# LAWN MOWER PROJECT

# THE MOW-HAWK

## Requirement Specification Document

**Prepared by -**

**Aadithya Ramamurthy (Mat Nr - 3304090)**

**Vedhashruthi Harinath (Mat Nr - 3302834)**

# CONTENTS PAGE

## LIST OF FIGURES PAGE

# PROJECT DRIVERS

## 1. Purpose of the project

### 1a. The User Business or Background of the Project Effort

- The purpose of this project is to effectively learn Systems Engineering and Embedded systems used in Mechatronics by building a robotic autonomous lawnmower to help our customers have more time, save efforts by relieving them from the task of mowing their lawn manually even in undesired weather conditions. Through this project we aim at understanding the basic of the embedded systems and to clarify the misconception of the conventional lecture systems engineering as project-management. On one hand Systems Engineering is the process to develop systems and on the other, Project management deals with the operative techniques to manage projects.

### 1b. Goals of the Project

- Our goal is to develop and deliver a protype of a functioning mower robot which is based on a low-cost robot platform with an Arduino Uno R3 and nodeMCU ESP32 for WiFi connectivity and controlled by a mobile phone as per the requirements.

## 2. The Client, the Customer, and the Other Stakeholders

### 2a. The Client and the Customer

- The client here who is also the customer will be Prof. Dr.-Ing. Jörg Wollert who evaluates our project and grades it.

### 2b. The Stakeholders

- We, the students, are the stakeholders and also the contractor whose task is to do my best in this task by performing and working to my utmost capabilities in the given time constraints.

## 3. Users of the Product

- The project is fictitious purely intended for academic and learning purpose. Once the evaluation is done, it will be disassembled.

# CONSTRAINTS

## 4. Mandated Constraints

### 4a. Solution Constraints

- Base for this project is a DIY kit available on Ebay for about 20 Euros named “Smart Robot Car Assembly DIY Kit set”.

Fig 1: Smart Robot Car Assembly DIY Kit set

- Arduino Uno R3 is to be used as the microcontroller to control the operations of the mower.


Fig 2: Arduino Uno R3

- We use the NodeMCU which is Arduino-programmable for the communication between the robot and mobile phone. The version is ESP32 which has both Wi-Fi and Bluetooth 4.2 connectivity.

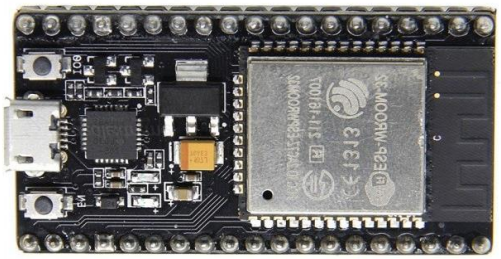

Fig 3: NodeMCU ESP32

**4b. Off-the-Shelf Software**

- We use the open source software Arduino IDE for the programming of the NodeMCU and the Arduino Uno.

**4c. Anticipated Workplace Environment**

- The mower is to be installed in a controlled environment with a boundary, in which the movements of the mower will be a constraint.
- It is to be implemented and the moved on a wooden floor surface.
- A good WiFi coverage range since the mower is controlled on a web application which uses the WiFi.
- Favourable weather conditions are required because the mower is not designed to be water, heat or cold resistant.
- The surface is dry and flat (free of any unevenness).
- The slope of the surface is less than 10 degrees.

**4d. Schedule Constraints**

- The mower should be delivered by the 7$^{th}$ of July 2021 to the Client in an operating condition.

**4e. Budget Constraints**

- The initial budget of 20 Euros for the DIY Kit and an additional budget of 20 Euros can be used for the hardware.

## 5. Naming Conventions and definitions

- The Motor Industry Software Reliability Association (MISRA) standard is to be followed.
- MISRA C is a set of software development guidelines for the C programming language developed by MISRA. Its aims are to facilitate code safety, security, portability and reliability in the context of embedded systems, specifically those systems programmed in ISO C / C90 / C99.

## 6. Relevant facts and assumptions

- The blades are not assembled on the system instead they are hypothetical.
- Similarly, the charging station for the mower is not present but is simulated. However, the charging of the bot is manually done after it reaches the fictional charging station when battery is low.
- The sensors and microcontrollers used in the mower will not fail during the building of the bot.

# FUNCTIONAL REQUIREMENTS

## 7. Scope of the work

### 7a. The Current Situation

Selling points of some existing mowers were studied during market research and are implemented in our mower. They are:

- User friendly control panel and mobile application.
- Battery detection and return to charging station.
- Free navigating.
- Overcoming a slope of 10 degrees.

### 7b. The Context of the Work

- The Tentative Work Context Diagram of the Mower Manufacturing is as follows:

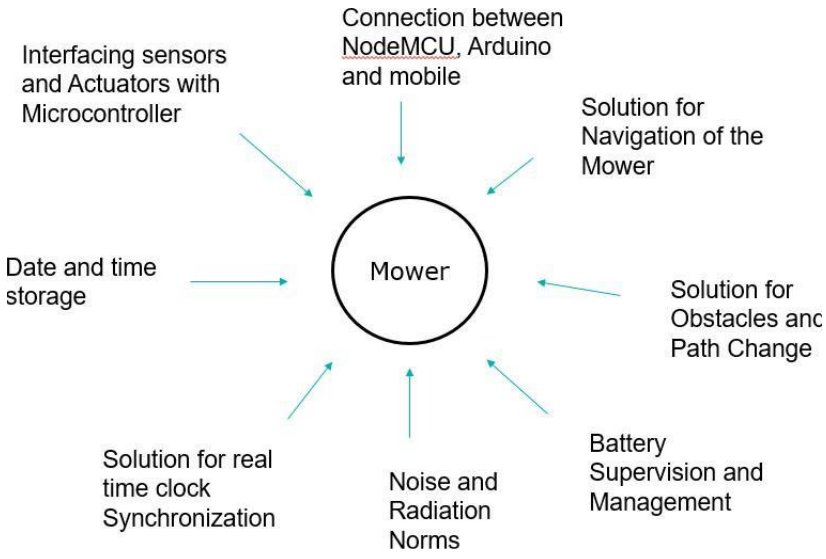


Fig 4: Work context diagram

# 8. Scope of the Product

- Battery supervision and management.
- It will have a clean and interactive user interface for the web application.
- Easy to use and easy administration of the product with one touch operation.

# 9. Functional and Data requirements

## 9a. Functional Requirements

- **Autonomous navigation:**
  It should be a free navigating mower that moves forward and backward, takes left and right turns, detects and avoids obstacles using the Ultrasonic sensor and servo motor and detects boundaries which are the black tapes and moves inside it according to the path planned with the help of the IR sensors mounted on the mower.
- **Low battery detection and return to charging station:**
  The battery percentage is measured using the INA219 sensor. In case the battery percentage of the mower drops below a certain limit, the bot enters the line following state. The same boundaries act as the line and IR sensors are used again for this. The mower follows this black

line, and it reaches the fictional charging station and comes to a halt and the batteries are charged manually.

- **Overcoming the slope:**
  The mower must handle slops up to 10 degrees and function normally even in these slopes. Duty cycle value of the PWM signal given to the motors are given accordingly to get the mower to work normally on the slopes.
- **Edge Detection and avoidance:**
  When a sharp edge is detected, the mower should stop and return. This is done using the help of IR sensors.
- **Manual mode:**
  Manual mode of operation is also incorporated on the web app so that whenever there are issues with the sensors, manual mode can be used to operate the mower.
- **On-board switch**
  A switch is provided on the mower to start or stop the power supply from the batteries to the motors, microcontrollers, and the sensors.
- **Power On/Off and Emergency stop through web app.**
  There is also an option to power on and off the motors on the web app and emergency stop button is also provided on the web app for safety purposes and in case of any malfunctioning of the mower.
- **Human Machine Interface:**
  The Mower will display the ongoing states, procedures and the battery levels on the LCD display connected to the Arduino via an I2C module and these will be displayed on the web app as well. Using the web app we can initialise the mowing process, select manual or automatic mode, monitor the battery, and log the date and time of mowing.

**9b. Data Requirements**

- Algorithms in the form of C++ codes in Arduino IDE are compiled in the Microcontroller.
- Operation time and date of mowing will be stored on the web app.
- Real time clock is synchronized with the mobile phone.
- This is browser based. Hence no additional software or apps are necessary.

# NON-FUNCTIONAL REQUIREMENTS

## 10. Look and Feel requirements

- The connecting wires, circuit boards and control units shall be arranged in an organized manner and be aesthetic, and it shall not look unorganized.
- Compact and clear functionality.
- Simple Human-Machine interface.

## 11. Usability and humanity requirements

- It is user friendly as it can be controlled with a minimum button set.
- Can be used and controlled by any layman without any training or technical background or even a physically handicapped person.
- The user shall be able to follow the steps provided in the user manual on how to use the mower.

## 12. Performance Requirements

- The desired area of 3m*3m must be covered in a single charge.
- The mower runs as long as the battery lasts.
- The boundary detection and obstacle avoidance should be highly accurate and response time should be quick to avoid mowing outside the boundary and avoid colliding with the obstacles.
- Sensors and the microcontrollers should be reliable and long lasting.
- The robot which will be equipped with suitable motors should have the required speed and torque for quick and smooth movement and mowing function.

## 13. Operational and Environmental requirements

- The web app should be accessible from mobile or PCs of any operating system.
- The functionality of the web app shall be the same in upcoming releases of the OS.
- The surface must be smooth and work environment must be free of wet conditions since the robot cannot detect wet conditions.

## 14. Maintainability and Support requirements

- Easy to maintain with all spare parts available in the market.
- External battery is used for easy replacement in case it is necessary.
- The mower should be free of any regular maintenance.
- A user manual is provided on how to set up and operate the mower.

## 15. Security requirements

- The Robot mower will be protected from unauthorised usage. It can only be accessed by those who has the Wi-Fi credentials.
- Credentials are required to login to the web app.

## 16. Legal Requirements

- The Mower will comply with the local norms and legal requirements.

## 17. Cultural and Political Requirements

- This project would of course not have any cultural and political obligations.

# PROJECT ISSUES

## 18. Open Issues

- Path planning of the mower to cover the desired area completely.
- Voltage from the batteries were not sufficient to power all the components at once including the motors.
- Programming of the mower to make it work as per the requirements.
- Connections to the microcontroller with limited number of pins.

## 19. Off-The-Shelf Solution

- We make use of these solutions for a quick realization of the product to complete the product within the time and budget constraints.
- We take help of open-source platforms to develop the prototype.
- There exists a lot of libraries and example codes which can be made use of while programming our mower.

- An external power bank is used to power the microcontroller and the sensors while 6V AA batteries were used to power the motor.
- Arduino V5 Shield is used to make the connections easier.

# 20. Tasks

## 20a. Project Planning using Trello board

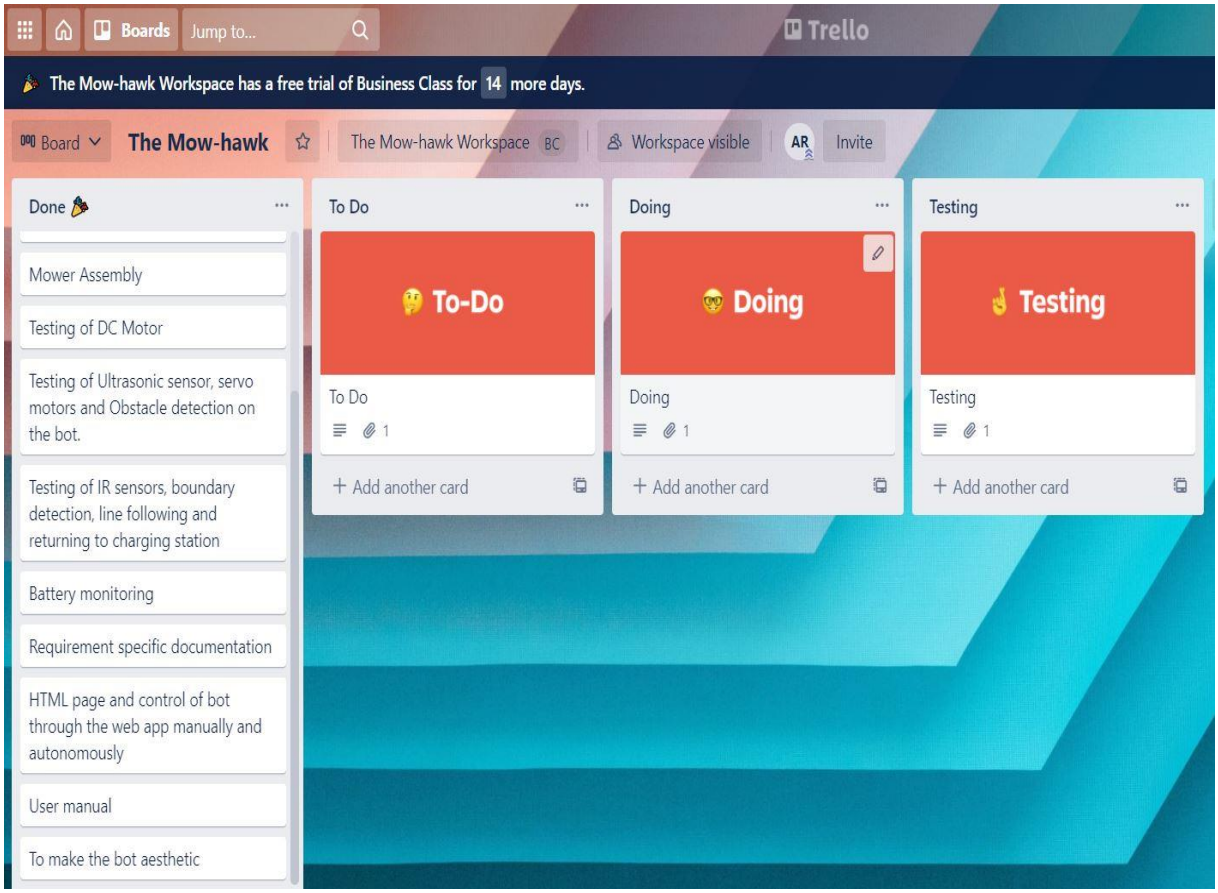


Fig 5: Project plan on Trello board

# 21. Risks

- The major risks involved are the malfunctioning of the sensors and the microcontrollers when in use. This poses a serious threat while operating the mower.

### 21a. Design Failure mode and Effect analysis (DFMEA)

| | The Mow-Hawk Design Failure mode and Effect Analysis | | | | | | |
|---|---|---|---|---|---|---|---|
| | System: Mower Robot<br>Components: Arduino Uno, V5 shield, NodeMCU ESP32, L298N, Ultrasonic Sensor, Servo motors, DC Motor, IR Sensors | | System Failure Mode and Effect Analysis | | | | |
| | Function | Requirement | Potential Failures | Severity (1 to 10) | Potential effects | Potential Cause of Failures | Recommended measures |
| 1 | Bot moving forward | Batteries having charge | Battery Voltage not being monitored | 7 | Robot stops mid-field | Electrical connection is loose | Securing all connections and supports provided |
| | | | Lost track of returning to base | 6 | Robot stops mid-field or travels outside the boundary | Unable to detect the boundary | Calibrate IR sensors |
| | | DC Motor functioning properly | DC Motor stops functioning | 8 | Bot Stops | DC Motor end of life | Replace DC Motor |
| | | | | | | DC Motor connection failure on L298N | Secure Connection |
| | | | | | | Soldering failure | Soldering joints and fastening the joints to bot for support |
| | | L298N functioning properly | Wheels not rotating as per code | 4 | Does not avoid obstacles<br>Keeps rotating in circles<br>Does not cover entire area | Battery charge is lower than danger levels | Check Battery monitoring setup<br>Check battery connections with L298N for possibility of short or loose connection |
| | | Arduino Functioning properly | Servo motor malfunction | 5 | Obstacles not detected | | Reset Arduino and upload Code again |
| | | | L298N malfunction | 9 | Bot moves not as intended | | |
| | | | Ultrasonic sensor malfunction | 8 | Obstacles not detected | | |
| | | | IR sensors malfunction | 9 | Boundary not detected | | |
| 2 | Staying within the Boundary | IR Sensors | Failure in tracking the boundary line | 7 | Bot moves outside Boundary | IR sensors not able to detect Line | Calibrating IR sensors |
| 3 | Detecting Obstacles | Ultrasonic Sensor, Servo motor | Fails in detecting obstacles | 8 | Bot gets damaged | Ultrasonic sensor not working | Calibrating Ultrasonic sensor |
| | | | | | | Servo motors not working | Calibrating Servo motors |
| 4 | Returning to Station for charging | Battery voltage monitoring, IR sensors for following boundary line. | Battery levels not checked | 6 | Battery charge completely drained out | Battery voltage monitoring code failure | Check connections between battery and Arduino |
| | | | Boundary Tape not detected | 9 | Bot moves out of defined boundary | IR Sensors unable to detect lines | Calibrate IR Sensors |

## 22. User documentation and Training

- A user manual is provided with details about installation and setup of the mower.
- The user shall also be informed about the limitation of the mower such as weather conditions, slope etc.

## 23. Waiting Room

Many ideas can be implemented in future generation of the mower such as:

- GPS for navigation and path planning.
- Lidar or IFM camera for obstacle detection and avoidance.
- Solar panels can be installed for efficient power supply.

# SYSML DIAGRAMS

## 24. Use case diagram

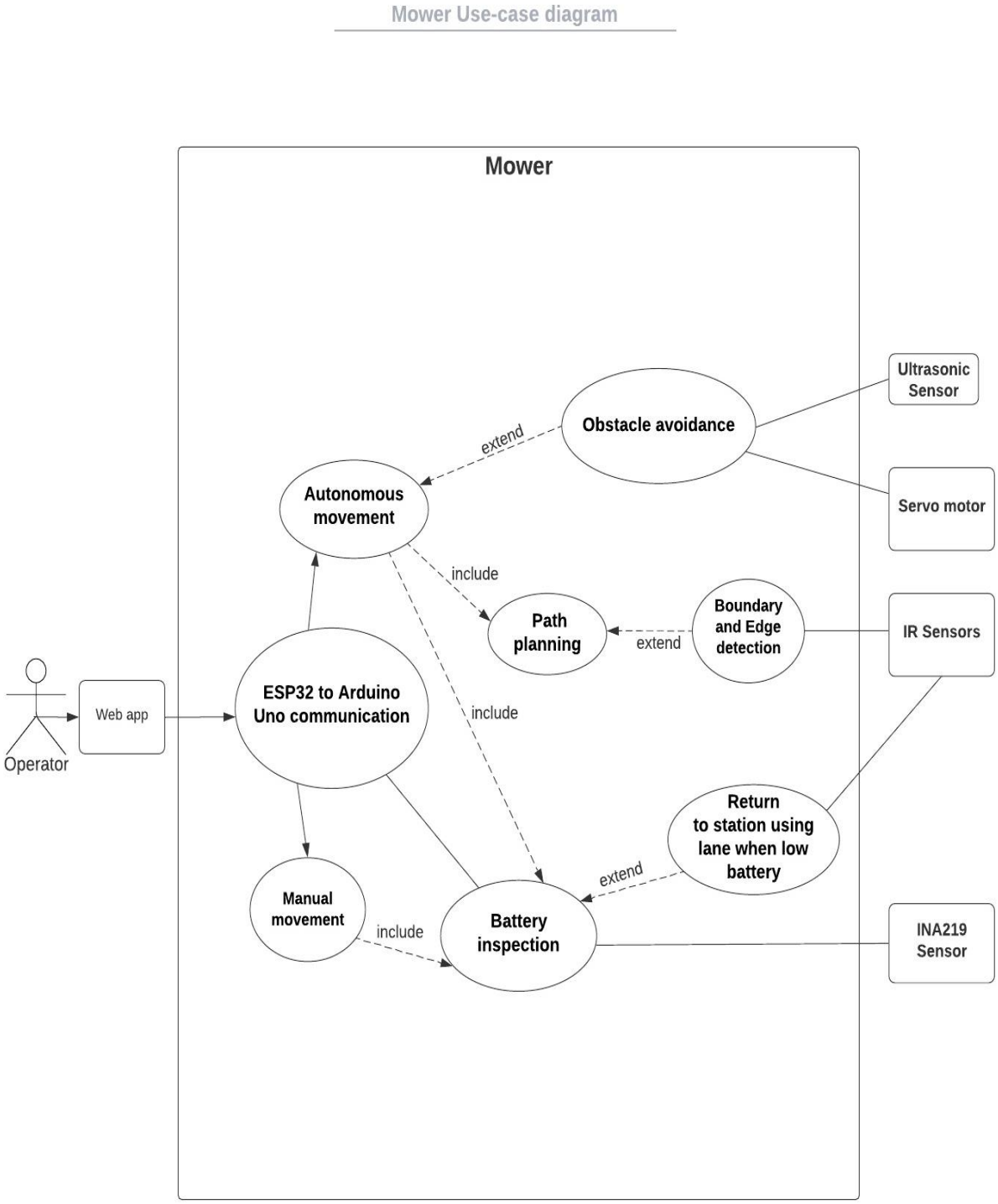


Fig 6: Use case diagram

# 25. Parametric diagram

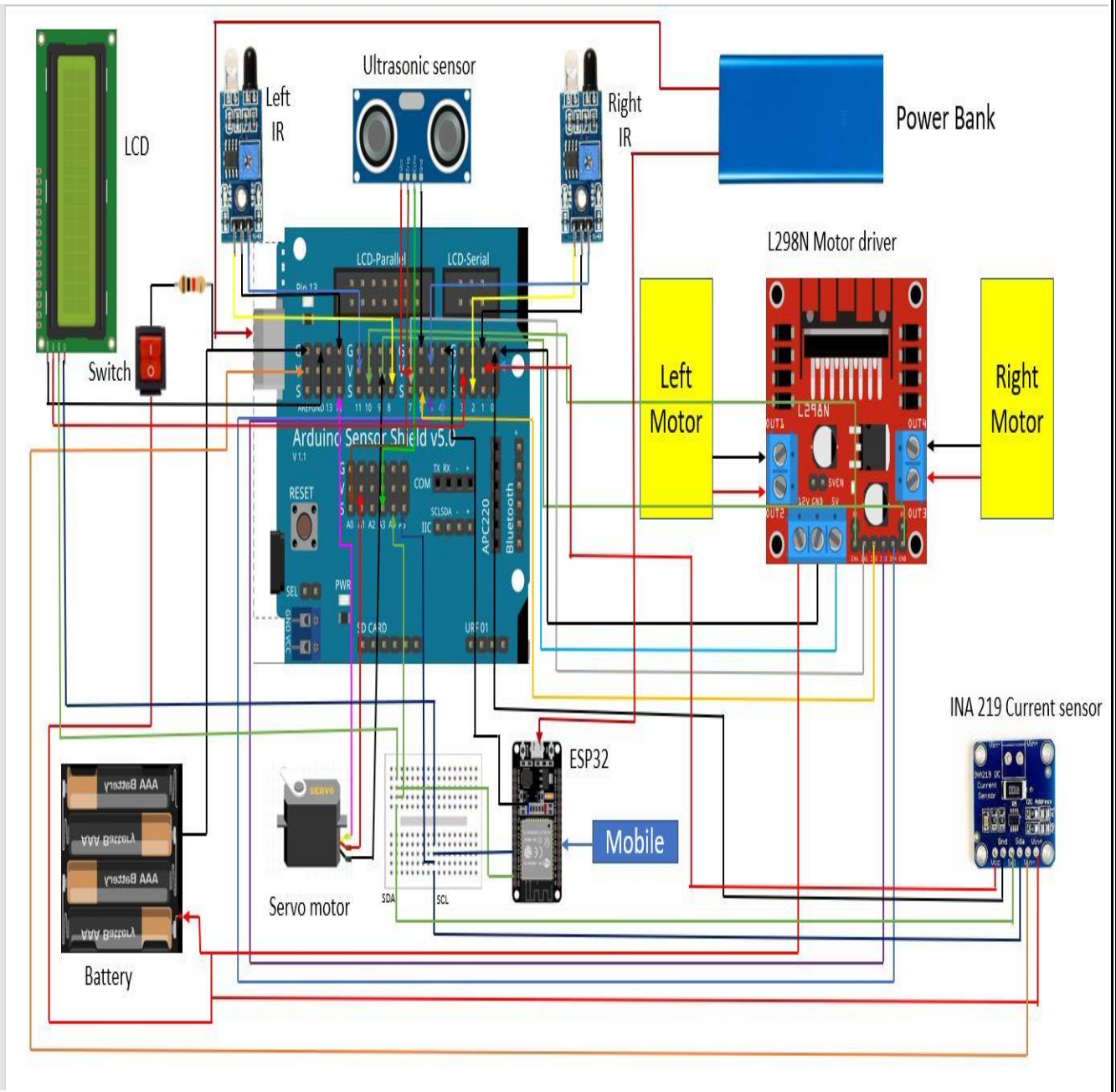


Fig 7: Parametric diagram

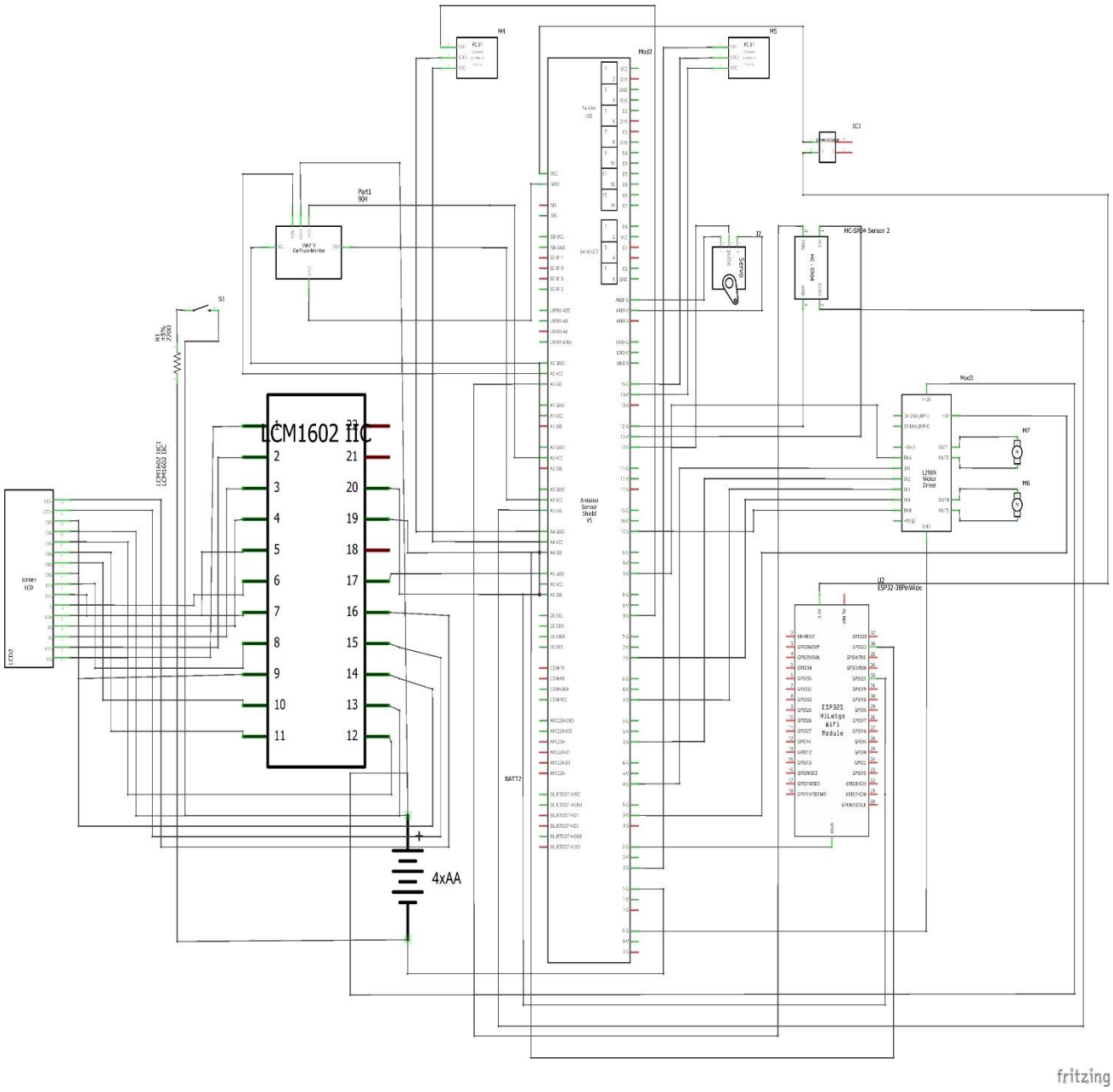


Fig 8: Parametric schematic diagram

# 26. State machine diagram

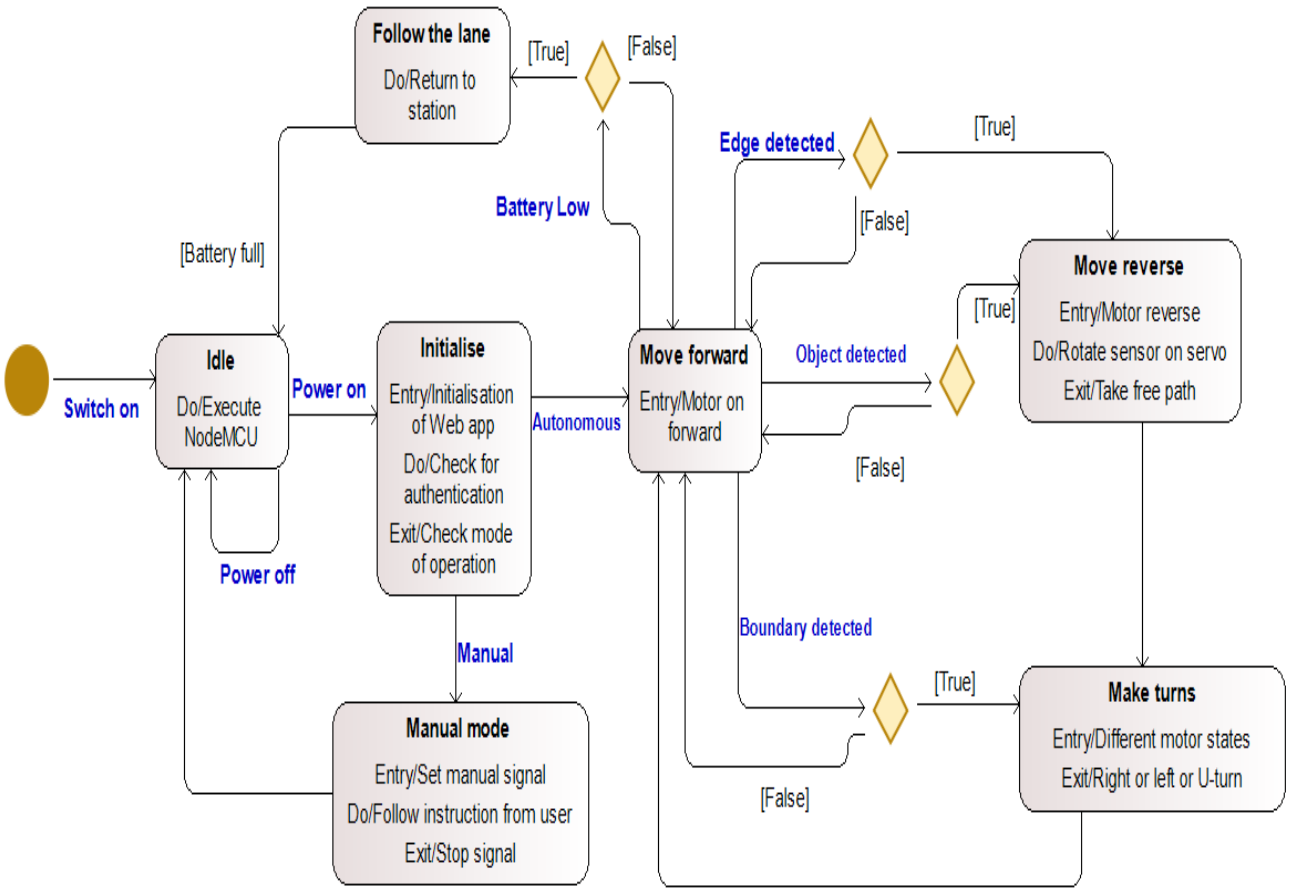


Fig 9: State machine diagram