PROJECT REPORT ON

SMART AGRICULTURAL BOT

SUBMITTED BY:

- P.HEMANTH KUMAR
- S.ASHISH
- A.RAMYA
- VARSHITHA
- CHARAN
- T.KOUSHIK SAI
- M.EESHA
- SHRUTHI
- AKSHAYA

THE ROBOTICS CLUB

Integrating Knowledge...

Mr. Rohit Chand R N

Mentor

Robotics Club - SNIST

Ms. Bhavishya

Mentor

Robotics Club - SNIST

Certification

This is to certify that the group project titled **”SMART AGRICULTURAL BOT”** has been successfully completed and submitted by the following team members **P.HEMANTH KUMAR, T.KOUSHIK SAI, S.ASHISH, M.EESHA, A.RAMYA, SHRUTHI, VARSHITHA, AKSHAYA, CHARAN**. The project was carried out as a part of our engineering, under the supervision of **Mr. Rohit Chand R N** and **Ms. Bhavishya.** The team has shown dedication and teamwork in conducting research, analysis, and development of the project.

Mr. Rohit Chand R N
Mentor
Robotics Club - SNIST

Ms. Bhavishya
Mentor
Robotics Club - SNIST

Acknowledgement

We, the members of the project group, would like to express our profound gratitude and appreciation to all those who have contributed to the successful completion of this project titled **”SMART AGRICULTURAL BOT”.**

First and foremost, we extend our deepest thanks to our project supervisor, **Mr. Hemanth Kumar** and **Mr. Rohit Chand R N**, for their continuous guidance, support, and constructive feedback throughout the course of this project. Their expertise and insights have been valuable in helping us understand the complexities of the subject and guiding us through various stages of the project.

We would like to acknowledge the contributions of **Mr. Rohit Chand R N** who provided critical advice, data, or technical support during the project. We would also like to thank our peers for their constructive criticism and valuable suggestions.

Team Members:

- P.HEMANTH KUMAR
- S.ASHISH
- A.RAMYA
- VARSHITHA
- CHARAN
- T.KOUSHIK SAI
- M.EESHA
- SHRUTHI
- AKSHAYA

Declaration

We, the undersigned, hereby declare that the project report entitled **”SMART AGRICULTURAL BOT”** submitted, is a result of our original work conducted under the supervision of **Mr. Rohit Chand R N** and **Ms. Bhavishya** We further declare that,This report has not been submitted to any other institution or organization for any other degree, diploma or certification. All work presented here is our own, except where specific references have been made to the work of others. We have fully acknowledged all the sources, literature, and data used in the completion of this project. This project was completed in accordance with the ethical and academic standards prescribed by our supervisior and the coordinator. We understand that any violation of this declaration may result in disciplinary action as per the guidelines of the our supervisor and coordinator.

Team Members:

- P.HEMANTH KUMAR
- S.ASHISH
- A.RAMYA
- VARSHITHA
- CHARAN
- T.KOUSHIK SAI
- M.EESHA
- SHRUTHI
- AKSHAYA

Contents

List of Figures

ABSTRACT

THE PROBLEM

Agricultural monitoring in small-scale and indoor farming environments often relies on manual labor or stationary systems that are inefficient, require regular maintenance, and lack mobility. Power limitations, human dependency, and the inability to remotely monitor environmental conditions in real-time are major challenges that reduce productivity and increase operational costs.

TEAMS APPROACH TO SOLVE THE PROBLEM

To address the growing challenges in agricultural automation, this project proposes the Smart Agricultural Charging Line-Following Bot, an autonomous robotic platform that combines mobility, real-time sensing, wireless charging, and automation for smart farming. The bot navigates through agricultural fields or indoor environments using a line-following algorithm powered by a five-channel infrared sensor, thereby eliminating the need for GPS and allowing precise path tracking across predefined routes. A key innovation of this system is its on-the-go inductive charging mechanism, which enables the bot to be wirelessly powered while in motion. This ensures uninterrupted, maintenance-free operation, significantly increasing operational time and reliability. Additionally, a linear actuator is integrated into the system — triggered when the bot reaches a specific location. Upon activation, the actuator lowers a probe into the soil, allowing the soil moisture sensor to collect real-time data. After measurement, the actuator retracts automatically, optimizing sensor placement without manual intervention.

The bot is equipped with an ESP32-CAM module mounted on a pan-tilt servo mechanism, offering live video streaming through a local web server for remote field surveillance. Real-time environmental data — including soil moisture, temperature, humidity (via DHT11), gas presence (via MQ-4), and light intensity (LDR) — is transmitted and visualized on the Blynk IoT platform, making the system accessible through any smartphone or connected device. This project showcases a powerful integration of embedded systems, wireless charging, IoT connectivity, and automated actuation, making it a compact and scalable solution for smart agriculture. It not only reduces manual labor and enhances environmental monitoring but also supports sustainable, data-driven farming practices, pushing the boundaries of precision agriculture and rural automation.

Chapter 1

INTRODUCTION

1.1 INTRODUCTION

Agriculture remains one of the most crucial sectors in the global economy, particularly in countries where a significant portion of the population depends on farming for their livelihood. Despite the advances in technology, many agricultural practices, especially in small and medium-sized farms, still rely on manual labor and conventional methods that are time-consuming, inefficient, and susceptible to human error. As the demand for food continues to rise with the growing population, there is an urgent need to adopt smart and sustainable solutions that can optimize agricultural operations while reducing labor dependency and energy consumption.

One of the major challenges in modern precision farming is the need for constant field monitoring and environmental sensing to ensure optimal crop health and resource usage. Traditional monitoring techniques are limited by the lack of mobility, continuous power supply, and real-time communication. Furthermore, manual methods of data collection and observation often lead to delayed responses and inaccurate readings. In areas with limited access to power sources or in vast greenhouses, it becomes even more challenging to maintain continuous surveillance or automated data collection.

To address these issues, this project introduces a Smart Agricultural Charging Line Following Bot, a compact and intelligent robotic system capable of autonomously navigating through predefined agricultural paths using a line-following algorithm. Unlike conventional bots, this system integrates an on-the-go inductive charging system, which ensures uninterrupted operation by wirelessly charging the bot's battery as it moves along the line. This feature eliminates the downtime typically associated with battery charging and significantly extends the system's field usability without manual intervention. At the heart of the bot are two microcontrollers: the Arduino UNO for line-following control and sensor interfacing, and the ESP32-CAM for video streaming and IoT communication. The ESP32-CAM is mounted on a pan-tilt servo mechanism, allowing it to provide a wide field of view for real-time video monitoring. The live footage is streamed to a local web server, allowing remote users to observe field conditions via Wi-Fi-connected devices.

In addition to video surveillance, the bot is equipped with a suite of environmental sensors, including a DHT11 sensor for measuring temperature and humidity, an MQ-4 gas sensor for detecting harmful gases, an LDR (Light Dependent Resistor) for measuring light intensity, and a soil moisture sensor to monitor the water content in the soil. These sensors continuously collect data, which is then transmitted to the Blynk IoT platform, where users can visualize and monitor real-time values through a mobile app interface.

This project showcases a seamless blend of robotics, IoT, wireless power transfer, and embedded systems to solve a real-world agricultural problem. By automating both monitoring and mobility, and by integrating remote sensing with live video and charging independence, this system offers a cost-effective, scalable, and intelligent solution that can be deployed in agricultural fields, greenhouses, or even industrial domains for similar applications. In the coming sections, the report will detail the design, development, and deployment of the system, including hardware schematics, software architecture, testing outcomes, and potential future enhancements.

1.2 PURPOSE OF THE PROJECT

The main function of the Smart Agricultural Charging Line Following Bot is to provide a low-cost, scalable, and autonomous system that makes field monitoring in agriculture and controlled setups like greenhouses or polyhouses simpler and more efficient. As the population continues to grow and demand more food and there is also a growing lack of skilled labor in the countryside, a change towards smart and automated farming methods is critically needed. Conventional manual farm practices are no longer adequate to serve the increased needs for productivity, accuracy, and sustainability. One of the most serious challenges encountered in agriculture is the absence of real-time monitoring of crop health and environmental factors such as soil moisture content, temperature, humidity, gas concentration, and light intensity. The lack of monitoring these critical parameters in real-time and in a uniform manner normally results in lesser crop output, excessive utilization of resources such as water and fertilizers, and even total crop loss under severe conditions. Even though there are some fixed IoT solutions available in the market, they do not possess mobility, flexibility, and round-the-clock operation because of battery constraints or need for manual recharging.

This project seeks to fill that gap by creating a mobile robotic system capable of continuous field monitoring and eliminating the disruption of manual charging. Incorporating an inductive wireless charging system enables the bot to recharge its battery while moving, such that the monitoring is never disrupted. This facilitates longer autonomous working, which is essential where vast or inaccessible grounds have to be monitored. The bot's ability to follow a fixed path using line sensors ensures precise navigation without requiring GPS or complex mapping algorithms, which are often unreliable in closed or rural environments. Also, the project offers the option of visually observing the environment through an onboard ESP32-CAM module with pan-tilt capabilities. The live video feed can be viewed through a local web server, providing users with a real-time visual output of the surroundings of the bot. This is especially valuable when detecting intrusions, crop abnormalities, or possible dangers without physically being in the field.

Another central objective of this project is to aggregate all sensor readings using the Blynk IoT platform so that farmers or users can remotely access important environmental information like soil water content levels, ambient temperature and humidity levels, gas leaks (e.g., methane from compost), and light intensity from any global location. This type of remote monitoring enables data-driven irrigation, fertilization, and pest control decisions that result in intelligent farming practices.

Technically, this project also seeks to illustrate the coupling of embedded systems, IoT platforms, and renewable energy principles such as wireless charging into one affordable robotic package. The system is modular and upgradeable, with applications in education, research, and small industrial use.

1.3 EXISTING SYSTEMS

With the passage of time, agriculture has incorporated various new technologies for increasing productivity and efficiency. Some of these technologies include sensor-based Internet of Things (IoT) systems, line-following robots, solar-powered monitoring stations, and surveillance bots. Each of these systems, however, works with some limitations, rendering them ineffective as end-to-end solutions for autonomous field monitoring, real-time communication, and constant operation. One of the most prevalent technologies employed in smart agriculture nowadays is the fixed sensor-based IoT system. Such systems place fixed sensors in the field to track environmental parameters like soil moisture, temperature, humidity, and light intensity. The data thus gathered is frequently sent wirelessly to a cloud platform such as Blynk or ThingSpeak for easy real-time access. Though these systems are precise and low-power, they suffer from a main disadvantage of being static. A single sensor node can sense over a small area, so many units are needed to cover an entire field. This adds greatly to cost and upkeep. Also, these systems are usually powered by batteries or solar panels, which themselves are prone to erratic performance, especially in shaded or indoor settings.

Conversely, simple line-following robots are programmed to trace black or white lines on a predetermined route with the aid of infrared sensors. They have limited use in school settings or for minimal automation purposes such as carrying items within warehouses. While these robots provide rudimentary mobility, they lack any sensors for observing environmental circumstances. Their manual charging dependency also restricts their running time, and they do not have any real-time communication features like IoT or video streaming.

Another type of available systems are surveillance robots that employ microcontrollers like the ESP32-CAM or Raspberry Pi to offer live video streaming over Wi-Fi. These systems are usually deployed indoors for security or lab observation purposes. They can be remotely accessed by web servers or applications, providing a measure of control and flexibility. But they are still hampered by their battery life since they need to be manually recharged regularly. In addition, these systems do not provide any other environmental sensing capabilities such as soil moisture or gas sense, so they are also not appropriate for field use in agriculture. In certain agricultural environments, solar-powered monitoring systems are used to solve the problem of short battery life. Although this system has a renewable source of power, it is unsuitable for indoor farms and areas with limited sunlight. Also, solar panels contribute to the weight, expense, and maintenance needs of the system, yet they still fail to meet the mobility requirement across a wide area. On the upper end of the scale are commercial farming drones and autonomous ground robots that use AI and GPS for precision agriculture. But they are too expensive, hard to operate, and require maintenance, making them unavailable for small-scale farmers or for localized use such as greenhouse monitoring.

1.4 LIMITATIONS OF EXISTING SYSTEMS

Even though there have been significant developments in intelligent agriculture and automation, the currently deployed systems in field-level agricultural settings are fraught with several constraints that limit their performance, scalability, and repeated usage. These constraints stem from issues like limited mobility, power reliance, premium costs, local sensing ability, and the lack of holistic platforms that enable frequent, autonomous operation.

One of the most common constraints pertains to static sensor-based IoT systems. These systems rely on discrete nodes that are installed in fixed locations within the field to gather environmental information such as soil moisture, temperature, and humidity. Although they are good at gathering localized data, they lack field coverage unless several sensors are deployed over extensive areas. This not only raises the overall cost but also adds complexity to maintenance. Further, such stationary systems rely on batteries or solar power, both of which have limitations — batteries require routine replacement or recharging, while solar-powered systems do not work in low-light environments like greenhouses or during cloudy days.

Another popular solution is line-following robots, which tend to be implemented for simple path-following functions and are extensively used in schooling or small automation tasks. While these robots provide rudimentary mobility, they are deprived of essential features like environmental perception, autonomous data communication, and power self-sustenance. Their need for constant manual charging renders them inadequate for sustained deployment in actual agricultural environments. In addition, the absence of communication interfaces within the robots prevents them from sending data remotely, hence necessitating the use of physical proximity for monitoring and data acquisition.

Surveillance robots with modules such as the ESP32-CAM provide real-time video streaming in Wi-Fi and are applied in different indoor applications. Though they offer some monitoring, their main purpose is only video capture and doesn't involve integrations with other sensors. Additionally, the systems are usually run based on battery power, restricting their usage time. The necessity for regular manual recharging interrupts uninterrupted use and dilutes their efficiency in 24/7 monitoring jobs.

Solar IoT devices, although a boon in diminishing the reliance on manual recharging, have their own limitations. They do not function in shaded or indoor conditions, and changes in the availability of sunlight can cause inconsistent functionality. The added expense and area needed for the installation of solar panels also render such solutions infeasible for small and portable systems. Additionally, such systems are typically fixed position, which limits their capacity to dynamically monitor multiple spaces within a field or a facility. On the upper side, commercial farm drones and autonomous farm robots provide sophisticated features such as GPS navigation, aerial monitoring, and robotic spraying. These systems are too costly for small or medium-scale farmers to afford and need specialized training to operate and maintain. They rely on scheduled recharging and, being highly complex, pose software reliability and long-term use issues.

1.5 PROPOSED SYSTEM

To address shortcomings witnessed in current agricultural monitoring and automation systems, we suggest the design and development of a ”Smart Agricultural Charging Line Following Bot”. The system seeks to integrate mobility, real-time environmental monitoring, autonomous charging, and remote data access into an integrated platform. The system envisaged here is a multi-purpose robot system that moves around pre-determined paths by itself, gathers current sensor data, streams video for visual surveillance, and is wirelessly charged through inductive power transfer during operations. The fundamental concept of the system is derived from a line-following robot that travels along a predetermined path that is created by drawing a black line on a white surface. This eliminates the reliability or availability issue of GPS in indoor settings such as greenhouses. The robot utilizes a 5-channel infrared sensor array for precise line detection and navigation. The mobility enables the system to cover greater dynamic areas with fewer sensor units than stationary monitoring installations, thus enhancing efficiency and cost-effectiveness.

One important aspect of the presented system is its on-the-go inductive recharging ability. In contrast to conventional robots, which must come to a halt and physically be attached to a charging dock, the novel bot has an inductive coil receiver module that charges the onboard battery continuously as the robot drives along a wireless power transmission line. This eliminates the need for periodic recharging by humans. It greatly improves the uptime and independence of the robot, which is perfect for long-term deployment in agriculture.

In environmental awareness, the robot carries several sensors such as a DHT11 sensor for temperature and humidity, an MQ4 gas sensor to identify methane and other combustible gases, a soil moisture sensor to check water levels in the soil, and an LDR (Light Dependent Resistor) to sense ambient light. These sensors continuously gather data, which is processed and transmitted in real-time by means of the ESP32 microcontroller. The data is uploaded to the Blynk IoT platform, on which readings can be visualized through a mobile application. Farmers or facility managers can then monitor environmental conditions remotely and act quickly on abnormal readings or field condition changes. An addition of particular importance in this system is the addition of an ESP32-CAM module mounted on a pan-tilt servo mechanism. This enables live streaming of the environment around the robot via a local web server, which can be accessed from a smartphone or laptop that has the same Wi-Fi connection as the robot. Remote observation of the bot's motion and environment in real time offers significant improvements in security and monitoring efficiency, particularly for enclosed or valuable farming areas.

The entire control and coordination of the robot are controlled by two microcontrollers: one Arduino UNO that controls IR sensor input and motor control for the line-following mechanism, and one ESP32 that controls the sensor data acquisition, video streaming, and IoT communication. The system runs effectively with the help of a rechargeable battery that is stored by the inductive charging module.

1.6 ADVANTAGES OF PROPOSED SYSTEM

The proposed system, titled **"Smart Agricultural Charging Line Following Bot"**, offers several significant advantages over traditional and existing systems in the field of precision agriculture and autonomous monitoring. By integrating mobility, environmental sensing, real-time monitoring, and continuous wireless charging into a single robotic platform, this system effectively overcomes the key limitations of current technologies while offering new benefits in terms of automation, cost-effectiveness, and sustainability.

One of the most notable advantages of the proposed system is its **autonomous mobility** through line-following capability. Unlike stationary sensor nodes or semi-mobile bots that rely on manual navigation, this bot uses a five-channel infrared sensor to follow predefined black-line paths. This allows it to traverse through agricultural fields, greenhouses, or indoor farms without human intervention or the need for GPS. As a result, the bot can consistently monitor multiple zones of a field, ensuring broader and more dynamic coverage.

Another key innovation is the incorporation of an **on-the-go inductive wireless charging system**. This feature eliminates the need for manual charging, which is a common limitation in battery-operated monitoring systems. The bot receives power wirelessly as it moves over a charging track, ensuring continuous operation without interruptions. This not only reduces the need for maintenance but also extends the system's operational uptime, making it highly suitable for long-term deployments in remote or difficult-to-access areas. The system also integrates an **ESP32-CAM module with pan-tilt mechanism**, allowing real-time video monitoring through a local web server. This enables farmers or users to visually inspect field conditions remotely, enhancing the level of situational awareness and decision-making. The ability to stream live footage from the bot's surroundings in real time is particularly beneficial for detecting pests, intrusions, or anomalies that may not be captured by sensors alone.

Furthermore, the proposed system features a comprehensive array of **environmental sensors**, including a DHT11 (for temperature and humidity), MQ-4 gas sensor, soil moisture sensor, and LDR (for light intensity). These sensors provide critical insights into field conditions, enabling precise environmental monitoring. The sensor data is uploaded to the Blynk IoT platform, allowing users to monitor real-time values from their smartphones or other internet-connected devices. This remote monitoring capability is a major step forward in enabling smart, data-driven agriculture.

In addition to its technical advantages, the system is designed to be **cost-effective and modular**. It utilizes commonly available components such as Arduino UNO, ESP32-CAM, and standard sensor modules, making it accessible to students, researchers, and small-scale farmers. Its modular design also allows easy upgrades, modifications, or replacement of individual components without overhauling the entire system. Another advantage is its **energy efficiency**. Since the bot is powered via wireless inductive charging, there is minimal energy waste, and it eliminates downtime associated with battery depletion. Moreover, the system is optimized for indoor and semi-indoor environments such as greenhouses, where solar power may be ineffective. The integration of multiple features into one platform mobility, self-charging, environmental sensing, live video, and IoT connectivity—sets the proposed bot apart from conventional systems that typically focus on just one or two functionalities. This makes the proposed solution highly versatile and adaptable for multiple applications beyond agriculture, including warehouse automation, surveillance, environmental research, and smart city infrastructure.

1.7 ADVANTAGES OF PROPOSED SYSTEM

The proposed system, titled **"Smart Agricultural Charging Line Following Bot"**, offers several significant advantages over traditional and existing systems in the field of precision agriculture and autonomous monitoring. By integrating mobility, environmental sensing, real-time monitoring, and continuous wireless charging into a single robotic platform, this system effectively overcomes the key limitations of current technologies while offering new benefits in terms of automation, cost-effectiveness, and sustainability.

One of the most notable advantages of the proposed system is its **autonomous mobility** through line-following capability. Unlike stationary sensor nodes or semi-mobile bots that rely on manual navigation, this bot uses a five-channel infrared sensor to follow predefined black-line paths. This allows it to traverse through agricultural fields, greenhouses, or indoor farms without human intervention or the need for GPS. As a result, the bot can consistently monitor multiple zones of a field, ensuring broader and more dynamic coverage.

Another key innovation is the incorporation of an **on-the-go inductive wireless charging system**. This feature eliminates the need for manual charging, which is a common limitation in battery-operated monitoring systems. The bot receives power wirelessly as it moves over a charging track, ensuring continuous operation without interruptions. This not only reduces the need for maintenance but also extends the system's operational uptime, making it highly suitable for long-term deployments in remote or difficult-to-access areas.

The system also integrates an **ESP32-CAM module with pan-tilt mechanism**, allowing real-time video monitoring through a local web server. This enables farmers or users to visually inspect field conditions remotely, enhancing the level of situational awareness and decision-making. The ability to stream live footage from the bot's surroundings in real time is particularly beneficial for detecting pests, intrusions, or anomalies that may not be captured by sensors alone.

Furthermore, the proposed system features a comprehensive array of **environmental sensors**, including a DHT11 (for temperature and humidity), MQ-4 gas sensor, soil moisture sensor, and LDR (for light intensity). These sensors provide critical insights into field conditions, enabling precise environmental monitoring. The sensor data is uploaded to the Blynk IoT platform, allowing users to monitor real-time values from their smartphones or other internet-connected devices. This remote monitoring capability is a major step forward in enabling smart, data-driven agriculture. In addition to its technical advantages, the system is designed to be **cost-effective and modular**. It utilizes commonly available components such as Arduino UNO, ESP32-CAM, and standard sensor modules, making it accessible to students, researchers, and small-scale farmers. Its modular design also allows easy upgrades, modifications, or replacement of individual components without overhauling the entire system.

Another advantage is its **energy efficiency**. Since the bot is powered via wireless inductive charging, there is minimal energy waste, and it eliminates downtime associated with battery depletion. Moreover, the system is optimized for indoor and semi-indoor environments such as greenhouses, where solar power may be ineffective.

The integration of multiple features into one platform—mobility, self-charging, environmental sensing, live video, and IoT connectivity sets the proposed bot apart from conventional systems that typically focus on just one or two functionalities. This makes the proposed solution highly versatile and adaptable for multiple applications beyond agriculture, including warehouse automation, surveillance, environmental research, and smart city infrastructure.

1.8 HARDWARE REQUIREMENTS

1.8.1 5 Chamber IR sensor

The 5-chamber IR sensor module is a crucial component in line-following robotic systems, designed to detect contrasting surface patterns such as black lines on white backgrounds. It comprises five infrared emitter-receiver pairs arranged in a single row, allowing it to scan a wider surface area compared to single-channel sensors. Each chamber functions by emitting infrared light towards the ground and detecting the amount of reflected light. On white or reflective surfaces, the IR light is bounced back to the receiver, resulting in a HIGH logic output, whereas black or non-reflective surfaces absorb the light, leading to a LOW output. This differential response enables the sensor to distinguish the exact position of the line in relation to the robot's body. By analyzing which sensors are detecting black and which are detecting white, the robot's microcontroller can adjust motor speeds to maintain alignment, take sharp turns, or correct its path. The center sensor is usually calibrated to follow the middle of the line, while the side sensors help the robot identify when it veers off course. This 5-sensor configuration offers better accuracy, faster response, and more intelligent navigation decisions compared to simpler IR setups. Additionally, the sensor outputs can be interfaced directly with digital I/O pins of microcontrollers like Arduino UNO, and require minimal external components, making them ideal for embedded systems in autonomous navigation, maze solving, and smart agricultural bots like the one in this project.

Figure 1.1: 5 Chamber IR sensor

1.8.2 Arduino UNO

The Arduino Uno is an open-source microcontroller board based on the ATmega328P chip. It has 14 digital I/O pins (6 of which can be used as PWM outputs) and 6 analog input pins. The board operates at 5V and runs at a clock speed of 16 MHz, powered via USB or an external adapter. It is widely used for beginner to advanced-level embedded and robotics projects due to its simplicity and large community support. Programming the Arduino Uno is done using the Arduino IDE, which uses a simplified version of C/C++.

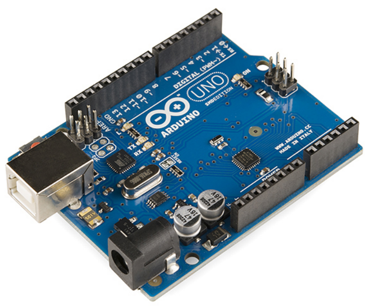

Figure 1.2: Arduino UNO

1.8.3 ESP 32

The ESP32 is a powerful, low-cost microcontroller with integrated Wi-Fi and Bluetooth, widely used in IoT and embedded systems. It features a dual-core processor, multiple GPIOs, ADC/DAC support, and is ideal for wireless data communication and real-time applications. Its compact design and built-in networking capabilities make it suitable for tasks like remote monitoring, sensor interfacing, and camera streaming.

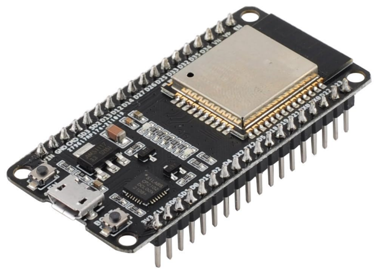

Figure 1.3: ESP 32

1.8.4 L298N

The L298N is a dual H-bridge motor driver module used to control the direction and speed of two DC motors or a single stepper motor. It operates at voltages from 5V to 35V and can deliver up to 2A current per channel. The module allows for forward/reverse motion and speed control using PWM signals, making it ideal for robotics and automation projects.

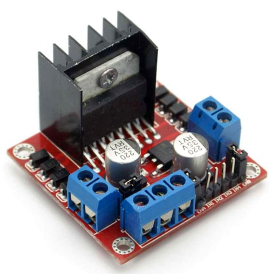

Figure 1.4: L298N

1.8.5 DC Motors

DC motors are electromechanical devices that convert direct current electrical energy into rotational mechanical motion. They are commonly used in robotics to drive wheels due to their simple control and high torque at low speeds. Speed and direction of rotation can be easily controlled using motor drivers and PWM signals from microcontrollers.

Figure 1.5: DC Motors

1.8.6 ESP 32 CAM

The ESP32-CAM is a low-cost microcontroller module with built-in Wi-Fi, Bluetooth, and an integrated OV2640 camera for real-time image and video capture. It supports streaming video over a local web server, making it ideal for surveillance and IoT applications. Its compact design and GPIO availability also allow integration with external devices like servos, sensors, and SD cards.

Figure 1.6: ESP 32 CAM

1.8.7 Pan-Tilt Servo

The Pan-Tilt Servo mechanism allows two degrees of rotational movement—horizontal (pan) and vertical (tilt)—enabling a mounted camera or sensor to adjust its viewing angle dynamically. It typically uses two servo motors, one for each axis, providing precise control over direction. In robotics, it's widely used for surveillance and object tracking applications, such as real-time monitoring with an ESP32-CAM.

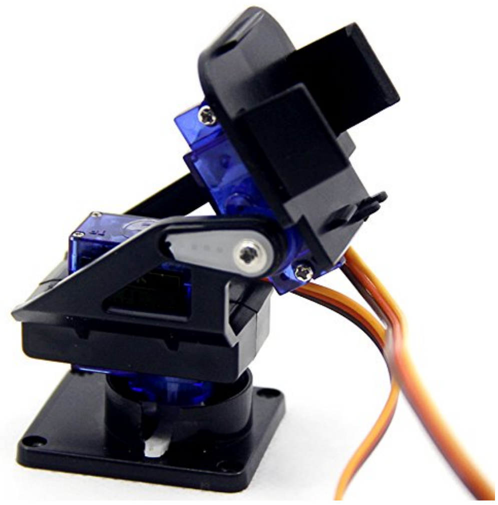

Figure 1.7: Pan-Tilt Servo

1.8.8 LDR Sensor

The LDR (Light Dependent Resistor) sensor is a passive component whose resistance decreases with increasing light intensity. It is commonly used to detect ambient light levels in applications like automatic lighting and daylight monitoring. In this project, the LDR helps assess environmental brightness, which can be logged or used for decision-making in the bot's operation.

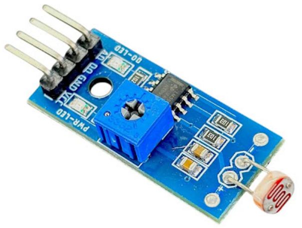

Figure 1.8: LDR Sensor

1.8.9 MQ-4 Sensor

The MQ-4 gas sensor is used to detect flammable gases like methane (CH), LPG, and natural gas in the environment. It operates by measuring changes in resistance of its sensing material when exposed to target gases. The sensor provides an analog output proportional to gas concentration, making it useful for air quality monitoring and gas leak detection.

Figure 1.9: MQ-4 Sensor

1.8.10 DHT 11 Sensor

The DHT11 sensor is a digital temperature and humidity sensor that provides calibrated output through a single-wire interface. It can measure temperature from 0°C to 50°C and relative humidity from 20 to 90 percentage with decent accuracy. The sensor is low-cost, easy to interface with microcontrollers like Arduino or ESP32, and suitable for environmental monitoring applications.

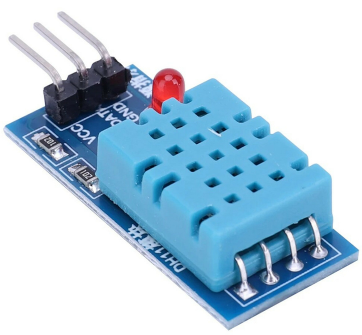

Figure 1.10: DHT 11 Sensor

1.8.11 Soil Moisture Sensor

The Soil Moisture Sensor is used to measure the water content in the soil by detecting its conductivity or resistance. It typically consists of two exposed probes that act as electrodes — the wetter the soil, the higher the conductivity between them. This sensor helps determine whether the soil needs irrigation, making it ideal for smart farming and automated watering systems.

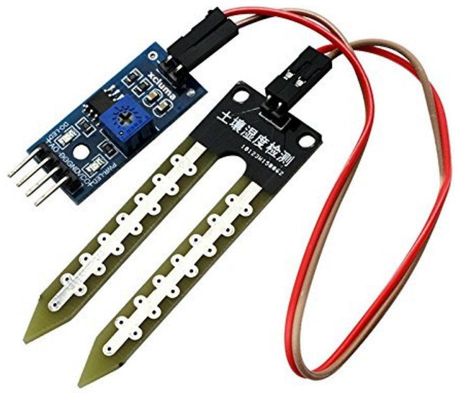

Figure 1.11: Soil Moisture Sensor

1.8.12 Bread Board

A breadboard is a reusable platform used for prototyping and testing electronic circuits without soldering. It allows components to be easily inserted and interconnected using jumper wires. Breadboards are ideal for experimenting with circuit designs before creating a permanent PCB.

Figure 1.12: Bread Board

1.8.13 Jumper Wires

Jumper wires are flexible, insulated conductors used to make temporary or permanent connections between components on a breadboard or between modules and microcontrollers. They come in three types are male-to-male, male-to-female, and female-to-female to suit different pin configurations. Jumper wires simplify prototyping by enabling quick and reliable connections without soldering.

Figure 1.13: Jumper Wires

1.8.14 Li-ion Batteries

Lithium-ion (Li-ion) batteries are rechargeable energy storage devices known for their high energy density, lightweight, and long cycle life. They are commonly used in portable electronics and robotics due to their compact size and efficient power delivery. In this project, they provide a reliable power source for continuous operation of sensors, motors, and microcontrollers.

Figure 1.14: Li-ion Batteries

1.8.15 Transmitter Coil

The transmitter coil is a key component in wireless power transfer systems, responsible for generating an alternating magnetic field. When powered by an AC signal or high-frequency driver circuit, it induces current in a nearby receiver coil through electromagnetic induction. It plays a vital role in enabling contactless charging in systems like the on-the-go inductive charging bot.

Figure 1.15: Transmitter Coil

1.8.16 Receiver Coil

The receiver coil is a crucial component in wireless power transfer systems that captures the magnetic field generated by the transmitter coil. It converts this magnetic energy into electrical energy through electromagnetic induction, which is then used to charge the battery or power the circuit. In mobile systems like robots, the receiver coil enables on-the-go inductive charging without physical contact.

Figure 1.16: Receiver Coil

1.8.17 MT3608

The MT3608 is a high-efficiency step-up (boost) DC-DC converter module that can increase an input voltage as low as 2V up to 28V. It is commonly used to power devices requiring higher voltages from low-voltage battery sources. The module features adjustable output, high switching frequency, and compact size, making it ideal for portable and embedded applications.

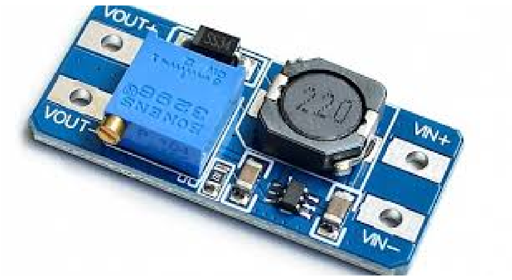

Figure 1.17: MT3608

1.8.18 TP4056

The TP4056 is a linear lithium-ion battery charger module designed for single-cell batteries, typically used with 3.7V Li-ion or Li-Po cells. It features constant current/constant voltage (CC/CV) charging and includes built-in overcharge, over-discharge, and short-circuit protection. The module is compact, easy to integrate, and commonly powered via a micro-USB or Type-C port.

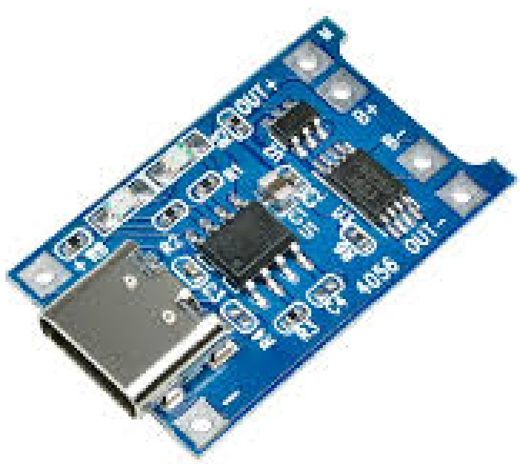

Figure 1.18: TP4056

1.8.19 Linear Actuator

A linear actuator is a device that converts rotational motion into straight-line motion, typically using electric, pneumatic, or hydraulic power. It is used to push, pull, lift, or position loads with precision. Commonly found in automation, robotics, and industrial systems, linear actuators are valued for their accuracy and controllability. Electric linear actuators often use a motor, lead screw, and nut assembly to generate motion.

Figure 1.19: TP4056

1.9 SOFTWARE REQUIREMENTS

1.9.1 Arduino IDE

The Arduino IDE (Integrated Development Environment) is an open-source platform used to write, compile, and upload code to Arduino and compatible microcontroller boards. It supports C/C++ programming and provides a simple interface for beginners and professionals. The IDE includes a code editor, message area, text console, and a toolbar with functions for uploading and verifying code. It also comes with a library manager and a vast collection of example sketches to help in rapid prototyping.

Figure 1.20: ARDUINO IDE

1.9.2 Fusion 360

Fusion 360 is a cloud-based 3D CAD, CAM, and CAE software developed by Autodesk. It allows users to design, simulate, and manufacture products all in one platform. Ideal for engineers, product designers, and hobbyists, Fusion 360 supports parametric modeling, assemblies, rendering, and animation. It also includes tools for CNC machining and 3D printing, making it a powerful tool for end-to-end product development.

Figure 1.21: FUSION 360

1.9.3 Cirkit Designer

Circuit Diagram is an intuitive, browser-based and desktop application used to design and simulate electronic circuits. It allows users to create circuit schematics using drag-and-drop components, annotate designs, and share or export them easily. It's especially useful for students, hobbyists, and educators who need a clean and quick way to visualize circuits. The platform supports breadboard layouts, PCB planning, and basic simulation features for prototyping.

Figure 1.22: Cirkit Designer

1.9.4 Blynk IOT

Blynk IoT is a cloud-based platform that allows users to build and monitor IoT projects with ease using smartphones or web dashboards. It supports microcontrollers like ESP32, Arduino, and NodeMCU. Blynk enables real-time data visualization using widgets like gauges, graphs, and buttons. Users can remotely control devices and receive alerts through the app. The platform uses a Blynk Auth Token to link hardware with the cloud. It also supports automation, scheduling, and integration with other smart services. Blynk is ideal for projects involving sensors, actuators, and real-time control applications like your smart agricultural bot.

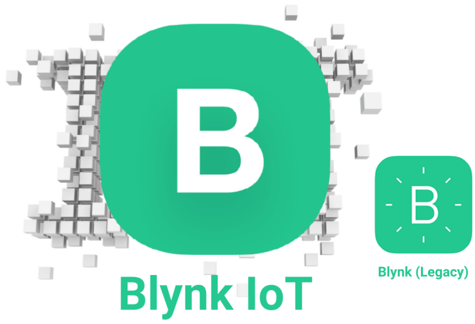

Figure 1.23: Blynk IOT

1.10 BLOCK DIAGRAM

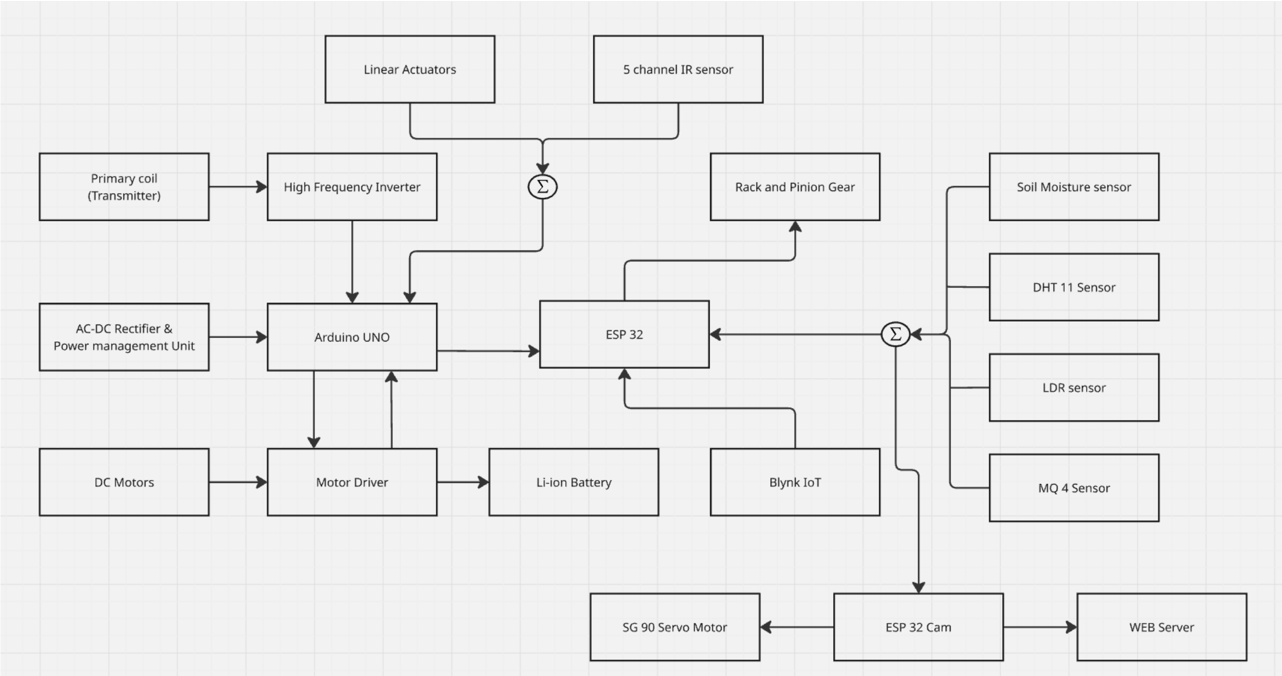

Figure 1.24: Block Diagram

The system block diagram of the proposed ”Smart Agricultural Charging Line Following Bot” represents a compact, integrated solution that combines autonomous navigation, real-time environmental monitoring, IoT connectivity, and wireless charging. Each module in the system plays a distinct role and collectively ensures the robot functions continuously and intelligently in agricultural or indoor farming environments. The process begins with the primary coil (transmitter) embedded in the ground path, which transmits energy wirelessly through electromagnetic induction. This energy is picked up by a receiver coil on the bot and fed into the high-frequency inverter, which stabilizes and converts the incoming AC signal for further use. The output is then processed by the AC-DC rectifier and power management unit, which converts it into usable DC voltage to power the components and charge the onboard Li-ion battery. This self-charging mechanism enables the bot to run continuously without manual intervention, even while in motion.

The central control of navigation is handled by an Arduino UNO, which is responsible for interpreting sensor data from the 5-channel IR sensor. This sensor is mounted at the front of the bot and detects line patterns on the ground, allowing the bot to follow a designated path autonomously. The IR sensor sends real-time feedback to the Arduino UNO, which in turn commands the motor driver. The motor driver controls the DC motors based on the sensor inputs to adjust the bot's movement, ensuring smooth navigation along the black line path. For enhanced intelligence and connectivity, the system includes an ESP32 module, which communicates with the Arduino UNO and handles all IoT-related functions. It receives environmental data from multiple sensors, including the DHT11 sensor for temperature and humidity, the soil moisture sensor, LDR sensor for light intensity, and the MQ-4 gas sensor for detecting harmful gases like methane. These values are collected by the ESP32 and uploaded to the Blynk IoT platform, allowing users to monitor real-time field conditions from a mobile or web interface. This remote access provides significant advantages in terms of decision-making and reduces the need for constant physical monitoring of the robot.

The system also includes a rack and pinion gear mechanism controlled by the ESP32, which is responsible for vertical motion or adjustments in camera or sensor orientation. To enhance field visibility and security, an ESP32-CAM module is incorporated and mounted on a pan-tilt controlled SG90 servo motor. This camera module captures real-time video footage of the bot's surroundings and streams it over a local web server, accessible through any device connected to the same network. This allows the user to visually inspect crop conditions, obstructions, or anomalies that are not detected by sensors alone. Power for all components is supplied from the Li-ion battery, which is continuously charged by the wireless charging unit. The battery provides stable output through the power management system to the Arduino UNO, ESP32, motors, sensors, and camera module, ensuring uninterrupted operation.

The integration of all these modules — line-following, wireless charging, environmental sensing, servo-controlled vision system, and IoT connectivity — results in a robust and self-sustaining platform for smart agriculture. This modular, scalable, and autonomous bot can navigate greenhouses or fields, collect environmental data, provide live video feedback, and recharge on-the-go — offering significant advantages over existing manual or semi-automated systems. Moreover, the use of low-cost components such as the Arduino UNO and ESP32 ensures that the system remains economically feasible for small and medium-scale farmers. The modular design also allows for easy upgrades, sensor expansion, or replacement, making the system flexible and future-proof. The integration with the Blynk IoT platform enhances accessibility by allowing users to monitor and control the bot remotely through a smartphone or web dashboard. Additionally, real-time alerts and data logs improve traceability and support data-driven agricultural decisions. Overall, the system effectively bridges the gap between automation, IoT, and sustainability in modern precision farming.

1.11 CIRCUIT DIAGRAM

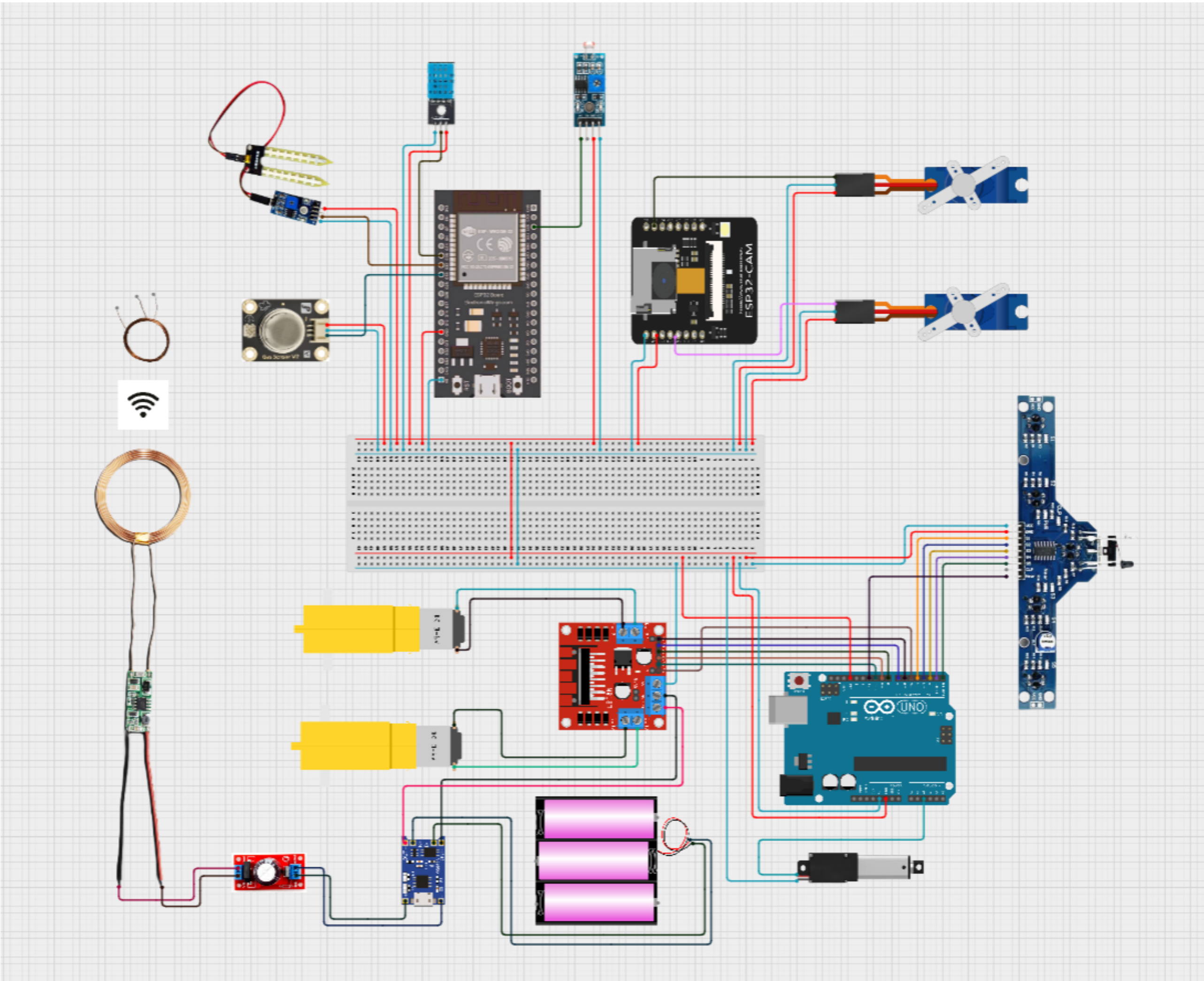

Figure 1.25: Circuit Diagram

The circuit diagram of the Smart Agricultural Charging Line Following Bot illustrates the integration of multiple embedded modules and sensors into a cohesive, autonomous system. Each component in the circuit plays a vital role, enabling navigation, environmental sensing, real-time video streaming, and wireless charging for uninterrupted performance. The design is implemented using a combination of the ESP32, ESP32-CAM, and Arduino UNO, along with various sensors, motor drivers, and power modules. At the heart of the environmental sensing and IoT control lies the ESP32 development board, which is responsible for reading sensor data, sending it to the cloud via Wi-Fi, and controlling auxiliary components such as servos and the camera interface. Connected to the ESP32 are three essential sensors: the DHT11 sensor for temperature and humidity, the MQ4 gas sensor for detecting methane or LPG leakage in the agricultural environment, and the capacitive soil moisture sensor for determining water content in the soil. All three sensors use digital or analog I/O pins on the ESP32, and the data collected is displayed in real-time on the Blynk IoT platform, allowing remote monitoring from smartphones or computers. The ESP32-CAM module is also integrated into the circuit and is connected to a pan-tilt SG90 servo motor setup, allowing the camera to rotate and tilt for live visual coverage of the bot's surroundings. This real-time video is hosted on a local web server, accessible through the ESP32-CAM's IP address. The servos are connected to PWM-enabled pins on the ESP32, allowing directional control based on user input or pre-programmed logic. This feature helps in remote surveillance and crop inspection, offering valuable visual insights alongside sensor data.

Navigation of the bot is achieved using two DC geared motors connected to an L298N motor driver module, which receives movement commands from the Arduino UNO. The UNO is programmed to respond to inputs from a 5-channel IR sensor array, which detects black lines on the ground. Based on the sensor's output, the Arduino adjusts motor speeds and directions via the L298N, ensuring smooth path-following behavior. The IR sensor is powered through the UNO, and its five output channels are connected to five digital input pins for accurate tracking and steering control. For power supply, the system uses a 3-cell Li-ion battery pack, which provides energy to the motors and electronic modules. A TP4056 battery charging module with protection is connected to manage safe charging and discharging of the battery. Additionally, a DC-DC buck converter module (LM2596) is used to step down the battery voltage to a stable 5V, which is suitable for powering the ESP32, ESP32-CAM, and Arduino UNO. The system is also equipped with an inductive wireless charging receiver coil, which connects to the TP4056 via a small power receiver module. This enables the bot to wirelessly charge while moving over a defined path where the primary transmitter coil is embedded. The entire circuit is built on a large breadboard for prototyping and testing purposes. Power rails from the Li-ion battery pack are carefully distributed across the breadboard to power various modules. Ground connections are shared using a common rail to maintain electrical consistency. Data lines are neatly routed to avoid interference, especially for I2C and serial communication lines between the ESP32 and sensors.

1.12 CAD Model

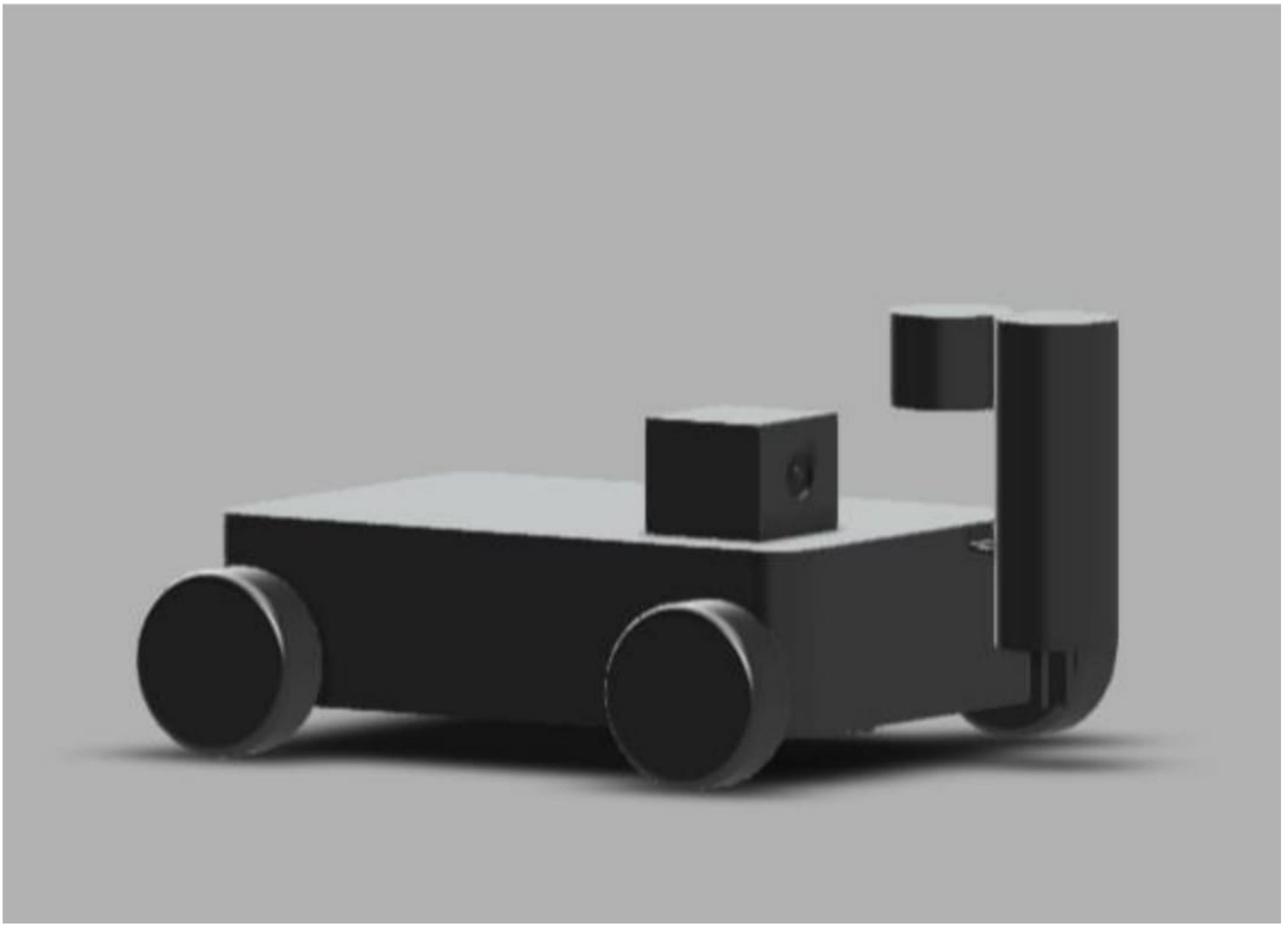

Figure 1.26: CAD Model

The 3D model shown represents the mechanical and structural design of the proposed Smart Agricultural Charging Line Following Bot. This prototype is engineered with simplicity and functionality in mind, tailored specifically for applications in greenhouse environments or semi-automated indoor farms. The design prioritizes compactness, modularity, and ease of maintenance, which are essential attributes for systems that must operate autonomously for extended periods in potentially humid or dusty agricultural environments. The body of the bot is modeled with a rectangular chassis, serving as the main structural frame where all electronic and mechanical components are mounted. The chassis is dimensioned proportionally to house the Li-ion battery pack, motor driver, Arduino UNO, ESP32, and sensor modules. It also accommodates wiring paths and access points for future modifications. The flat surface provides a sturdy base for mounting sensors, the ESP32-CAM module, and the servo-based camera movement mechanism, while ensuring a low center of gravity for stability during motion. The bot features four circular wheels, with the rear wheels typically powered by DC geared motors for forward and turning motion. The front wheels are free-rotating, allowing smoother navigation and turning without excessive mechanical friction. The motor arrangement supports differential drive control, which is ideal for line-following applications. The wheel size is chosen to ensure sufficient ground clearance and traction for smooth movement over flat surfaces, especially on line-guided tracks placed in a greenhouse or indoor farming zone.

On top of the chassis, the most visually prominent feature is the ESP32-CAM module, mounted centrally inside a square camera enclosure. This module is strategically placed for optimal field-of-view and is used for real-time video monitoring of the bot's surroundings. Next to the camera, a vertical mount holds a pan-tilt servo mechanism, allowing the camera to rotate horizontally and vertically. This mechanism is essential for dynamic surveillance, enabling the bot to adjust its view based on programmed instructions or user commands via the web interface. Toward the front or bottom of the chassis (not visible from the render), the IR sensor array is positioned underneath the bot for line detection. This 5-channel IR sensor is aligned perpendicularly to the bot's path and tracks dark lines on the floor, which guide the bot's navigation. Since the IR sensors are sensitive to surface contrast, their placement close to the ground ensures higher accuracy and minimal noise due to ambient light. Onboard sensors such as the DHT11 (temperature/humidity), LDR (light intensity), Soil Moisture Sensor, and MQ-4 (gas detection) are conceptually included within the internal space of the bot chassis or on small externally mounted brackets. These sensors collect real-time environmental data as the bot moves along its route. The data is processed by the ESP32 and transmitted via Wi-Fi to the Blynk IoT dashboard, making the bot a mobile sensing unit in addition to being a surveillance and navigation platform. The bot's power system, including a 3-cell Li-ion battery pack and a TP4056 charging module, is concealed inside the chassis. What sets this design apart is the integration of a wireless charging receiver coil, which allows the bot to recharge wirelessly as it passes over an embedded transmitter coil in the ground path. This ensures continuous operation without needing human intervention, a key innovation for automation in agriculture. Overall, the 3D design reflects a well-thought-out, functional prototype with all critical components accurately positioned for maximum operational efficiency. The clean, minimal aesthetic also makes the design suitable for future enhancements like a waterproof enclosure or mounting of robotic arms or sprinklers. It not only supports autonomous movement but also serves as a mobile data collector and live-streaming bot, optimized for real-world agricultural deployment.

1.13 RESULT

The Smart Agricultural Charging Line Following Bot was successfully developed, integrated, and tested in a controlled environment to evaluate its core functionalities. The system met all primary objectives, including autonomous navigation, environmental monitoring, wireless charging, and real-time video streaming. The line-following mechanism, based on a 5-channel IR sensor and Arduino UNO, demonstrated consistent and accurate tracking over black tape lines on various surfaces, even under fluctuating lighting conditions. The bot was able to make smooth turns and recover from minor path deviations, indicating the reliability of the motor control logic and IR sensor alignment.

The wireless charging system, built using a transmitter and receiver coil with a TP4056 battery management circuit, enabled efficient on-the-go recharging of the Li-ion battery. During testing, the battery voltage levels remained within safe operational limits, ensuring uninterrupted operation of the microcontrollers, motors, and sensors. This functionality validates the feasibility of integrating wireless power transfer in dynamic mobile systems, eliminating the need for manual charging and enhancing autonomy.

Environmental data from the DHT11 (temperature/humidity), soil moisture sensor, MQ-4 gas sensor, and LDR sensor was successfully collected by the ESP32 module and transmitted in real-time to the Blynk IoT platform. Users could monitor live sensor values on their smartphones, with minimal latency and stable Wi-Fi connectivity. This not only verified the ESP32's networking capability but also demonstrated practical use of IoT in agricultural condition monitoring.

Additionally, the ESP32-CAM module mounted on a pan-tilt servo mechanism provided real-time video footage of the bot's surroundings through a web interface. The camera stream was stable, with responsive servo controls allowing dynamic adjustment of the camera angle. This visual feature proved helpful for remote surveillance and enhanced situational awareness in environments where human access is limited.

In summary, the bot's performance confirms that the system is functionally complete and aligns well with the goals of smart farming. The prototype operated seamlessly during continuous trials, proving the robustness of both its electronic and mechanical subsystems. The results highlight its potential for deployment in greenhouse monitoring, automated crop inspection, and as a scalable unit in smart agriculture ecosystems.

Chapter 2

CONCLUSION AND FUTURE SCOPE

2.1 CONCLUSION

The Smart Agricultural Charging Line Following Bot stands as a testament to how embedded systems, IoT, automation, and sustainable energy can be integrated to address practical problems in agriculture. This project was conceptualized with the vision of improving the efficiency and sustainability of small- to medium-scale farming, particularly in greenhouse or indoor environments. Throughout the development process, each module was carefully selected and integrated to achieve a fully autonomous mobile unit capable of navigating along predefined paths, collecting essential environmental data, recharging wirelessly, and providing real-time visual feedback through the internet. The system architecture is built around two main microcontrollers — the Arduino UNO and ESP32 — which work in tandem to handle navigation and wireless communication respectively. The use of a 5-channel IR sensor ensures accurate line-following capability, enabling the robot to move autonomously along a path marked on the floor. By employing DC motors and a motor driver, the system achieves smooth locomotion and responsive control based on sensor feedback. This level of autonomy significantly reduces the need for human supervision in structured agricultural setups, such as rows of crops or greenhouse lanes.

A major advancement presented in this project is the integration of on-the-go wireless charging using inductive power transfer. The robot features a wireless charging receiver coil and a TP4056 module that together allow seamless recharging while moving over a predefined route. This feature eliminates the problem of frequent manual recharging, making the bot more practical for long-term deployments. In agricultural fields where uninterrupted monitoring is crucial, this technology can offer huge operational advantages and reduce downtime. Another key functionality of the bot is environmental monitoring, made possible through the integration of sensors such as the DHT11 for temperature and humidity, Soil Moisture Sensor, LDR sensor for light intensity, and the MQ-4 gas sensor for detecting harmful gases. These sensors feed real-time data to the ESP32, which uploads it to the Blynk IoT platform. This capability allows farmers or researchers to track critical parameters remotely through their smartphones or web interfaces. Such remote monitoring reduces physical visits and enables more informed decisions, contributing to precision agriculture practices.

The addition of the ESP32-CAM with a pan-tilt mechanism enhances the bot's usability by providing live video streaming of its surroundings. This real-time visual feedback is valuable for identifying issues such as pest infestations, crop damage, or physical obstructions on the path. Furthermore, the pan-tilt setup gives flexibility in camera orientation, making it possible to survey a wider area with fewer blind spots. In terms of design, the prototype is compact and modular, which not only ensures portability but also simplifies maintenance and upgrades. The chassis layout accommodates all components neatly, and the overall structure is stable for operation on various surfaces. The 3D model reflects a functional yet simple design that can easily be adapted or expanded based on specific use-case needs.

Through this project, we successfully demonstrated how embedded technology can be applied in innovative ways to develop a smart, self-sufficient agricultural bot. The synergy of autonomous mobility, wireless charging, IoT data acquisition, and visual monitoring paves the way for scalable and sustainable agricultural automation solutions. While the prototype focuses on core functionalities, the design leaves room for future upgrades such as robotic arms for sowing or harvesting, water-sprinkling systems, or integration with AI-based decision-making models.

2.2 FUTURE SCOPE

The proposed Smart Agricultural Charging Line Following Bot introduces a unique blend of automation, real-time sensing, and wireless charging for efficient field monitoring in controlled agricultural environments. While the current prototype successfully demonstrates basic mobility, environmental data acquisition, and wireless power reception, its future potential remains vast.

By integrating more advanced features and scalability options, the system can evolve into a comprehensive solution for large-scale, precision farming and agricultural robotics. One of the most promising extensions of this system lies in upgrading its navigation intelligence. Currently, the bot relies on a line-following mechanism guided by an IR sensor array. In future iterations, this can be enhanced by incorporating AI-based vision tracking or GPS modules for outdoor field deployment, enabling it to navigate dynamically and adaptively even in unstructured terrains. Advanced path planning algorithms and obstacle avoidance using ultrasonic or LiDAR sensors could further improve autonomy and mobility across varied agricultural landscapes. Another area of development is the integration of machine learning for crop and soil analytics. The existing ESP32-CAM setup can be upgraded with onboard image processing or edge AI capabilities to detect plant diseases, pest infestations, or nutrient deficiencies from visual patterns. This would allow the bot not only to monitor but also to assess crop health, providing actionable insights in real-time. Combined with environmental sensor data, this AI integration could revolutionize how farmers diagnose and treat agricultural issues proactively. The power system can also be scaled up. While the prototype uses a wireless charging module and a basic battery management system (BMS), future models can feature solar charging panels, more efficient MPPT controllers, and long-range wireless energy harvesting for improved sustainability. This would allow the bot to function entirely off-grid, reducing operational costs and dependency on scheduled manual charging. The IoT integration also has room for significant enhancement. Moving forward, integration with platforms beyond Blynk—such as Google Firebase, AWS IoT Core, or Azure IoT Hub—can offer advanced analytics, cloud storage, and cross-device communication. Additionally, incorporating LoRa or NB-IoT modules would allow long-range communication in rural areas where Wi-Fi infrastructure is unavailable. From a mechanical standpoint, the current four-wheel base can be upgraded with adaptive terrain wheels, robotic arms, or modular attachments like pesticide sprayers, seed dispensers, or water delivery units. These expansions would transform the bot from a monitoring tool into a multifunctional autonomous farming assistant capable of performing multiple agricultural operations with minimal human intervention. Moreover, by applying swarm robotics principles, multiple bots of this kind can be deployed across larger fields. They can operate collaboratively under a central monitoring unit, sharing data and tasks for optimized coverage and efficiency.

Chapter 3

REFERENCES

1. S. Patil, R. Raut, and M. Shaikh, "Smart Agriculture System Using IoT," *International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)*, vol. 5, no. 6, pp. 11060–11066, 2017.

2. A. Kumar and P. Singh, "Development of Line Follower Robot Using Arduino and IR Sensors," *International Journal of Scientific & Engineering Research*, vol. 9, no. 4, pp. 1238–1241, 2018.

3. S. Jain and A. Sharma, "Design and Implementation of Wireless Charging System for Electric Vehicles Using Inductive Coupling," *IEEE International Conference on Power Electronics*, pp. 1–5, 2020.

4. H. Li, X. Wang, and D. Wang, "Smart Agriculture Based on Cloud Computing and IoT," *Journal of Sensors*, vol. 2020, Article ID 7593485, 2020.

5. M. S. Shinde, "IoT Based Smart Farming System," *International Journal of Engineering Research and Technology (IJERT)*, vol. 8, no. 7, pp. 1–4, 2019.

6. A. Bansal, K. Gaur, and V. Yadav, "Wireless Power Transmission for Electric Vehicles Using Inductive Coupling," *IJSREM*, vol. 4, no. 3, pp. 56–62, 2020.

7. S. Srivastava, "Design of a Line Following Robot Using Arduino," *International Journal of Scientific and Technology Research*, vol. 8, no. 6, pp. 62–65, 2019.

8. N. Verma and R. Joshi, "ESP32 and Its Applications in IoT," *International Journal of Advanced Research in Computer Science*, vol. 10, no. 3, pp. 47–50, 2019.

9. M. E. Haque and A. Reza, “Blynk Based IoT Smart Agriculture Monitoring System,” *IEEE Region 10 Conference (TENCON)*, pp. 1234–1239, 2021.

10. T. Singh, R. Raj, and N. Gupta, “Real Time Monitoring of Agricultural Environment using IoT,” *International Research Journal of Engineering and Technology (IRJET)*, vol. 6, no. 4, pp. 754–757, 2019.

11. D. Sharma, “Implementation of ESP32-CAM Based Surveillance Robot,” *Journal of Embedded Systems*, vol. 7, no. 2, pp. 45–49, 2020.

12. R. Gupta and M. S. Patil, “IoT Based Soil Moisture Monitoring System Using ESP32,” *International Journal of Engineering Research and Technology (IJERT)*, vol. 9, no. 8, pp. 172–176, 2020.

13. M. Hossain, “Design and Fabrication of an Autonomous Robot for Smart Agriculture,” *IEEE International Conference on Electrical, Computer and Communication Engineering (ECCE)*, pp. 1–4, 2021.

14. K. Singh and A. Gupta, “Energy Efficient Wireless Charging System for Robotics,” *International Journal of Renewable Energy Research*, vol. 11, no. 2, pp. 784–790, 2021.

15. R. Kumar and A. Kumar, “Smart Irrigation System Using IoT and Cloud,” *International Journal of Scientific Research in Engineering and Management (IJSREM)*, vol. 6, no. 5, pp. 28–34, 2022.

Chapter 4

SOURCE CODE

4.1 Movement Code

```
// IR Sensor Pins
#define IR_L2 A0
#define IR_L1 A1
#define IR_C  A2
#define IR_R1 A3
#define IR_R2 A4
// Motor Driver Pins
#define IN1 8
#define IN2 9
#define IN3 10
#define IN4 11
// Linear Actuator Pins
#define ACTUATOR_EXTEND 6
#define ACTUATOR_RETRACT 7
bool actuatorTriggered = false; // To prevent repeated triggers
void setup() {
  // Set IR pins as input
  pinMode(IR_L2, INPUT);
```

```
  pinMode(IR_L1, INPUT);
  pinMode(IR_C, INPUT);
  pinMode(IR_R1, INPUT);
  pinMode(IR_R2, INPUT);
  // Set motor pins as output
  pinMode(IN1, OUTPUT);
  pinMode(IN2, OUTPUT);
  pinMode(IN3, OUTPUT);
  pinMode(IN4, OUTPUT);
  // Set actuator pins as output
  pinMode(ACTUATOR_EXTEND, OUTPUT);
  pinMode(ACTUATOR_RETRACT, OUTPUT);
  // Initially stop actuator
  digitalWrite(ACTUATOR_EXTEND, LOW);
  digitalWrite(ACTUATOR_RETRACT, LOW);
  Serial.begin(9600);
}
void loop() {
  int l2 = digitalRead(IR_L2);
  int l1 = digitalRead(IR_L1);
  int c  = digitalRead(IR_C);
  int r1 = digitalRead(IR_R1);
  int r2 = digitalRead(IR_R2);
  Serial.print("L2: "); Serial.print(l2);
  Serial.print(" L1: "); Serial.print(l1);
  Serial.print(" C: ");  Serial.print(c);
  Serial.print(" R1: "); Serial.print(r1);
  Serial.print(" R2: "); Serial.println(r2);
  // === Movement logic ===
  if ((l2 == LOW) && (l1 == LOW) && (c == HIGH) && (r1 == LOW) && (r2 == LOW)) {
```

```
    moveForward();
  } else if ((l1 == HIGH) || (l2 == HIGH)) {
    turnLeft();
  } else if ((r1 == HIGH) || (r2 == HIGH)) {
    turnRight();
  } else {
    stopMotors();
  }
  // === Actuator trigger condition ===
  if (c == HIGH && l1 == LOW && r1 == LOW && !actuatorTriggered) {
    activateActuator();
    actuatorTriggered = true;
  }
  // Reset actuator trigger (optional, e.g., after passing over black line)
  if (c == LOW) {
    actuatorTriggered = false;
  }
}
void moveForward() {
  digitalWrite(IN1, HIGH);
  digitalWrite(IN2, LOW);
  digitalWrite(IN3, HIGH);
  digitalWrite(IN4, LOW);
}
void turnLeft() {
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, HIGH);
  digitalWrite(IN3, HIGH);
  digitalWrite(IN4, LOW);
}
```

```
void turnRight() {
  digitalWrite(IN1, HIGH);
  digitalWrite(IN2, LOW);
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, HIGH);
}
void stopMotors() {
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, LOW);
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, LOW);
}
// === Actuator Function ===
void activateActuator() {
  Serial.println("Extending actuator...");
  digitalWrite(ACTUATOR_EXTEND, HIGH);
  digitalWrite(ACTUATOR_RETRACT, LOW);
  delay(2000); // Adjust based on actuator speed
  Serial.println("Retracting actuator...");
  digitalWrite(ACTUATOR_EXTEND, LOW);
  digitalWrite(ACTUATOR_RETRACT, HIGH);
  delay(2000); // Retract delay
  // Stop actuator
  digitalWrite(ACTUATOR_EXTEND, LOW);
  digitalWrite(ACTUATOR_RETRACT, LOW);
}
```

4.2 Sensors Integration

```
#define BLYNK_TEMPLATE_ID "TMPL3pu2PTbx1"
#define BLYNK_TEMPLATE_NAME "Agriculture Bot"
#define BLYNK_AUTH_TOKEN "pHZdB5iOwslMoCPHeXpHU4guBY_5M55N"
#include <WiFi.h>
#include <BlynkSimpleEsp32.h>
#include <DHT.h>
#define DHTPIN 4        // DHT11 connected to GPIO 4 (D4)
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);
// Sensor Pins
#define SOIL_PIN 36     // GPIO 36 - VP
#define LDR_PIN 39      // GPIO 39 - VN
#define MQ4_PIN 34      // GPIO 34
// WiFi credentials
char ssid[] = "YourWiFiSSID";
char pass[] = "YourWiFiPassword";
// Blynk Timer
BlynkTimer timer;
void sendSensorData() {
  float h = dht.readHumidity();
  float t = dht.readTemperature();
  int soilValue = analogRead(SOIL_PIN); // 0{4095
  int ldrValue = analogRead(LDR_PIN);
  int gasValue = analogRead(MQ4_PIN);
  int soilPercent = map(soilValue, 4095, 1500, 0, 100);
  int lightPercent = map(ldrValue, 4095, 0, 0, 100);
  int gasPercent = map(gasValue, 0, 4095, 0, 100);
  Serial.println("Sending sensor data to Blynk...");
```

```
  Serial.print("Temp: "); Serial.println(t);
  Serial.print("Humidity: "); Serial.println(h);
  Serial.print("Soil: "); Serial.println(soilPercent);
  Serial.print("Light: "); Serial.println(lightPercent);
  Serial.print("Gas: "); Serial.println(gasPercent);
  // Send to Blynk
  Blynk.virtualWrite(V0, t);
  Blynk.virtualWrite(V1, h);
  Blynk.virtualWrite(V2, soilPercent);
  Blynk.virtualWrite(V3, lightPercent);
  Blynk.virtualWrite(V4, gasPercent);
}
void setup() {
  Serial.begin(115200);
  dht.begin();
  Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass);
  timer.setInterval(2000L, sendSensorData);  // Every 2 sec
}
void loop() {
  Blynk.run();
  timer.run();
}
```

4.3 Camera Code

```
#include <WiFi.h>
#include <WebServer.h>
#include <ESP32Servo.h>
#include "esp_camera.h"
// Replace with your network credentials
```

```
const char* ssid = "YourWiFiSSID";
const char* password = "YourWiFiPassword";
// Define GPIOs for servos
#define PAN_SERVO_PIN 14
#define TILT_SERVO_PIN 15
Servo panServo;
Servo tiltServo;
int panAngle = 90;  // Initial position
int tiltAngle = 90;
WebServer server(80);
// ESP32-CAM (AI-Thinker) camera pin definition
#define PWDN_GPIO_NUM    -1
#define RESET_GPIO_NUM   -1
#define XCLK_GPIO_NUM    0
#define SIOD_GPIO_NUM    26
#define SIOC_GPIO_NUM    27
#define Y9_GPIO_NUM      35
#define Y8_GPIO_NUM      34
#define Y7_GPIO_NUM      39
#define Y6_GPIO_NUM      36
#define Y5_GPIO_NUM      21
#define Y4_GPIO_NUM      19
#define Y3_GPIO_NUM      18
#define Y2_GPIO_NUM      5
#define VSYNC_GPIO_NUM   25
#define HREF_GPIO_NUM    23
#define PCLK_GPIO_NUM    22
void startCameraServer();
void handleRoot() {
  String html = "<!DOCTYPE html><html><head><title>
```

```
  ESP32-CAM Pan-Tilt</title></head><body>";
  html += "<h2>ESP32-CAM Pan-Tilt Control</h2>";
  html += "<img src=\"/stream\" style=\"width: 320px;
  height: 240px; display: block;
  margin-bottom: 20px;
  \"><br>";
  html += "<button onclick=\"location.href='/left'\">Left</button> ";
  html += "<button onclick=\"location.href='/right'\">Right</button><br><br>";
  html += "<button onclick=\"location.href='/up'\">Up</button> ";
  html += "<button onclick=\"location.href='/down'\">Down</button><br>";
  html += "</body></html>";
  server.send(200, "text/html", html);
}
void handleLeft() {
  panAngle = constrain(panAngle - 10, 0, 180);
  panServo.write(panAngle);
  server.sendHeader("Location", "/");
  server.send(303);
}
void handleRight() {
  panAngle = constrain(panAngle + 10, 0, 180);
  panServo.write(panAngle);
  server.sendHeader("Location", "/");
  server.send(303);
}
void handleUp() {
  tiltAngle = constrain(tiltAngle - 10, 0, 180);
  tiltServo.write(tiltAngle);
  server.sendHeader("Location", "/");
  server.send(303);
```

```
}
void handleDown() {
  tiltAngle = constrain(tiltAngle + 10, 0, 180);
  tiltServo.write(tiltAngle);
  server.sendHeader("Location", "/");
  server.send(303);
}
void setup() {
  Serial.begin(115200);
  // Attach servos
  panServo.attach(PAN_SERVO_PIN);
  tiltServo.attach(TILT_SERVO_PIN);
  panServo.write(panAngle);
  tiltServo.write(tiltAngle);
  // Connect to Wi-Fi
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.print("Connected to WiFi. IP address: ");
  Serial.println(WiFi.localIP());
  // Camera configuration
  camera_config_t config;
  config.ledc_channel = LEDC_CHANNEL_0;
  config.ledc_timer   = LEDC_TIMER_0;
  config.pin_d0       = Y2_GPIO_NUM;
  config.pin_d1       = Y3_GPIO_NUM;
  config.pin_d2       = Y4_GPIO_NUM;
```

```
config.pin_d3       = Y5_GPIO_NUM;
config.pin_d4       = Y6_GPIO_NUM;
config.pin_d5       = Y7_GPIO_NUM;
config.pin_d6       = Y8_GPIO_NUM;
config.pin_d7       = Y9_GPIO_NUM;
config.pin_xclk     = XCLK_GPIO_NUM;
config.pin_pclk     = PCLK_GPIO_NUM;
config.pin_vsync    = VSYNC_GPIO_NUM;
config.pin_href     = HREF_GPIO_NUM;
config.pin_sscb_sda = SIOD_GPIO_NUM;
config.pin_sscb_scl = SIOC_GPIO_NUM;
config.pin_pwdn     = PWDN_GPIO_NUM;
config.pin_reset    = RESET_GPIO_NUM;
config.xclk_freq_hz = 20000000;
config.pixel_format = PIXFORMAT_JPEG;
if(psramFound()) {
  config.frame_size = FRAMESIZE_QVGA;
  config.jpeg_quality = 10;
  config.fb_count = 2;
} else {
  config.frame_size = FRAMESIZE_QQVGA;
  config.jpeg_quality = 12;
  config.fb_count = 1;
}
// Initialize camera
esp_err_t err = esp_camera_init(&config);
if (err != ESP_OK) {
  Serial.printf("Camera init failed with error 0x%x", err);
  return;
}
```

```
  // Setup Web Server
  server.on("/", handleRoot);
  server.on("/left", handleLeft);
  server.on("/right", handleRight);
  server.on("/up", handleUp);
  server.on("/down", handleDown);
  // MJPEG stream
  server.on("/stream", HTTP_GET, []() {
    WiFiClient client = server.client();
    String response = "HTTP/1.1 200 OK\r\n";
    response += "Content-Type: multipart/x-mixed-replace; boundary=frame\r\n\r\n";
    server.sendContent(response);
    while (1) {
      camera_fb_t * fb = esp_camera_fb_get();
      if (!fb) continue;
      response = "--frame\r\n";
      response += "Content-Type: image/jpeg\r\n\r\n";
      server.sendContent(response);
      server.sendContent((const char*)fb->buf, fb->len);
      server.sendContent("\r\n");
      esp_camera_fb_return(fb);
      if (!client.connected()) break;
    }
  });
  server.begin();
  Serial.println("Web server started!");
}
void loop() {
  server.handleClient();
}
```