Ideation Document:
Application Based Robotics: Solutions for sustainability

Team of Indian Institute of Technology, Gandhinagar (IITGN)

Mentor:

Madhu Vadali, Associate Professor, Mechanical Engineering
(madhu.vadali@iitgn.ac.in)

Team Members:

1) Harshada Kale, 3rd-year B.Tech, Mechanical Engineering Major
(harshada.kale@iitgn.ac.in)

2) Dhruv Bhavsar, 1st-year B.Tech, Electrical Engineering Major
(bhavsar.dhruv@iitgn.ac.in)

3) Archit Sinha, 1st-year B.Tech, Integrated Circuit Design and Technology Major
(archit.sinha@iitgn.ac.in)

4) Ansh Pendharkar, 1st-year B.Tech, Chemical Engineering Major
(ansh.pendharkar@iitgn.ac.in)

Robofest 5.0

Contents

0.1 Abstract

The IIT Gandhinagar team presents a project for Robofest 5.0. This project aims to design and develop an autonomous agricultural rover to support sustainable farming practices. The rover is capable of moving across uneven farmland terrain and following defined paths for navigation. It distinguishes between weeds and crops in real time using image processing algorithms. Based on this classification, the rover sprays pesticides selectively, helping to reduce unnecessary agrochemical use. The system combines computer vision, navigation, and mechanical design to lower pesticide consumption, improve crop yield, and promote environmentally conscious farming methods. Future iterations of this project plan to integrate a weed removal system using image processing and robotic arms, with the goal of making precision farming more accessible and scalable.

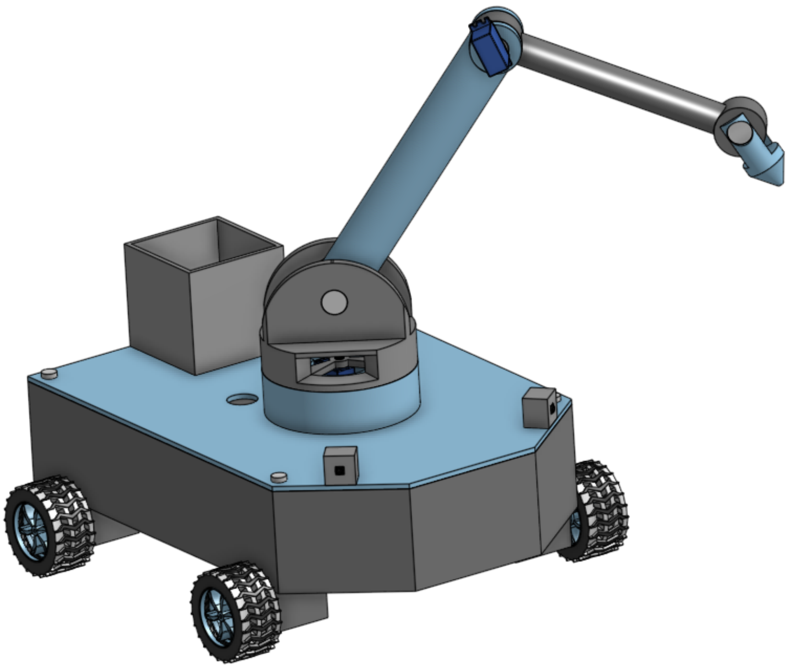


Figure 1: Proposed design

0.2 Mechanical Design

0.2.1 3 DoF Robotic Arm

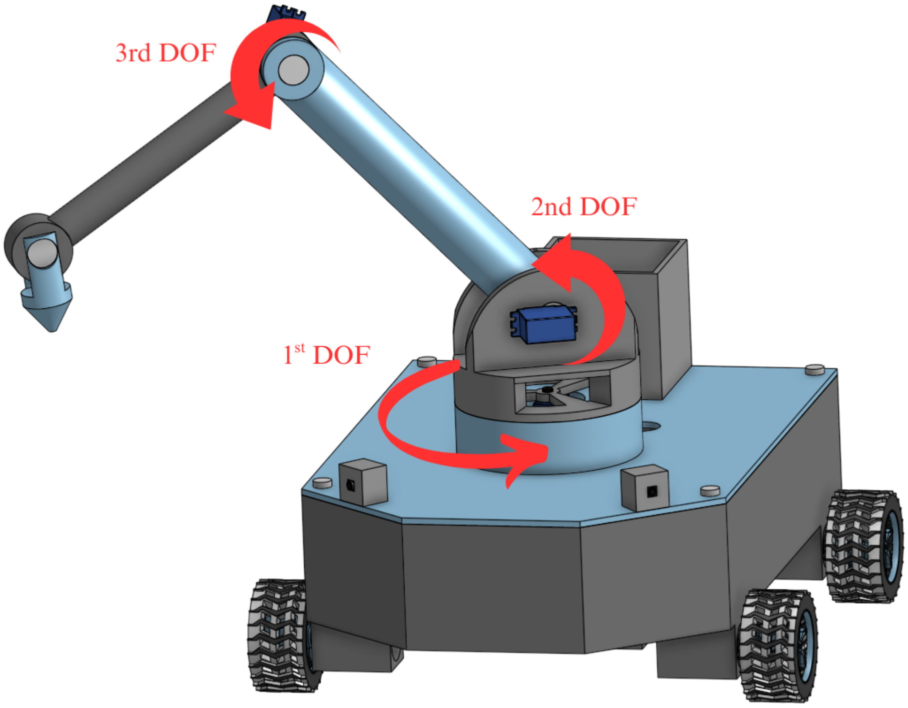


Figure 2: Robotic arm

- A robotic arm with 3 rotational degrees of freedom that will allow the robot to do multitude of tasks on field.
- The end of the robotic arm has a bore to attach the nozzle that will allow the robot to spray herbicide, pesticide or water depending on the requirement.
- The robotic arm allows the precise navigation to the detected crop/weed and helps the robot complete the task.

0.2.2 Mobility and Navigation Mechanism

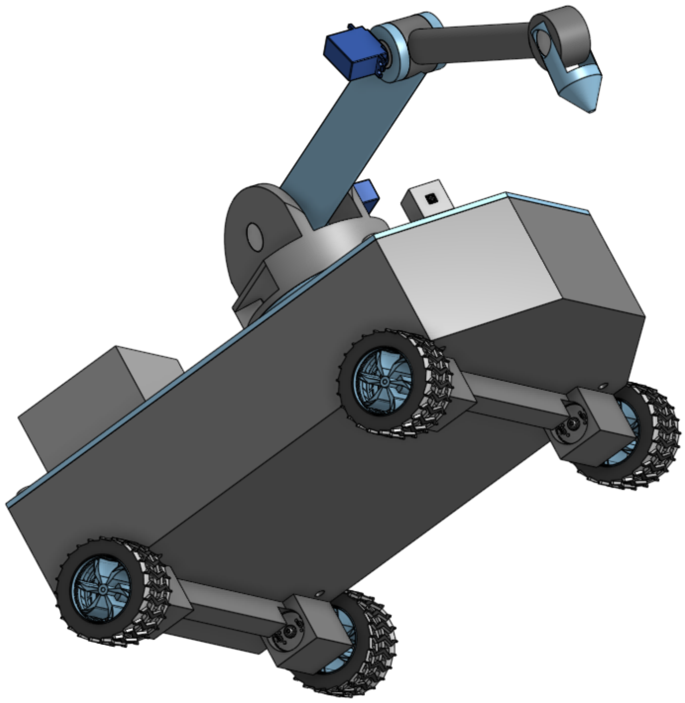


Figure 3: Elevated Wheel design

- The navigation system of this rover is based on a 4-wheel drive configuration, providing excellent traction and stability across uneven terrain.
- The rover is capable of moving forward, backward, and performing stationary rotations to change its orientation without requiring additional space.
- The steering system is controlled using a PID (Proportional-Integral-Derivative) algorithm, allows us to accurately navigate to thte desired locations.
- The PID-based control ensures stability during movement and efficient response to real-time environmental changes.

0.2.3 Weight Optimization

- Material selection: The casing has been built using a lightweight but durable aluminium alloy.
- Component Choice: We've chosen smaller, more energy efficient electronic components without compromising performance.
- Part distribution: The load has been distributed evenly among all 4 wheels.

0.2.4 Pesticide Tank

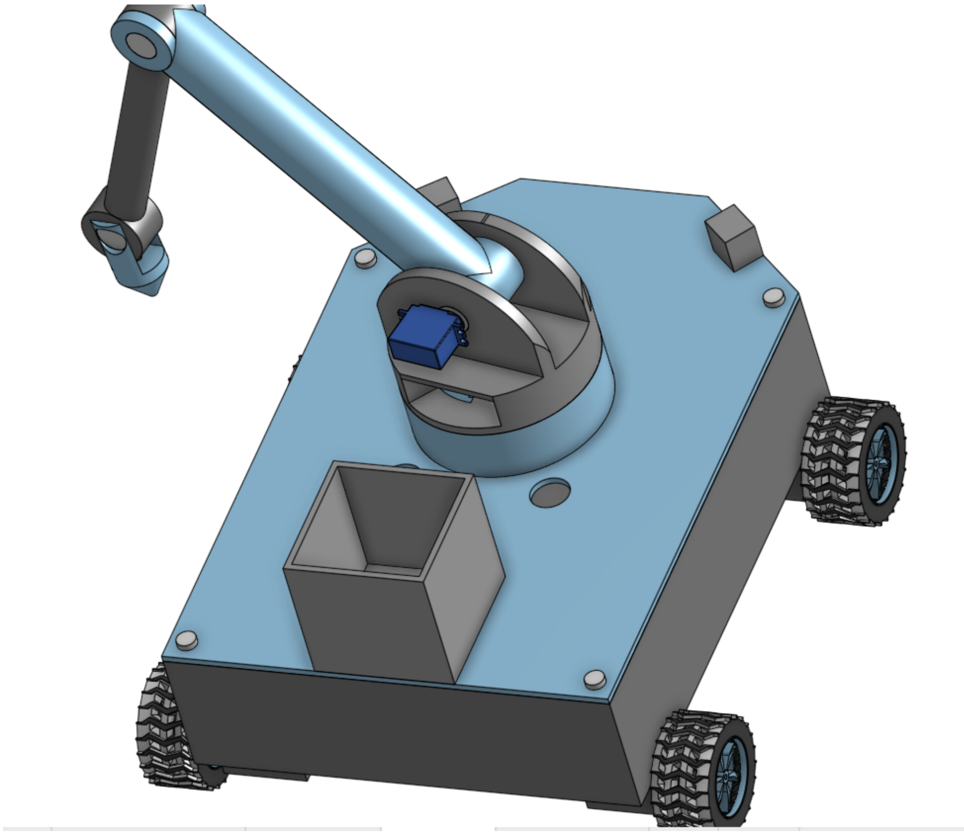


Figure 4: Pesticide Storage tank

- The robot is equipped with a pesticide storage tank to hold the required chemical for spraying tasks.

0.2.5 Chassis

The electronics for autonomous navigation of robot will be placed inside the chassis, which isolates the electronics that otherwise could be damaged due to environmental conditions. Also has bores for electrical connections to the robotic arm and the pump for spraying mechanism.

0.3 Electronics and circuit

0.3.1 Circuit Design

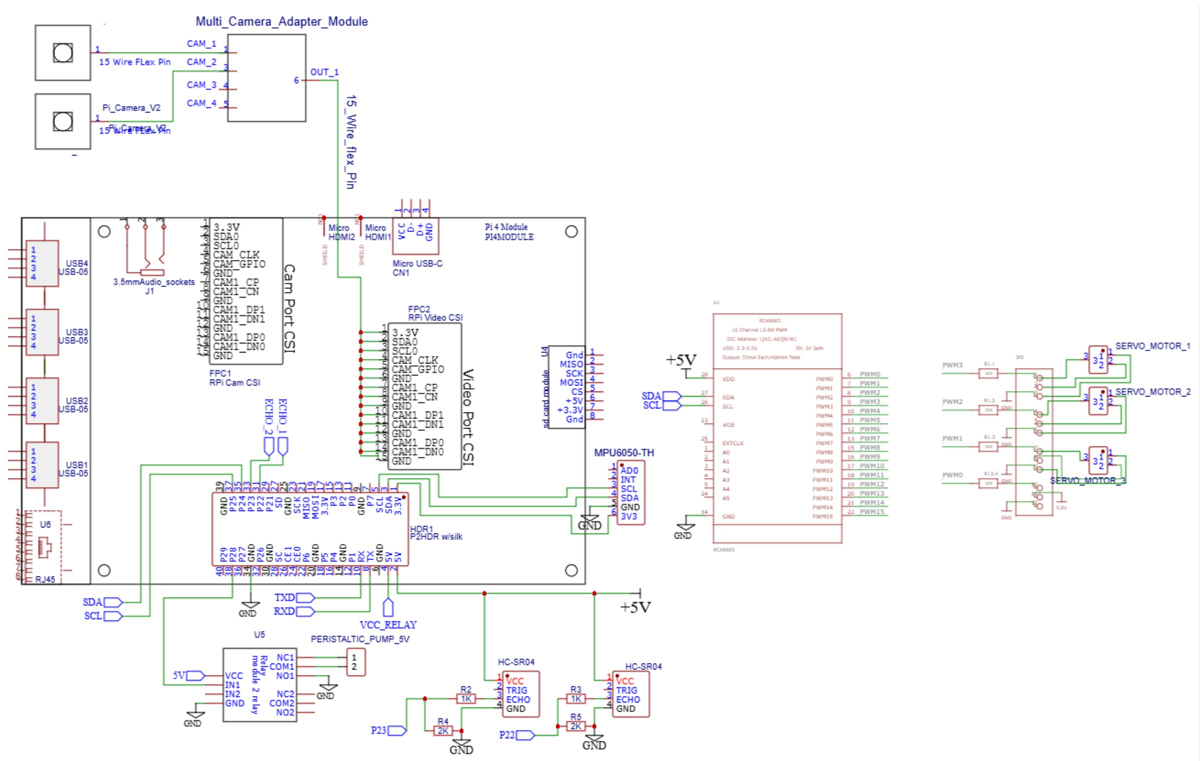


Figure 5: Circuit Diagram For Weed Detection And Herbicide Spraying

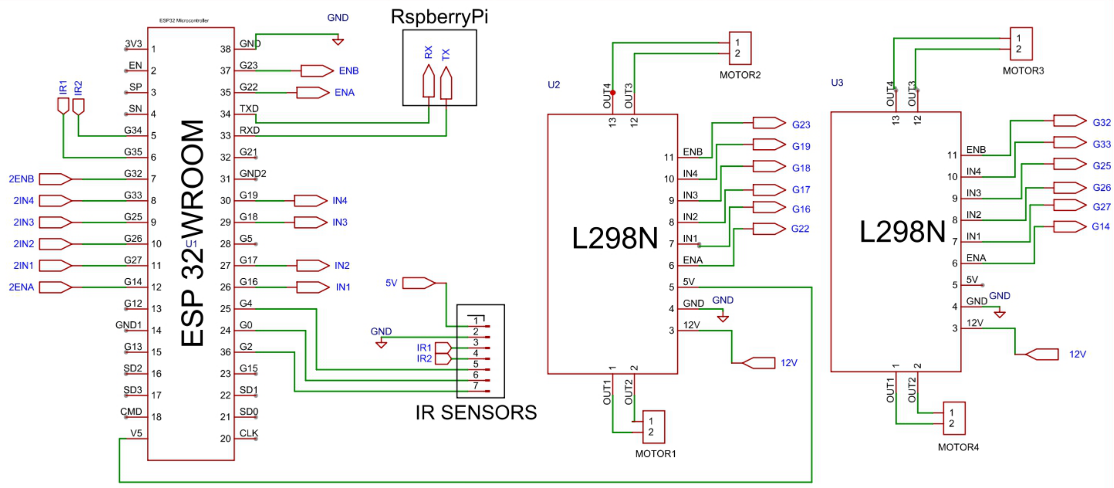
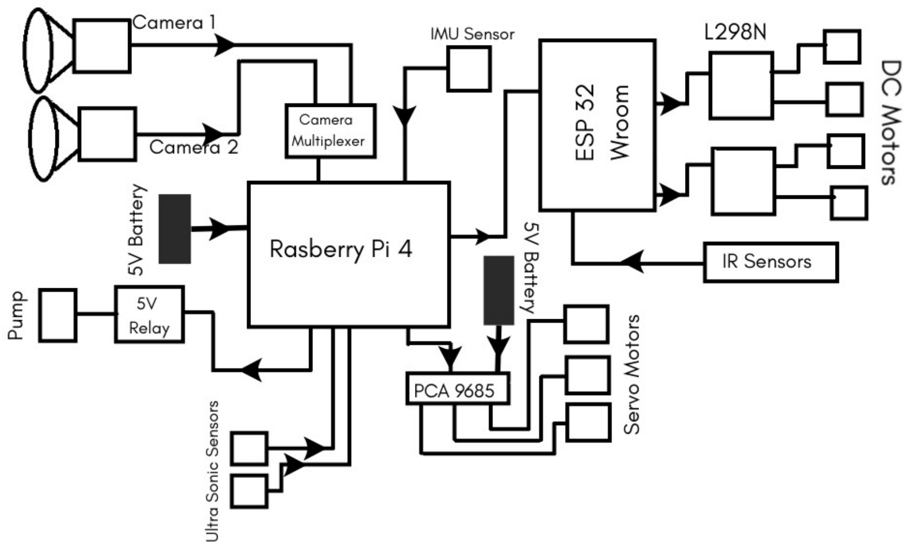


Figure 6: Circuit Diagram Of The Line Following System

Electronic components

- Li-ion battery of 12 V, 3000 mAh
- 5V batteries
- Motor Driver(L298N)
- PCA 9685
- Arducam Multi camera board
- MG90S Servo Motor
- 5V Relay module
- Invento 5 IR Sensor Array
  - Enables real time line detection and accurate path following
- HC-SR04 Ultrasonic Sensor Module
  - Calculates distance for tracking positions.
- Raspberry Pi Board
  - Employs real time image processing to enable the spray mechanism
- Raspberry Pi Camera V2
  - Enables distance mapping and color detection.
- 5V DC Peristaltic Pump
- Inertial Measurement unit (IMU) - Bosch BN055

Figure 7: Electrical design



Electrical assembly

- The robot will be powered by a 12 V battery, which supplies power to both L298N motor driver modules.
- Two L298N drivers will control four DC motors in total, with one driver assigned to each side of the robot (left and right motor pairs).
- An ESP32-WROOM (38-pin) microcontroller will generate PWM and direction signals for the motor drivers, handling speed regulation and steering.
- A 5-channel IR sensor array will be connected to the analog inputs of the ESP32 to provide real-time line-tracking feedback, implemented via a PID algorithm.
- The ESP32 will derive its power from the 5 V regulated output of one L298N driver; all grounds (battery, motor drivers, ESP32, IR array, Raspberry Pi) will share a common reference.
- A Raspberry Pi (primary image-processing unit) will communicate with the ESP32 via a two-wire UART link (TX/RX) and shared ground.
- Under standard conditions, motor control is handled locally by the ESP32 using IR inputs; the Raspberry Pi can override this and issue a global stop command, halting all motors simultaneously when required.
- The Raspberry Pi will be powered separately through a stable 5 V supply rated $\geq$ 3 A; it is not powered from the motor driver's regulator to ensure reliability.

- The Raspberry Pi 4 is powered by a dedicated 5 V battery supply, independent of the ESP32 and motor drivers, to ensure stable operation for image processing and peripherals.
- Two Raspberry Pi Camera Module 2 units are connected to the Pi through a multi-camera adapter board. The adapter sends signals from both cameras to the Pi's CSI video interface via a 15-pin flex cable, enabling synchronized image capture for object detection and navigation.
- The Raspberry Pi also controls a robotic arm used for pesticide spraying:
  - The arm consists of three servo motors, controlled using a PCA9685 PWM servo driver module.
  - The PCA9685 is powered separately by another dedicated 5 V battery, ensuring isolation of high-current servo loads from the Pi's power rail.
- For pesticide spraying, a 5 V peristaltic pump is used:
  - The pump is controlled through a 5 V relay module interfaced with the Raspberry Pi GPIO pins.
  - This allows on-demand activation of the pump for pesticide delivery when the robotic arm positions the nozzle over the target.
- Two ultrasonic sensors are mounted on the robot to measure the distance of target objects detected via the camera modules. This enables precise spraying without collision or overspray.
- The data from the IMU sensor combined with the AHRS algorithm in the Raspberry Pi track the robot's position and ensure that detected objects are not re-identified and sprayed multiple times.
- All components share a common ground reference for stable signaling between subsystems.

0.4 Algorithm

Line Follower with PID + Stop/Resume Command From Raspberry Pi

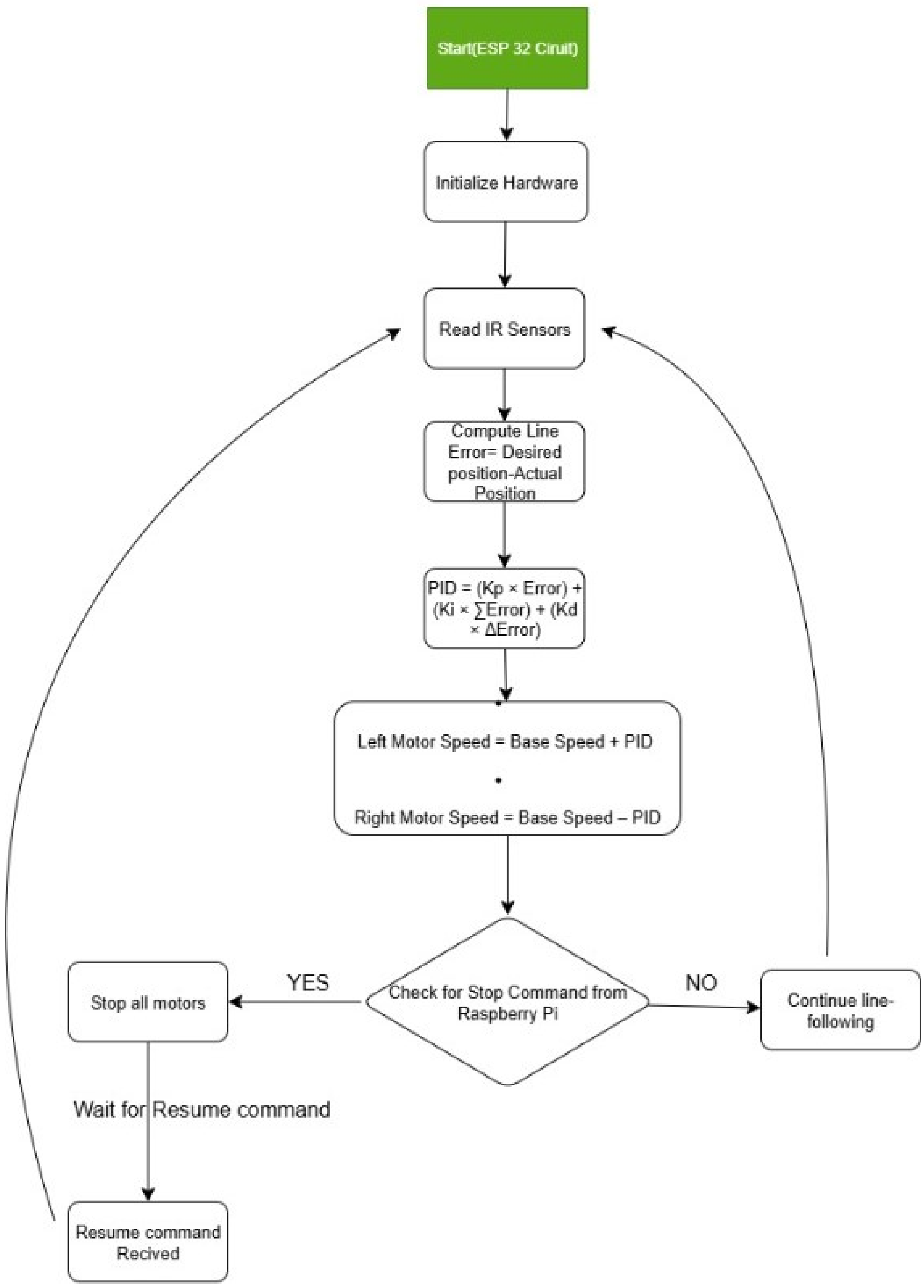


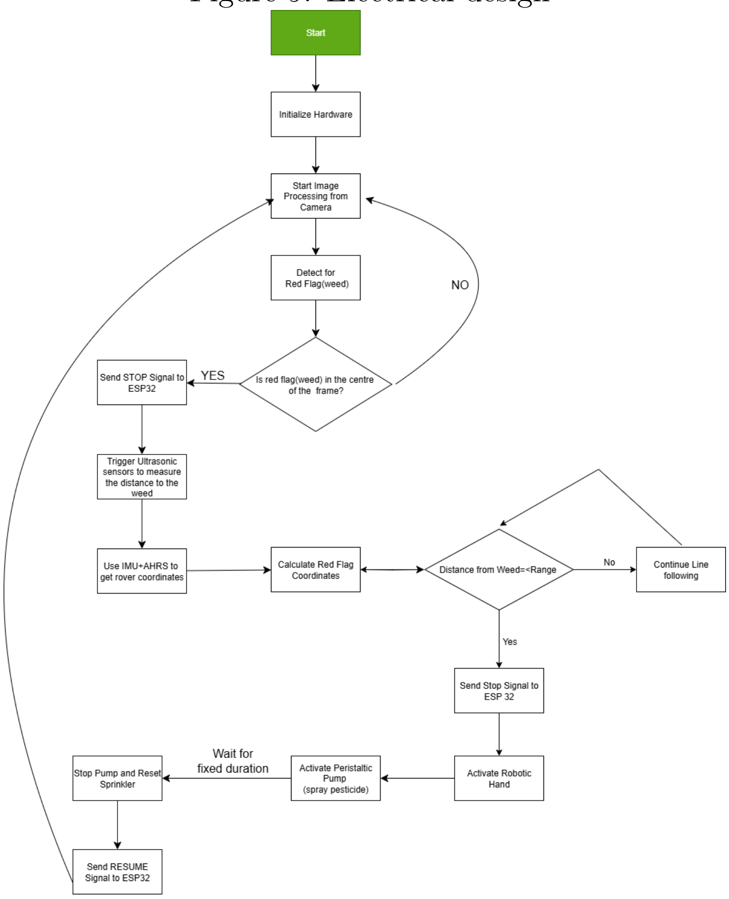
Figure 8: Control algorithm for Line following

- The rover's line-following mechanism uses an IR array sensor to accurately detect its position relative to the line and adjust its movement.
- PID (Proportional-Integral-Derivative) control system is integrated into the algo-

rithm to enhance the rover's speed and improve its efficiency by making smooth, precise adjustments.

- Altitude and Heading Reference System (AHRS) algorithms fuse data from the IMU's gyroscope, accelerometer, and magnetometer to estimate the rover's real-time orientation (roll, pitch, yaw).
- Relative position tracking is achieved by combining heading information with distance travelled from encoders or ultrasonic triggers to update coordinates from an initial origin (0,0,0).
- Camera input is combined with ultrasonic sensor readings to determine the distance and position of each detected weed.
- Since the rover's position is known from the AHRS, the coordinates of each detected weed can be accurately mapped relative to the rover.
- This integrated mechanism helps avoid repeated detections of the same weed and reduces false positives.

Figure 9: Electrical design



0.5 Software

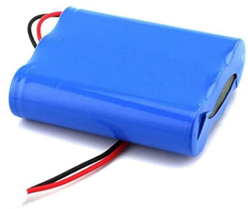
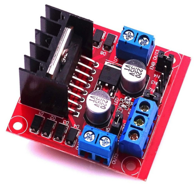
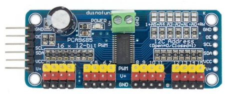
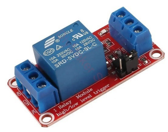
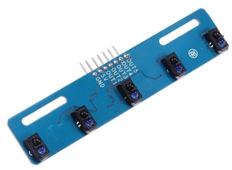
0.5.1 Pseudo Code

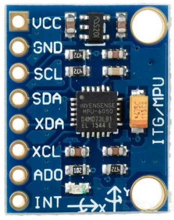
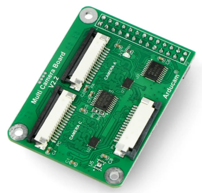
```
START
//DEFINE:
    Motor pins for LEFT driver: IN1, IN2, ENA, IN3, IN4, ENB (PWM)
    Motor pins for RIGHT driver: IN1, IN2, ENA, IN3, IN4, ENB (PWM)
    IR sensor pins: S1..S5 (left to right)
    UART RX/TX pins for communication with Raspberry Pi

SET:
    PID constants: Kp, Ki, Kd
    BASE_SPEED (normal motor speed)
    MAX_SPEED (upper limit)
    Previous_error = 0
    Integral = 0
    run_flag = TRUE  // Indicates whether motors should run

INITIALIZE:
    Configure motor pins as OUTPUT
    Configure PWM for ENA and ENB
    Configure IR sensor pins as INPUT (ADC or digital)
    Initialize UART on RX/TX pins for communication

LOOP forever:
    // ---- CHECK FOR COMMANDS FROM RASPBERRY PI ----
    if UART data available:
        command = READ UART data
        if command == "STOP":
            run_flag = FALSE
            STOP all motors (set PWM to 0)
        if command == "RESUME":
            run_flag = TRUE

    // ---- IF STOPPED, WAIT UNTIL RESUME ----
    if run_flag == FALSE:
        continue LOOP  // skip sensor reading & movement until resume
```

```
    // ---- SENSOR READING ----
    READ 5 sensor values (S1..S5)

    // Map sensor readings to a position error
    ASSIGN weights: [-2, -1, 0, +1, +2]
    For each sensor:
        if line detected under sensor: add weight to sum
    CALCULATE position_error = weighted_sum / number_of_active_sensors
    (if no sensor detects line: use previous error or fallback strategy)
 // ---- PID CALCULATION ----
    Proportional = position_error
    Integral += position_error
    Derivative = position_error - Previous_error
    PID_output = (Kp * Proportional) + (Ki * Integral) + (Kd * Derivative)

    Previous_error = position_error
 // ---- MOTOR SPEED CONTROL ----
    Left_motor_speed = BASE_SPEED - PID_output
    Right_motor_speed = BASE_SPEED + PID_output

    LIMIT Left_motor_speed, Right_motor_speed to 0..MAX_SPEED

    // Drive motors with calculated speeds
    if Left_motor_speed >= 0:
        set LEFT_IN1 = HIGH, LEFT_IN2 = LOW
    else:
        set LEFT_IN1 = LOW, LEFT_IN2 = HIGH
    PWM LEFT_ENA = abs(Left_motor_speed)

    if Right_motor_speed >= 0:
        set RIGHT_IN3 = HIGH, RIGHT_IN4 = LOW
    else:
        set RIGHT_IN3 = LOW, RIGHT_IN4 = HIGH
    PWM RIGHT_ENB = abs(Right_motor_speed)
    WAIT small delay (10-20ms) for loop timing
END LOOP//
```

0.6 List of components

Electrical Components

Component	Application	IMAGES
Battery; 12V, 3000mAh	Power Source	
Motor Driver; L298N	Controlling the motors	
Servo Motor; MG90S	Steering	
Servo Driver; PCA 9685	Precise motion control	
5V Relay module	Relaying signals to the pump	
Invento 5 IR Sensor Array	Line tracking and path detection	
Raspberry Pi V2.1 Camera	Line tracking	

Component	Application	IMAGES
5V DC Peristaltic Pump	Spraying Pesticides/Herbicides	
IMU	Measures acceleration	
HC-SR04 Ultrasonic sensors	Distance Mapping	
Arducam Camera module	Multi Camera adapter	

Mechanical Components

Component	Application	IMAGES
Aluminium sheet	Chassis	-
Servo motor	Steering	MG90S
PLA	robotic arm	-
Nuts and bolts	assembly	H9,H3
Wheels	Navigation	
Nozzle	spraying	-

0.7 Applications

- Uses image processing to distinguish weeds from crops, ensuring healthier plant growth and improved yields.
- Can be used to monitor soil and crop health using additional sensors, like moisture, pH, and temperature.
- Enables precise spraying of pesticides/herbicides, reducing excess chemical runoff into the soil, promoting sustainable practices.
- Automates repetetive manual labor, reducing operation costs.

0.7.1 Social Benefites

- Detects and targets weeds precisely, reducing excessive pesticide use and protecting the environment.
- Lowers costs for farmers while enhancing crop yield and quality.
- Improves worker safety by automating hazardous tasks.
- Uses smart sensors and data to support sustainable farming practices.
- Prevents overuse of chemicals for eco-friendly agriculture.
- Provides affordable automation for small and rural farmers.

0.8 Repository

Github repository link

- The repository will be a storage space for all essential files, CAD models, 3D printing oututs, weed detection algorithms, circuit diagrams and other required materials.
- This also allows us to collaborate with the team efficiently.
- This repository will be maintained throughout the project.