran, Md. Ahasonul Habib, Md. Tanvir Hasan, and Omar Farrok “PID Controller Based Automatic Solar Power-driven Grass Cutting Machine” (IC4ME2), 11-12 July, 2019

[11] Sivarao, T J S Anand, Hambali, Minhat, Faizul, “Review of Automated Machines towards few Approach in Developing SemiAutomated Grass Cutter”, International Journal of Mechanical and Mechatronics Engineering IJMME-IJENS, 2010.

PAPER PUBLICATION

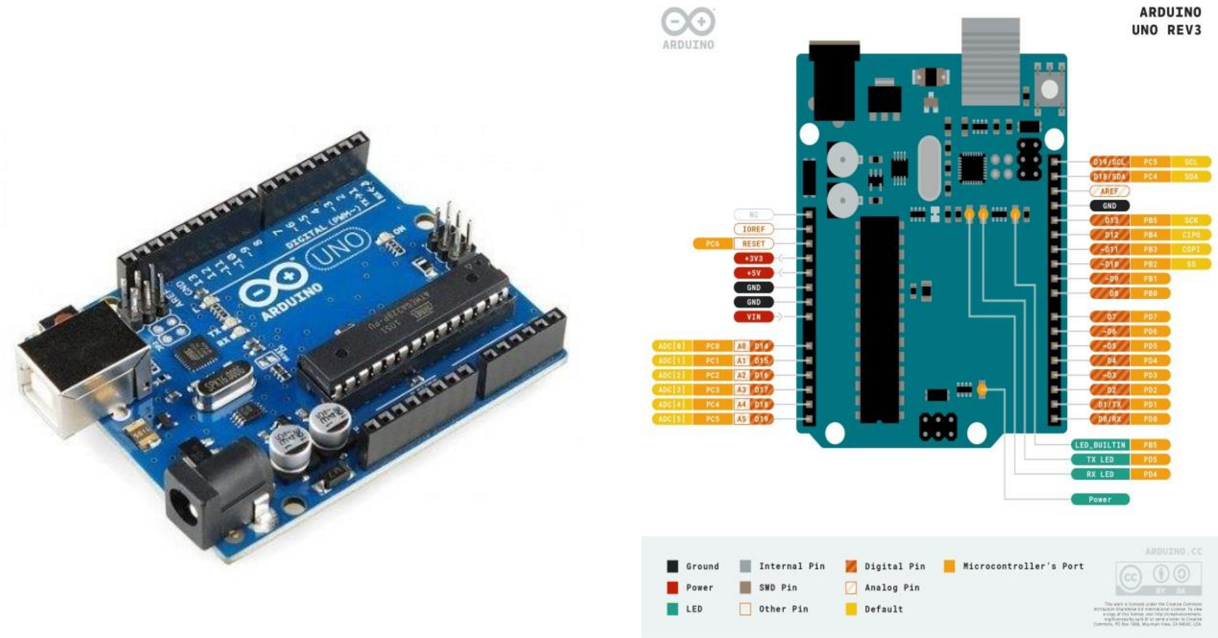
Manikandababu.C.S, Venugopal.R, Vijayakrishna.R, Vishnuprasad.Y, “Development of an Autonomous Solar Grass Cutting Robot with a Path Memorizing Mechanism”, 2024 3rd International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI-2024), Chennai Section [Accepted on 05.04.2024 and to be presented on 09.05.2024]

APPENDIX – I

SYSTEM SPECIFICATIONS:

S.NO	COMPONENTS	SPECIFICATIONS	QUANTITY
1	SOLAR PANEL	18V, 20watts	2
2	DC MOTOR (for cutter)	• Diameter 35MM • Length 50MM • Weight: 160g • 3V, 8400 rpm, 1.30A	1
3	BATTERIES	LITHIUM ION 18650 -2000 MAH	8
4	MICROCONTROLLER	ARDUINO UNO 3	1
5	INFRARED SENSOR	Detection distance: 2-40 cm	1
6	SOLAR CHARGE CONTROLLER	PWM,12V/24V,10A	2
7	L298N MOTOR DRIVER	Operating Voltage (VDC): 3 ~ 30V	2
8	DC-DC BUCK CONVERTER STEP DOWN MODULE	• Input: DC 3V to 40V • Output: DC 1.5V to 35V	2
9	BLADE	Stainless steel, Thickness = 0.5mm	1
10	ROBOT WHELLS WITH DC MOTOR	BLDC 300rpm	4

ARDUINO UNO 3



yuio Arduino uno 3 board and its pin diagram

The Arduino Uno 3 shown in Figure 3.5 is a widely-used microcontroller board renowned for its versatility, ease of use, and extensive community support. Powered by the Atmega328P microcontroller, it offers a plethora of features ideal for a wide range of projects, including the development of the autonomous solar grass-cutting robot. With its compact form factor and 14 digital input/output pins (of which 6 can be used as PWM outputs), the Arduino Uno provides ample connectivity options for interfacing with various sensors, actuators, and peripherals. Additionally, it boasts 6 analog input pins, enabling precise analog sensor readings for applications requiring sensor data acquisition.

FEATURES

●High performance, low power AVR® 8-bit microcontroller

● Advanced RISC architecture

    - 131 powerful instructions – most single clock cycle execution
    - 32 * 8 general purpose working registers
    - Fully static operation
    - Up to 16MIPS throughput at 16MHz
    - On-chip 2-cycle multiplier
  - High endurance non-volatile memory segments
    - 32K bytes of in-system self-programmable flash program memory
    - 1Kbytes EEPROM
    - 2Kbytes internal SRAM
    - Write/erase cycles: 10,000 flash/100,000 EEPROM
    - Optional boot code section with independent lock bits
    - In-system programming by on-chip boot program
    - True read-while-write operation
    - Programming lock for software security
- Peripheral features
  - Two 8-bit Timer/Counters with separate prescaler and compare mode
  - One 16-bit Timer/Counter with separate prescaler, compare mode, and capture mode
  - Real time counter with separate oscillator
  - Six PWM channels
  - 8-channel 10-bit ADC in TQFP and QFN/MLF package
  - Temperature measurement

- Programmable serial UART
- Master/slave SPI serial interface
- Byte-oriented 2-wire serial interface (Phillips I2 C compatible)
- Programmable watchdog timer with separate on-chip oscillator
- On-chip analog comparator
- Interrupt and wake-up on pin change

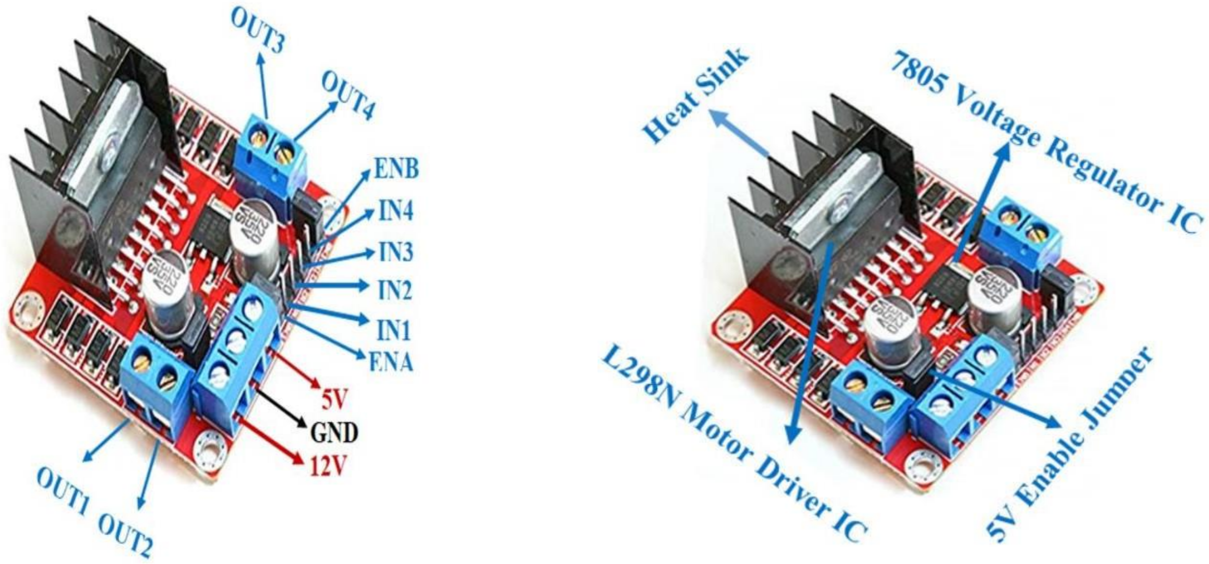
- Special microcontroller features
    - Power-on reset and programmable brown-out detection
    - Internal calibrated oscillator
    - External and internal interrupt sources
    - Six sleep modes: Idle, ADC noise reduction, power-save, power-down, standby, and extended standby
- I/O and packages
    - 23 programmable I/O lines
    - 32-lead TQFP, and 32-pad QFN/MLF
- Operating voltage:
    - 2.7V to 5.5V for ATmega328P
- Temperature range:
    - Automotive temperature range: –40°C to +125°C
- Speed grade:
    - 0 to 8MHz at 2.7 to 5.5V (automotive temperature range: –40°C to +125°C)

- 0 to 16MHz at 4.5 to 5.5V (automotive temperature range: –40°C to +125°C)

● Low power consumption

- Active mode: 1.5mA at 3V - 4MHz
- Power-down mode: 1µA at 3V

L298N-MOTOR DRIVER



L298N motor driver and its pin diagram

The L298N motor driver shown in Figure 3.6 is a popular integrated circuit (IC) commonly used to control DC motors and stepper motors in a wide range of projects, including robotics and automation applications. It consists of two H-bridge circuits, each capable of driving a single DC motor or one winding of a stepper motor. The L298N can handle a wide range of motor voltages (up to 46V) and currents (up to 2A per channel), making it suitable for controlling various types of motors. Its robust design and built-in protection features, such as overcurrent and thermal protection, enhance reliability and safety during operation. The IC features four input pins (IN1, IN2, IN3, and IN4) for controlling the direction and speed of the connected motors. By toggling these input pins in specific sequences, users can control the rotation direction (forward, reverse) and speed (via pulse-width modulation) of the motors. Additionally, the L298N includes enable pins (ENA and ENB) for each H-bridge, allowing users to enable or disable the corresponding motor outputs as needed. This feature is particularly useful for energy-saving purposes or when implementing braking functions. The L298N also provides built-in diodes (called freewheeling diodes) across the motor outputs, which help protect the IC and other components from voltage spikes generated by the motor during deceleration or when the motor is turned off.

Features & Specifications:

- Driver Model: L298N
- Driver Chip: Double H Bridge L298N
- Motor Supply Voltage (Maximum): 46V
- Motor Supply Current (Maximum): 2A
- Logic Voltage: 5V
- Driver Voltage: 5-35V
- Driver Current:2A
- Logical Current:0-36mA
- Maximum Power (W): 25W
- Current Sense for each motor
- Heatsink for better performance
- Power-On LED indicator

DC MOTOR

DC motor used for grass cutter and DC motor used in wheels

DC (Direct Current) motors shown in Figure 3.7 are electromechanical devices widely used for converting electrical energy into mechanical motion. They operate based on the principles of electromagnetism, where the interaction between magnetic fields and current-carrying conductors generates rotational motion. A typical DC motor consists of two main components: the stator and the rotor. The stator comprises a stationary set of magnets or electromagnets, while the rotor consists of a rotating armature with coils wound around it. When a DC voltage is applied to the motor terminals, current flows through the coils, creating a magnetic field that interacts with the magnetic field of the stator. This interaction produces a torque on the rotor, causing it to rotate. DC motors come in various types, including brushed and brushless motors. Brushed DC motors use brushes and a commutator to control the direction of current flow in the rotor windings, resulting in rotational motion. Brushless DC motors, on the other hand, utilize electronic commutation to achieve the same effect, eliminating the need for physical brushes and offering advantages such as improved efficiency and reliability.

Motor(wheels) Specifications:

- Standard 130 Type DC motor
- Operating Voltage: 4.5V to 9V
- Recommended/Rated Voltage: 6V
- Current at No load: 70mA (max)
- No-load Speed: 9000 rpm
- Loaded current: 250mA (approx)
- Rated Load: 10g*cm
- Motor Size: 27.5mm x 20mm x 15mm
- Weight: 17 grams

Motor (grass cutter) Specifications:

- Diameter 35MM
- Length 50MM
- Weight: 160g
- 3V, 8400 rpm, stroom 1.30A
- 4.5V, 12400 rpm, stroom 1.47A
- 6V, 16500 rpm, stroom 1.60A
- 7.5V, 20800 rpm, stroom 1.70A
- 9V, 33100 rpm, stroom 2.50

LITHIUM ION BATTERY

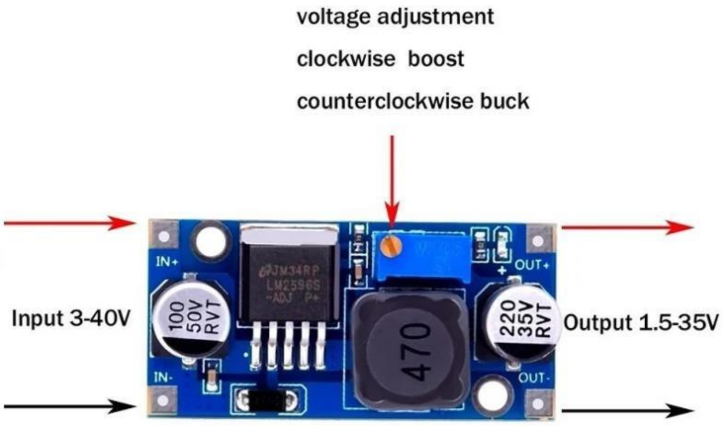
Lithium-ion battery 18650 and its pin diagram

The lithium-ion battery shown in Figure 3.8 is a rechargeable cylindrical cell widely used in various electronic devices due to its high energy density, long cycle life, and relatively low self-discharge rate. The name "18650" refers to its dimensions, with an approximate diameter of 18mm and a length of 65mm. These batteries utilize lithium-ion technology, where lithium ions move between the positive and negative electrodes during charging and discharging cycles. The electrodes are typically made of materials such as lithium cobalt oxide ($LiCoO_2$) for the positive electrode and graphite for the negative electrode. The electrolyte is a lithium salt dissolved in a solvent, facilitating ion movement. 18650 batteries are known for their high energy density, meaning they can store a large amount of energy in a relatively compact size. This makes them ideal for applications where space and weight are crucial factors, such as in portable electronic devices like laptops, power tools, and electric vehicles. One of the key advantages of lithium-ion 18650 batteries is their long cycle life, which refers to the number of charge-discharge cycles they can undergo before experiencing significant degradation in performance. Proper care and maintenance, such as avoiding deep discharges and extreme temperatures, can help prolong their lifespan. Safety is also an important consideration with lithium-ion batteries. While they offer high energy density and rechargeability, they can be prone to overheating and thermal runaway if not handled properly. Built-in safety features, such as overcharge and over-discharge protection circuits, help mitigate these risks and ensure safe operation.

18650 Cell Features and Technical Specification:

- Nominal Voltage: 3.6V
- Nominal Capacity: 2,850 mAh
- Minimum Discharge Voltage: 3V
- Maximum Discharge current: 1C
- Charging Voltage: 4.2V (maximum)
- Charging current: 0.5C
- Charging Time: 3 hours (approx)
- Charging Method: CC and CV
- Cell Weight: 48g (approx)
- Cell Dimension: 18.4mm (dia) and 65mm (height)

DC-DC BUCK CONVERTER STEP DOWN MODULE



LM2596 Step Down DC-DC Buck Converter Adjustable Module

A DC-DC buck converter step-down module is an electronic device used to efficiently lower the voltage level of a direct current (DC) power source to a desired level. It operates on the principle of pulse width modulation (PWM) to regulate the output voltage by controlling the duty cycle of a switching transistor. The module typically consists of several key components, including an input capacitor, an inductor, a switching transistor (such as a MOSFET), a diode, and an output capacitor. During operation, the input voltage is applied to the converter, and the switching transistor rapidly switches on and off. When the transistor is on, current flows through the inductor, storing energy. When the transistor is off, the inductor releases energy, and the diode conducts, providing a continuous current flow to the load. By adjusting the duty cycle of the switching transistor, the output voltage can be regulated to the desired level. One of the main advantages of DC-DC buck converters is their high efficiency, typically ranging from 80% to 95%, depending on the specific design and operating conditions. This efficiency is achieved by minimizing power losses during the voltage conversion process, resulting in less heat generation and better energy utilization.

Technical Specifications:

- Input Voltage: 3.2V- 40V DC
- Output Voltage: 1.25V- 35V
- DC Output Current: 2A, Max 3A (Additional Heat Sink is required)
- Rectification: Non-Synchronous Rectification
- Thermal shutdown and current-limit protection
- Excellent line and load regulation specifications
- High efficiency
- TTL shutdown capability
- Low power standby mode, IQ, typically 80µA

.

SOLAR PANEL

Solar panel

Solar panels, also known as photovoltaic (PV) panels shown in Figure 3.10 are devices that convert sunlight directly into electricity using the photovoltaic effect. They consist of multiple solar cells made of semiconductor materials, typically silicon, arranged in a grid-like pattern on a flat surface. When sunlight strikes the surface of a solar panel, photons from the sunlight are absorbed by the semiconductor material. This absorption of photons creates an electric field across the layers of the semiconductor, generating a flow of electrons. This flow of electrons creates a direct current (DC) electrical output, which can then be used to power electrical devices or stored in batteries for later use. Solar panels are designed to harness sunlight efficiently and convert it into usable electricity. The efficiency of a solar panel depends on various factors, including the quality of the materials used, the design of the solar cells, and environmental conditions such as sunlight intensity and temperature. Solar panels come in different types and configurations, including monocrystalline, polycrystalline, and thin-film solar panels. Monocrystalline panels are made from single-crystal silicon and offer high efficiency and longevity. Polycrystalline panels are made from multiple silicon crystals and are slightly less efficient but more cost-effective. Thin-film panels are made from various materials deposited in thin layers and are lightweight and flexible, making them suitable for a wide range of applications. Solar panels are used in various applications, including residential and commercial rooftop installations, solar farms, portable solar chargers, and off-grid power systems.

Specifications

- Polycrystalline Solar Panel.
- Wattage: 3W.
- Voltage: 12V.
- Length- 28cm
- Breadth- 12.5cm
- Thickness- 1.5cm

Efficiency- 18.6%

3.1 BLADE

Blades used in grass-cutting robots are designed for durability, sharpness, and efficiency. They are often made of high-quality stainless steel or other sturdy materials to withstand the rigors of cutting thick or tough grasses. Additionally, the blades may have special coatings or treatments to enhance their resistance to corrosion, wear, and tear. The design of the blade shown in Figure 3.2 can vary depending on the specific requirements of the grass-cutting robotand the type of grass it will be cutting. For example, some blades may have a straight edge for standard grass cutting, while others may feature serrated edges or specialized shapes for handling thicker vegetation or challenging terrain. The blades are typically attached to a motor or shaft within the grass-cutting robot's cutting mechanism. The motor rotates the blades at high speeds, allowing them to slice through the grass effectively.

Blade for grass cutter

Source code:

```
#include <IRremote.h>
#include <EEPROM.h>
/*
  Left Motor
*/
// IN 1
#define ena    5
#define in1    6
// IN 2
#define in2    7
/*
  Right Motor
*/
// IN 3
#define in3    8
// IN 4
#define in4    9
#define enb    10
// IR receiver
```

```
# define RECV_PIN 11
IRrecv irrecv(RECV_PIN);
decode_results results;
// HEX codes for buttons
#define FWD       0x1FED827 // go forward
#define LFT       0x1FE30CF // go left
#define RGT       0x1FE708F // go right
#define BWD       0x1FEF00F // go backward
#define STOP      0x1FEB04F // stop
#define RPEAT     0x1FE10EF // repeat the stored sequence of movement from the temporary memory(automatically stores)
#define DEL       0x1FE20DF // delete the stored sequence of movement from temporary memory(EQ)
#define PERST     0x1FE7887 // copy the sequence from temp. memory to the peramanent memory(EEPROM)
#define PLAYEPROM 0x1FE58A7 // repeat the sequence stored in EEPROM(FASTFWD)
#define RESET     0x1FE48B7 // Resets the Arduino Board(RED)
unsigned long int value = 0;
byte seq = 0; //stores the current number of executed sequences
byte seq_Array[50];// array to store the movement sequence in terms of integers(1 for FWD, 2 for LEFT and so on..)

//counter for counting the number of times program pass through a movement function(fwd, lft etc.)
int fwd_Counter = -1;
int lft_Counter = -1;
int rgt_Counter = -1;
```

```
int bwd_Counter = -1;
int stp_Counter = -1;

//global "current time" variables for different movement functions(fwd, lft etc.)
unsigned long int current_Time0 = 0;// for FWD movement
unsigned long int current_Time1 = 0;// for LEFT movement
unsigned long int current_Time2 = 0;// for RIGHT movement
unsigned long int current_Time3 = 0;// for BWD movement
unsigned long int current_Time4 = 0;// for STOP

//total time spend by the pgm in executing the movement(fwd, lft etc.) for a particular movement counter
unsigned long int total_Fwd_Time[10];
unsigned long int total_Lft_Time[10];
unsigned long int total_Rgt_Time[10];
unsigned long int total_Bwd_Time[10];
unsigned long int total_Stp_Time[10];

#define RESET_PIN A0

void setup() {
  // set mode of the pins as output
  for (int i = 2; i <= 7; i++) {
    pinMode(i, OUTPUT);
  }
  // start serial communication
```

```
  Serial.begin(9600);
  // In case the interrupt driver crashes on setup, give a clue
  // to the user what's going on.
  Serial.println("Enabling IRin");
  irrecv.enableIRIn(); // Start the receiver
  Serial.println("Enabled IRin");
}
void loop() {
  if (irrecv.decode(&results)) {
    value = results.value;
    Serial.println(value, HEX);
    irrecv.resume(); // Receive the next value
    delay(200);
  }

  delay(100);
  check_Inst(value);
  value = 0;
}
void check_Inst(long int value)
{
  switch (value)
  {
    case FWD:
      go_Forward();
      delay(10);
      break;
```

```
case LFT:
  go_Left();
  delay(10);
  break;
case RGT:
  go_Right();
  delay(10);
  break;
case BWD:
  go_Backward();
  delay(10);
  break;
case STOP:
  go_Stop();
  delay(10);
  break;
case RPEAT:
  go_In_Seq();
  delay(10);
  break;
case DEL:
  del_From_Local_Mem();
  delay(10);
  break;
case PERST:
  write_To_Permt_Mem();
  delay(10);
```

```
      break;
    case PLAYEPROM:
      Read_Permt_Mem();
      delay(10);
      break;
    case RESET:
      pinMode(RESET_PIN, OUTPUT);
      digitalWrite(RESET_PIN, HIGH);
      break;
    default:
      value = 0;

  }
}
void go_Forward()
{
  movement_Inst_Fwd();
  current_Time0 = millis();
  int i = seq_Array[(seq - 1)];
  switch (i)
  {
    case 2:
      total_Lft_Time[lft_Counter + 1] = (current_Time0 - current_Time1);
      lft_Counter++;
      break;
    case 3:
      total_Rgt_Time[rgt_Counter + 1] = (current_Time0 - current_Time2);
```

```
      rgt_Counter++;
      break;
    case 4:
      total_Bwd_Time[bwd_Counter + 1] = (current_Time0 - current_Time3);
      bwd_Counter++;
      break;
    case 5:
      total_Stp_Time[stp_Counter + 1] = (current_Time0 - current_Time4);
      stp_Counter++;
      break;
  }
  seq_Array[seq] = 1;
  seq++;
}
void go_Left()
{
  movement_Inst_Lft();
  current_Time1 = millis();
  int i = seq_Array[(seq - 1)];
  switch (i)
  {
    case 1:
      total_Fwd_Time[fwd_Counter + 1] = (current_Time1 - current_Time0);
      fwd_Counter++;
      break;
```

```
      case 3:
        total_Rgt_Time[rgt_Counter + 1] = (current_Time1 - current_Time2);
        rgt_Counter++;
        break;
      case 4:
        total_Bwd_Time[bwd_Counter + 1] = (current_Time1 - current_Time3);
        bwd_Counter++;
        break;
      case 5:
        total_Stp_Time[stp_Counter + 1] = (current_Time1 - current_Time4);
        stp_Counter++;
        break;
    }
    seq_Array[seq] = 2;
    seq++;
  }
  void go_Right()
  {
    movement_Inst_Rgt();
    current_Time2 = millis();
    int i = seq_Array[(seq - 1)];
    switch (i)
    {
      case 1:
        total_Fwd_Time[fwd_Counter + 1] = (current_Time2 - current_Time0);
```

```
        fwd_Counter++;
        break;
      case 2:
        total_Lft_Time[lft_Counter + 1] = (current_Time2 - current_Time1);
        lft_Counter++;
        break;
      case 4:
        total_Bwd_Time[bwd_Counter + 1] = (current_Time2 - current_Time3);
        bwd_Counter++;
        break;

      case 5:
        total_Stp_Time[stp_Counter + 1] = (current_Time2 - current_Time4);
        stp_Counter++;
        break;
    }
    seq_Array[seq] = 3;
    seq++;
  }
  void go_Backward()
  {
    movement_Inst_Bwd();
    current_Time3 = millis();
    int i = seq_Array[(seq - 1)];
    switch (i)
    {
```

```
    case 1:
      total_Fwd_Time[fwd_Counter + 1] = (current_Time3 - current_Time0);
      fwd_Counter++;
      break;
    case 2:
      total_Lft_Time[lft_Counter + 1] = (current_Time3 - current_Time1);
      lft_Counter++;
      break;

    case 3:
      total_Rgt_Time[rgt_Counter + 1] = (current_Time3 - current_Time2);
      rgt_Counter++;
      break;
    case 5:
      total_Stp_Time[stp_Counter + 1] = (current_Time3 - current_Time4);
      stp_Counter++;
      break;
  }
  seq_Array[seq] = 4;
  seq++;
}
void go_Stop()
{
  movement_Inst_Stp();
  current_Time4 = millis();
  int i = seq_Array[(seq - 1)];
```

```
    switch (i)
    {
     case 1:
      total_Fwd_Time[fwd_Counter + 1] = (current_Time4 - current_Time0);
      fwd_Counter++;
      break;
     case 2:
      total_Lft_Time[lft_Counter + 1] = (current_Time4 - current_Time1);
      lft_Counter++;
      break;
     case 3:
      total_Rgt_Time[rgt_Counter + 1] = (current_Time4 - current_Time2);
      rgt_Counter++;
      break;
     case 4:
      total_Bwd_Time[bwd_Counter + 1] = (current_Time4 - current_Time3);
      bwd_Counter++;
      break;
    }
    seq_Array[seq] = 5;
    seq++;
   }
   void go_In_Seq(void)
   {
    value = 0;
```

```
for (int i = 0; i < (seq + 1); i++)
{
  int value1 = 0;
  value1 = seq_Array[i];
  switch (value1)
  {
    case 1:
      static int j = 0;
      go_Forward_Seq(j);
      j++;
      break;
    case 2:
      static int k = 0;
      go_Left_Seq(k);
      k++;
      break;
    case 3:
      static int l = 0;
      go_Right_Seq(l);
      l++;
      break;
    case 4:
      static int m = 0;
      go_Backward_Seq(m);
      m++;
      break;
    case 5:
```

```
        static int n = 0;
        go_Stop_Seq(n);
        n++;
        break;
      default:
        j = 0; k = 0; l = 0; m = 0; n = 0;
    }
  }
}
void del_From_Local_Mem()
{
  //set the movement counters to their default values
  fwd_Counter = -1;
  lft_Counter = -1;
  rgt_Counter = -1;
  bwd_Counter = -1;
  stp_Counter = -1;
  //set the total movement time to its default value
  for (int i = 0; i < 10; i++)
  {
    total_Fwd_Time[i] = 0;
    total_Lft_Time[i] = 0;
    total_Rgt_Time[i] = 0;
    total_Bwd_Time[i] = 0;
    total_Stp_Time[i] = 0;
  }
```

```
// Reset the sequence array(stored movement instructions)
  for (int i = 0; i < 50; i++)
  {
    seq_Array[i] = 0;
  }
  seq = 0;
}
void write_To_Permt_Mem()
{
  // total number of movement is stored in a random address i.e, 100
  EEPROM.write(100,seq);
  //writing the movement sequence
  for(int i=0; i<seq; i++)
  {
  EEPROM.write(2*i,seq_Array[i]);
  }
  //storing the time bw two successive movements
  for(int i=1; i<seq+1; i++)
  {
  if(seq_Array[i-1]==1)
  {
    static byte a=0;
    EEPROM.write(2*i-1,(total_Fwd_Time[a])/1000);           // Note: One location can store maximum value of 255, hence the time is divided by 1000 here. And then multiplied by 1000 while retreiving the data from EEPROM location
    a++;
   }
```

```
  else if(seq_Array[i-1]==2)
  {
    static byte b=0;
    EEPROM.write(2*i-1,(total_Lft_Time[b])/1000);
    b++;
    }
  else if(seq_Array[i-1]==3)
  {
    static byte c=0;
    EEPROM.write(2*i-1,(total_Rgt_Time[c])/1000);
    c++;
    }
  else if(seq_Array[i-1]==4)
  {
    static byte d=0;
    EEPROM.write(2*i-1,(total_Bwd_Time[d])/1000);
    d++;
    }
  else if(seq_Array[i-1]==5)
  {
    static byte e=0;
    EEPROM.write(2*i-1,(total_Stp_Time[e])/1000);
    e++;
    }
  }
 }
void Read_Permt_Mem()
```

```
{
  // Read from permanent memory
   byte x = EEPROM.read(100);
   for(int i=0; i<x; i++)
   {
    byte r = EEPROM.read(2*i);
    switch(r)
    {
      case 1:
        movement_Inst_Fwd();
        break;
      case 2:
        movement_Inst_Lft();
        break;
      case 3:
        movement_Inst_Rgt();
        break;
      case 4:
        movement_Inst_Bwd();
        break;
      case 5:
        movement_Inst_Stp();
        break;
     }
     delay((EEPROM.read(i+1))* 1000);     // multiplied by thousand because the original time was divided by 1000 while storing in EEPROM.///delaying between two sequence
```

```
    }
  }
void go_Forward_Seq(int j)
{
  //go in forward direction sequence
  movement_Inst_Fwd();//
  delay(total_Fwd_Time[j]);
}
void go_Left_Seq(int k)
{
  //go in Left direction sequence
  movement_Inst_Lft();
  delay(total_Lft_Time[k]);
}
void go_Right_Seq(int l)
{
  //go in right direction sequence
  movement_Inst_Rgt();
  delay(total_Rgt_Time[l]);
}
void go_Backward_Seq(int m)
{
  //go in backward direction sequence
  movement_Inst_Bwd();
  delay(total_Bwd_Time[m]);
}
void go_Stop_Seq(int n)
```

```
{
  //go in Stop sequence
  movement_Inst_Stp();
  delay(total_Stp_Time[n]);
}
void movement_Inst_Fwd(void)
{
  Serial.println("Going_Forward");
  // MOTOR_A CLOCKWISE MAX SPEED
  digitalWrite(in1, HIGH);
  digitalWrite(in2, LOW);
  analogWrite(ena,127);
  // MOTOR_B CLOCKWISE MAX SPEED
  digitalWrite(in3, LOW);
  digitalWrite(in4, HIGH);
  analogWrite(enb, 127);
}
void movement_Inst_Lft(void)
{
  Serial.println("Going_Left");
  // LEFTSIDE
  // MOTOR_A CLOCKWISE MAX SPEED
  digitalWrite(in1, HIGH);
  digitalWrite(in2, LOW);
  analogWrite(ena, 255);
  // MOTOR_B CLOCKWISE MAX SPEED
  digitalWrite(in3, LOW);
```

```
  digitalWrite(in4, LOW);
  analogWrite(enb, 255);                    //1735==1.735s
  delay(1735);
  digitalWrite(in1, LOW);
  digitalWrite(in2, LOW);
  digitalWrite(in3, LOW);
  digitalWrite(in4, LOW);
}
void movement_Inst_Rgt(void)
{
  Serial.println("Going_Right");
  // RIGHTSIDE
  // MOTOR_A CLOCKWISE MAX SPEED
  digitalWrite(in1, LOW);
  digitalWrite(in2, LOW);
  analogWrite(ena, 255);
  // MOTOR_B CLOCKWISE MAX SPEED
  digitalWrite(in3, LOW);
  digitalWrite(in4, HIGH);
  analogWrite(enb, 255);                        //1735
  delay(1735);
  digitalWrite(in1, LOW);
  digitalWrite(in2, LOW);
  digitalWrite(in3, LOW);
  digitalWrite(in4, LOW);
}
void movement_Inst_Bwd(void)
```

```
{
  Serial.println("Going_Backward");
  // REVERSE
  // MOTOR_A COUNTERCLOCKWISE MAX SPEED
  digitalWrite(in1, LOW);
  digitalWrite(in2, HIGH);
  analogWrite(ena, 127);
  // MOTOR_B COUNTERCLOCKWISE MAX SPEED
  digitalWrite(in3, HIGH);
  digitalWrite(in4, LOW);
  analogWrite(enb, 127);
}
void movement_Inst_Stp(void)
{
  Serial.println("Stopping");
  // Stop both motors immediately
  digitalWrite(in1, LOW);
  digitalWrite(in2, LOW);
  digitalWrite(in3, LOW);
  digitalWrite(in4, LOW);
}
```