# CHAPTER-01

# INTRODUCTION

## 1.1 OVERVIEW OF ANIMAL ACCIDENTS NEAR RAILWAY TRACKS

Animal accidents near railway tracks have become a growing concern, particularly in regions with dense wildlife populations and railway networks intersecting natural habitats. Animals, including elephants, deer, cattle, and other wildlife, often cross railway tracks in search of food, water, or shelter, leading to unfortunate collisions with moving trains. These accidents not only cause severe injury and death to animals but also result in damage to locomotives and pose safety risks to passengers. Studies have shown that railway tracks built through forest corridors and grasslands increase the likelihood of such accidents. For instance, elephants, due to their slow movement, are more vulnerable to train collisions. Nighttime crossings, poor visibility, and high-speed trains further contribute to the high frequency of accidents. Additionally, grazing cattle or stray animals from nearby villages unknowingly enter railway tracks, increasing the risk of collisions. The lack of real-time surveillance and early warning systems exacerbates the problem, making it difficult for train operators to react in time. Furthermore, these accidents lead to ecological imbalances by disrupting wildlife populations and biodiversity. Governments and environmental agencies have been working to mitigate these risks by implementing measures such as constructing underpasses, installing warning systems, and deploying surveillance technologies. However, a comprehensive IoT-based remote monitoring system can significantly enhance the detection of animals near railway tracks and prevent accidents by alerting railway authorities in real time, ensuring the safety of both animals and train passengers.

Railway is one of the most significant transportation modes of our country but it’s a matter of great anguish that rail tracks of our country are veritably prone. That’s why, a vast number of accidents are passed every time due to this primitive type of rail tracks and as the consequences of those accidents we lose huge number of lives every time. These types of incidents motivate us to suppose over the below mentioned issue and take necessary way to cover those lives. Through our proposed system, we need to establish further ultramodern and secure rail system. Besides this, there’s no similar type of technology or system in our country which can stop the collision between two trains coming from the opposite direction of each other on the same track. We actually think over this matter and motivated to do so. More over natural disaster can throw any object on the rail track which cannot be removed very quickly in the

remote area. We thought if our system can detect those object or barrier and inform to the control room then they can take necessary steps to avoid accident. The Rail transport is growing at a rapid pace in India. It is one of the major mode of transport but still our facilities are not that accurate, safer as compared to international standards. A survey on the internet states that about 60% of all the railway accidents is due to derailments, recent measurements shows that about 90% are due to cracks on the rails. Hence, it is not safer for Human Life. This needs to be at the utmost attention. These goes unnoticed and the properly maintenance of tracks is not done.

In previously existing system, the work is to be done manually, but the proposed system has a robot which will run automatically on the tracks. System having LED and LDR sensor assembly, but the main disadvantage is that the LED and LDR must be placed opposite to each other and also the environment needs to be perfect to detect the track. To overcome this disadvantage, here sensors are used, which will detect the crack accurately. The existing system is slow, tedious and time consuming. This system has GSM and GPS module which will give the real time location or coordinates in the form of Short Message Service (SMS) to the nearest railway station

## 1.2 IMPORTANCE OF REMOTE SURVEILLANCE

Remote surveillance near railway tracks is essential to ensure the safety of both wildlife and train passengers. With railway tracks often passing through forests, grasslands, and rural areas, animals such as elephants, deer, and cattle frequently stray onto the tracks, leading to fatal accidents. Real-time monitoring using advanced IoT technologies can detect animal movements and alert authorities before any collision occurs. Remote surveillance systems equipped with sensors, cameras, and communication modules can provide continuous monitoring of vast railway corridors, allowing timely interventions. These systems can significantly reduce accidents by enabling train operators to take preventive actions, such as slowing down or stopping the train. Additionally, remote surveillance enhances the ability to monitor trespassing, unauthorized activities, and potential sabotage, ensuring the security of the railway infrastructure. By using GSM and Wi-Fi modules, real-time alerts can be sent to control center, enabling quick responses. Surveillance also helps in data collection and analysis, which aids in identifying high-risk areas and formulating effective mitigation strategies. Moreover, it reduces the reliance on manual patrolling, which is often inefficient and costly. Implementing remote surveillance systems not only

safeguards wildlife but also prevents delays and damages caused by accidents, ultimately contributing to a more secure and efficient railway operation.

## 1.3 ROLE OF IOT IN WILD LIFE MONITORING

The Internet of Things (IoT) plays a crucial role in wildlife monitoring by enabling real-time data collection, analysis, and transmission from remote locations. IoT devices, such as GPS trackers, motion sensors, camera traps, and environmental sensors, help researchers and conservationists track animal movements, behavior and habitat changes with high accuracy. By deploying IoT-enabled systems, wildlife monitoring becomes more efficient, reducing the need for human intervention in dangerous or inaccessible areas. GPS and RFID tags attached to animals provide precise location data, enabling researchers to analyze migration patterns and detect unusual activities. Ultrasonic sensors and infrared cameras placed near protected zones or railway tracks can detect the presence of animals and trigger automated alerts to prevent potential conflicts. IoT-based systems can also monitor environmental parameters, such as temperature, humidity, and air quality, helping in assessing the impact of climate change on wildlife habitats. Additionally, IoT devices can integrate with GSM and cloud technologies to transmit real-time data to remote servers, where artificial intelligence (AI) algorithms analyze the information for anomaly detection and pattern recognition. This proactive approach enhances wildlife protection by enabling authorities to respond quickly to potential threats. IoT-based monitoring systems contribute significantly to the conservation of endangered species and help mitigate human-wildlife conflicts, ensuring a balance between ecological sustainability and develop

# CHAPTER 2

# LITERATURE REVIEW AND OBJECTIVES

## 2.1 EXISTING METHODS FOR ANIMAL DETECTION & TRACKING

Several methods are currently used for animal detection and surveillance near sensitive areas such as railway tracks and highways to prevent accidents and monitor wildlife movements. One common method is the use of camera traps equipped with motion sensors that capture images or videos when animals pass by. These cameras provide visual data, but they often require manual retrieval and analysis, causing delays in real-time response. Thermal imaging cameras are another advanced approach that detects heat signatures of animals, allowing for effective monitoring even at night. RFID (Radio Frequency Identification) tags and GPS collars are widely used to track the movement of tagged animals, providing real-time location updates to researchers and authorities. However, this method is limited to individual animals that have been tagged. Acoustic sensors are deployed to capture animal sounds, which helps identify the presence of specific species. Infrared sensors and ultrasonic sensors are also used to detect movement and proximity of animals near critical zones. Some regions use fencing and physical barriers combined with warning systems that activate alarms when an animal approaches the protected area. Additionally, manual patrolling by forest guards or railway personnel is still practiced, although it is time consuming and less efficient. Drone-based surveillance has recently gained popularity, allowing wide-area monitoring from the sky, but it faces challenges such as battery limitations and signal interference. Despite these advancements, most of these methods lack real-time intervention capabilities, making IoT-based solutions a promising alternative for enhancing surveillance and reducing animal accidents.

## 2.2 EMERGING IOT SOLUTIONS FOR ANIMAL TRACKING NEAR RAILWAY TRACKS :

1. Ultrasonic Sensors for Proximity Detection
    - Detect animal movement near railway tracks.
    - Measure distance and trigger alerts when animals cross predefined zones.
2. Infrared (IR) Sensors for Heat Detection

- Identify animal presence by detecting heat signatures.
- Work effectively at night and in low-visibility conditions.

3. GPS and RFID-Based Animal Tracking
    - Attach GPS collars or RFID tags to track animal movements.
    - Provide real-time location updates to control centers.
4. Camera Traps with IoT Connectivity
    - Capture images or videos of animals when motion is detected.
    - Transmit data to cloud servers for analysis.
5. AI-Powered Image and Video Analysis
    - Use artificial intelligence to identify animals and predict movement patterns.
    - Analyze captured footage to detect potential threats.
6. GSM-Based SMS and Call Alerts
    - Send automated SMS or call alerts to railway authorities.
    - Notify operators when animals approach the tracks.
7. LORA WAN for Long-Range Communication
    - Enable low-power, long-range communication between IoT devices.
    - Ideal for remote areas with minimal network coverage.
8. Edge Computing for Faster Response
    - Process data locally at the sensor level.
    - Reduce latency by avoiding cloud dependence for critical actions.
9. Geofencing for Virtual Boundaries
    - Create virtual zones around railway tracks.
    - Trigger alerts when animals cross predefined boundaries.
10. MQTT Protocol for Real-Time Data Transfer
    - Use lightweight protocols for fast and efficient communication.
    - Ensure minimal delay in alert delivery.
11. Drone Surveillance with Live Streaming
    - Deploy drones to monitor large areas near railway tracks.
    - Provide real-time video feed for immediate action.
12. AI-Based Predictive Analytics
    - Analyze historical animal movement data to predict future behavior.

- Enable proactive intervention in high-risk zones.

13. Solar-Powered IoT Devices
    - Use solar panels to power sensors and communication modules.
    - Ensure continuous operation in remote areas.
14. Sound-Based Animal Detection Systems
    - Detect animal sounds and analyze frequency patterns.
    - Identify species and trigger appropriate responses.
15. Vibration Sensors for Track Monitoring
    - Detect unusual vibrations caused by animal movement.
    - Alert operators about potential animal activity.
16. Integration with Railway Signaling Systems
    - Connect IoT systems to train signaling for automatic response.
    - Slow down or stop trains upon animal detection.
17. Automated Light and Sound Deterrents
    - Activate lights and sounds to scare animals away from tracks.
    - Prevent animals from entering dangerous zones.
18. Water Sprinkler Systems for Animal Deterrence
    - Spray water when animals approach railway tracks.
    - Deter animals without causing harm.
19. IoT-Enabled Smart Barriers
    - Control barriers to restrict animal access.
    - Automatically close barriers when animals are detected.
20. AI-Based Animal Behavior Analysis
    - Analyze animal behavior patterns for predictive monitoring.
    - Identify unusual or risky behavior near tracks.
21. Acoustic Sensors for Sound Detection
    - Capture animal sounds and alert control centers.
    - Useful for monitoring nocturnal animals.
22. Time-Lapse Analysis for Movement Patterns
    - Analyze long-term data to identify high-risk zones.
    - Improve preventive strategies based on historical trends.

23. Blockchain for Secure Data Storage
    - Store animal tracking data securely using blockchain technology.
    - Ensure data integrity and prevent tampering.
24. IoT-Enabled Wildlife Corridors
    - Create safe passage routes for animals.
    - Guide animals to designated crossing points.
25. Automated Horn and Warning Systems
    - Trigger warning sounds to alert animals of approaching trains.
    - Prevent collisions through timely alerts.
26. Wi-Fi and Zigbee Networks for Short-Range Communication
    - Enable seamless communication between IoT devices.
    - Ensure reliable data exchange within a defined range.
27. Automated Rescue Dispatch Systems
    - Notify rescue teams for immediate action during emergencies.
    - Reduce response time in critical situations.
28. Temperature and Humidity Sensors
    - Monitor environmental changes affecting animal movement.
    - Predict high-risk periods based on weather conditions.
29. AI Models for False Alarm Reduction
    - Filter false positives using advanced machine learning algorithms.
    - Improve accuracy and reliability of the system.
30. IoT-Enabled Animal Warning Systems
    - Warn nearby animals about approaching trains.
    - Use light and sound signals for effective deterrence.
31. Cloud-Based Data Analysis and Reporting
    - Store and analyze data from multiple IoT devices.
    - Provide insights for preventive maintenance.
32. Integration with Emergency Response Systems
    - Automatically dispatch emergency teams when required.
    - Ensure timely assistance in critical situations.
33. IoT Beacons for Proximity Alerts

- Deploy Bluetooth beacons to detect animal movement.
- Provide location-based alerts for immediate action.

34. Traffic Control with IoT-Enabled Gates
    - Manage train speed and traffic flow based on real-time data.
    - Reduce collision risk through automated gate control.
35. Digital Twin Models for Simulation
    - Create virtual models of railway tracks and animal activity.
    - Simulate potential threats and refine response mechanisms.
36. Crowdsourcing Data from Local Communities
    - Gather real-time information about animal sightings.
    - Enhance system effectiveness through community participation.
37. Smart Wildlife Fences with IoT Integration
    - Deploy electric fences that trigger only when animals approach.
    - Minimize unnecessary power usage while maintaining safety.
38. Automated Speed Regulation for Trains
    - Control train speed based on real-time detection data.
    - Prevent high-speed collisions with animals.
39. Time-Based Animal Crossing Prediction Models
    - Predict peak crossing times based on historical data.
    - Schedule preventive actions during high-risk periods.
40. Emergency Alarm Activation on Animal Detection
    - Automatically activate alarms upon animal detection.
    - Warn train operators and nearby workers for immediate action.

## MAIN OBJECTIVES

41. Prevent Animal-Train Collisions
    - Detect animals near railway tracks in real time.
    - Send alerts to railway authorities to prevent accidents.
42. Real-Time Monitoring and Surveillance
    - Continuously monitor the railway tracks for animal movement.

- Use ultrasonic sensors to detect proximity and alert the system.

43. Automated Alert System Using GSM Module
    - Send SMS or call notifications to relevant authorities.
    - Enable immediate response and preventive action.
44. Integration with Railway Control System
    - Link the system with railway signaling to slow down or stop trains.
    - Ensure automatic response to minimize collision risks.
45. Accurate Distance Measurement with Ultrasonic Sensors
    - Calculate the distance of animals from the tracks.
    - Trigger alerts when animals cross predefined thresholds.
46. Data Logging and Analysis
    - Store detected data for future analysis.
    - Identify high-risk zones and improve system efficiency.
47. Low-Power, Cost-Effective Operation
    - Use ESP32 for efficient processing and communication.
    - Ensure minimal power consumption in remote areas.
48. Scalability and Remote Access
    - Enable system scalability to cover longer railway corridors.
    - Provide remote access to data through IoT cloud integration.
49. Enhance Wildlife Protection and Conservation
    - Minimize harm to endangered species.
    - Protect animal habitats near railway tracks.
50. Automated Warning System for Train Operators
    - Alert train operators when animals are detected.
    - Ensure timely decision-making to avoid accidents.

## 2.3 EXISTING SYSTEMS

Several IoT-based systems have been developed to monitor and track animal movements near railway tracks, aiming to reduce the risk of animal-train collisions. One common approach involves using camera traps equipped with motion detection capabilities, which capture images and videos

when animals pass by and transmit the data to control centers. However, these systems often require manual analysis, which delays response times. RFID (Radio Frequency Identification) tags and GPS collars are used to monitor individual animals by providing real-time location updates, but these systems are limited to tagged animals and cannot detect stray or untagged animals. Ultrasonic and infrared (IR) sensors are also deployed along railway tracks to detect the presence of animals by measuring distance and heat signatures. Some systems use GSM and LORAWAN modules to send real-time SMS or call alerts to railway authorities when animals are detected. Advanced solutions incorporate AI and machine learning algorithms to analyze captured images and detect animal species, improving system accuracy. Geofencing technology is another method that creates virtual boundaries around railway tracks and triggers alerts when animals cross these boundaries. Despite these advancements, most existing systems face challenges such as false positives, limited range, and poor performance in adverse weather conditions. Emerging IoT solutions aim to overcome these limitations by integrating multiple sensor technologies, AI-based predictive analysis, and real-time communication to enhance the reliability and effectiveness of animal tracking near railway tracks.

**1. Prevent Animal-Train Collisions**

- Detect animals near railway tracks in real time.
- Send alerts to railway authorities to prevent accidents.

**2. Real-Time Monitoring and Surveillance**

- Continuously monitor the railway tracks for animal movement.
- Use ultrasonic sensors to detect proximity and alert the system.

**3. Automated Alert System Using GSM Module**

- Send SMS or call notifications to relevant authorities.
- Enable immediate response and preventive action.

**3. Integration with Railway Control System**

- Link the system with railway signaling to slow down or stop trains.
- Ensure automatic response to minimize collision risks.

**5. Accurate Distance Measurement with Ultrasonic Sensors**

- Calculate the distance of animals from the tracks.
- Trigger alerts when animals cross predefined thresholds.

# CHAPTER 3

# SYSTEM REQUIREMENTS

## 3.1 HARDWARE REQUIREMENTS:

- **ESP32 Microcontroller:** The ESP32 is a low-cost, low-power microcontroller with integrated Wi-Fi and Bluetooth capabilities, making it ideal for IoT applications. Developed by Espressif Systems, it features a dual-core Tensilica LX6 processor with clock speeds up to 240 MHz. The ESP32 offers multiple GPIO pins, ADCs, DACs, UART, SPI, I2C, and PWM interfaces, allowing it to connect with various sensors and modules. It supports deep sleep mode, which helps in reducing power consumption for battery-powered projects. With its versatility and strong community support, the ESP32 is widely used in smart home, automation, and remote monitoring systems.
- **GSM Module (SIM900L):** The SIM900 GSM module is a quad-band GSM/GPRS module developed by SIM Com, used for wireless communication in embedded systems. It supports voice calls, SMS, and internet data transmission over 2G networks. The module operates on frequencies 850, 900, 1800, and 1900 MHz, making it compatible worldwide. It communicates with microcontrollers using AT commands over UART, making integration simple and efficient. The SIM900 is widely used in IoT projects for remote monitoring, tracking, and automation applications**.**
- **Ultrasonic Sensors (HC-SR04):** The HC-SR04 is a popular ultrasonic distance sensor used for measuring the distance to objects using sound waves. It works by emitting an ultrasonic pulse and measuring the time it takes for the echo to return after bouncing off an object. The sensor has a range of 2 cm to 400 cm with good accuracy, making it suitable for obstacle detection and distance measurement. It operates with 5V power and communicates using simple trigger and echo pins. Widely used in robotics and automation, the HC-SR04 is easy to interface with microcontrollers like the ESP32.
- **LCD Module (16x2):** The 16x2 LCD module is a widely used alphanumeric display that can show 16 characters per line on two lines. It operates using the HD44780 controller and can be interfaced with microcontrollers in 4-bit or 8-bit mode. The module supports both numbers and

letters, making it ideal for displaying sensor readings, status messages, and user interfaces. It typically operates at 5V and includes a backlight for visibility in low-light conditions. Easy to program and cost-effective, the 16x2 LCD is commonly used in embedded and IoT projects.

- **Motor Driver Module (L293D):** The L293D is a popular motor driver IC used to control the direction and speed of DC motors and stepper motors in embedded systems. It is a dual H-bridge driver, meaning it can control two motors independently in both forward and reverse directions. The IC operates at 5V logic and can drive motors with voltages up to 36V and current up to 600mA per channel. It receives control signals from a microcontroller and translates them into high-power signals suitable for motor operation. Compact and efficient, the L293D is widely used in robotics and automation projects.

## 3.2 SOFTWARE REQUIREMENTS:

- **Arduino IDE:** The Arduino IDE (Integrated Development Environment) is an open-source software used for writing, compiling, and uploading code to Arduino and compatible microcontroller boards like the ESP32. It supports C and C++ programming languages with a simple and user-friendly interface, making it ideal for beginners and professionals. The IDE includes a library manager and pre-built functions to easily control sensors, modules, and hardware components. It communicates with boards via USB and uses a serial monitor for debugging and data display. Widely used in embedded and IoT development, Arduino IDE simplifies the prototyping and programming process.


Figure 3.2 Arduino IDE

# CHAPTER 4

# HARDWARE DESCRIPTION

## 4.1 ESP32 MICROCONTROLLER:

The ESP32 is a very versatile System On a Chip (SoC) that can be used as a general purpose microcontroller with quite an extensive set of peripherals including Wifi and Bluetooth wireless capabilities. It is manufactured by Shanghai-based Espressif Systems, and costs less than $5. Although the ESP32 is a SoC, most users will not start by using just the ESP32 chip itself.

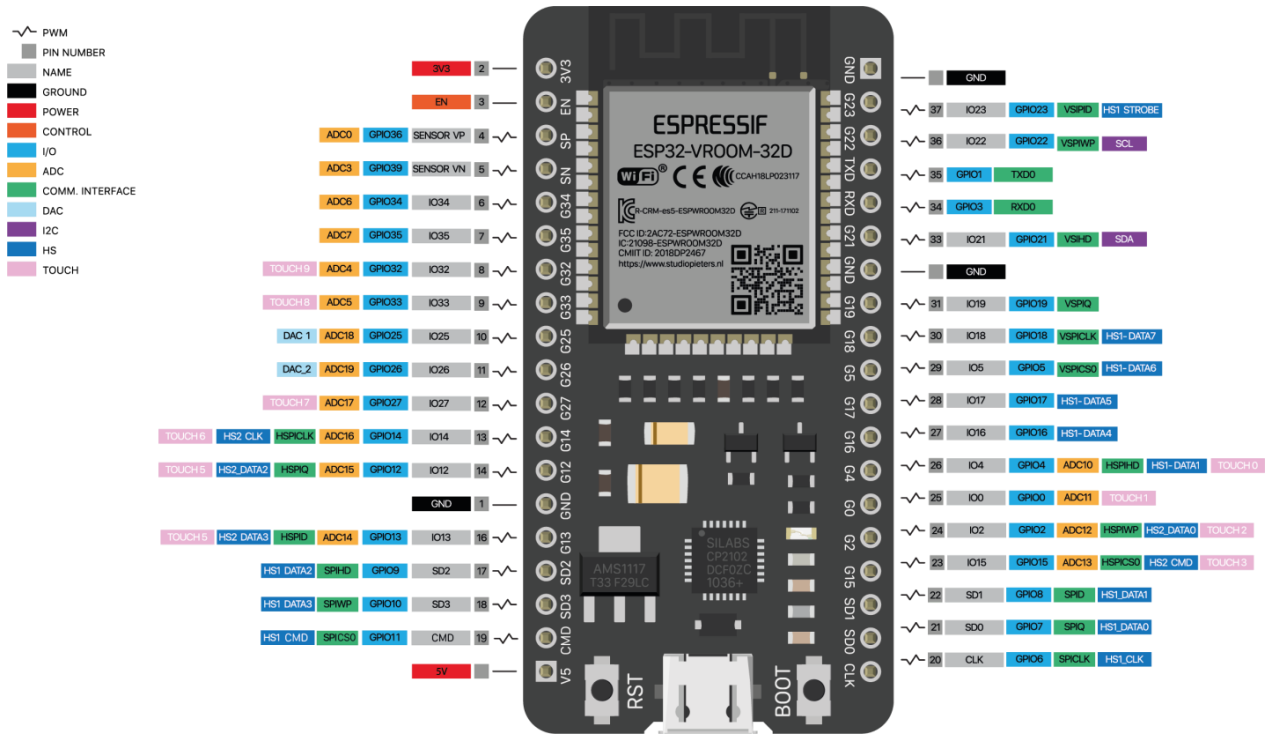



**Fig 4.1(a) ESP32- microcontroller**

While it is possible to design a product using the ESP32 SoC, this is not a common approach. Instead, most ESP32-based designs use pre-made modules that consist of an actual ESP-32 SoC, external flash memory, and a crystal and pre-tuned PCB antenna or an IPEX antenna connector.

The whole assembly is then placed under a shielded can (figure 2). This module is made by Espressif itself, and this link shows several versions.

One big advantage to using this module instead of designing from scratch is that Espressif has already pre-loaded the low-level device drivers, the wireless protocol stacks for Wifi b, g, n, Bluetooth and BLE, and Free RTOS as the base OS. In addition, a bootloader has also been loaded to allow for relatively easy downloading of user applications. Another module commonly referred to as an ESP32 is what is more appropriately called an ESP32 Development Module. This is basically an ESP32 module mounted on a board with additional support circuitry such as a voltage regulator and a serial to USB IC.

It allows direct connection to a desktop PC that can then be used to compile, download, and run programs directly on this module. Figure 3 shows two such development modules from different manufacturers.

Note that one has more of the pins of the ESP module available than the other one, and is slightly more expensive. Otherwise, they are very similar. They each allow a direct connection to a desktop development system through a USB cable.

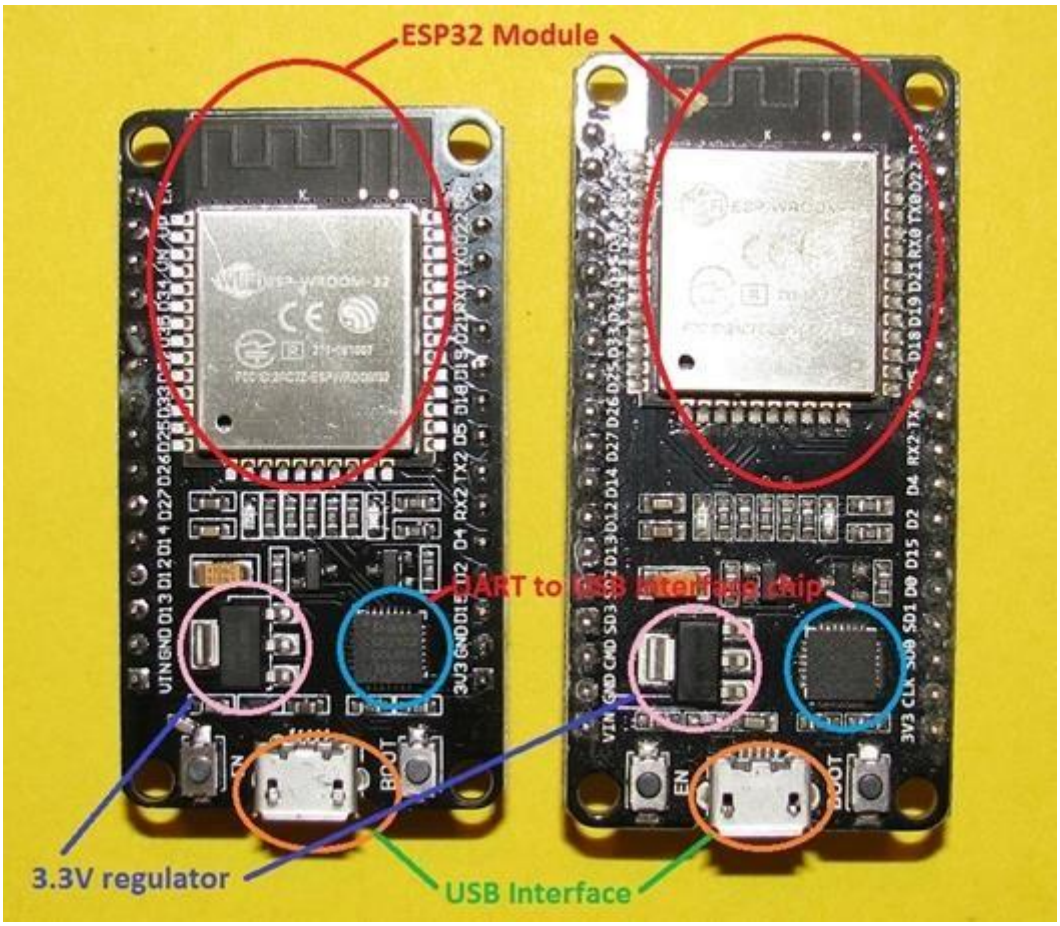



**Figure 4.1(b) ESP32- microcontroller components**

The recommended approach is to use the Development Board for Proof of Concept designs since it is fully self-contained. Then, switch to the more compact ESP32 module when the application is more fully developed, and the entire hardware design is ready to be integrated.

The application code can be downloaded to the ESP32 Module using an adapter board. It essentially provides all the functionality of the ESP32 Development Module. But the target is the ESP32 Module without the actual programming interface that is part of the Development Board.

## Wireless connectivity:

**Wi-Fi:** 802.11 b/g/n/e/i (802.11n @ 2.4 GHz up to 150 Mbit/s) **Bluetooth:** v4.2 BR/EDR and Bluetooth Low Energy (BLE) **Memory:**

**ROM:** 448 KB – For booting and core functions

**SRAM:** 520 KB – For data and instruction

**RTC fast SRAM:** 8 KB – For data storage and main CPU during RTC Boot from the deep-sleep mode

**RTC slow SRAM:** 8 KB – For co-processor accessing during deep-sleep mode

**Fuse:** 1KBit – Of which 256 bits are used for the system (MAC address and chip configuration) and the remaining 768 bits are reserved for customer applications, including Flash-Encryption and Chip-ID **Embedded flash:**

- 0 MB (ESP32-D0WDQ6, ESP32-D0WD, and ESP32-S0WD chips)
- 2 MB (ESP32-D2WD chip)
- 4 MB (ESP32-PICO-D4 SiP module)
- Flash memory is connected internally via IO16, IO17, SD_CMD, SD_CLK, SD_DATA_0 and SD_DATA_1 on ESP32-D2WD and ESP32-PICO-D4.
- External flash & SRAM: ESP32 supports up to four 16 MB external QSPI flashes and SRAMs with hardware encryption based on AES to protect developers' programs and data. ESP32 can access the external QSPI flash and SRAM through high-speed caches.
- Up to 16 MB of external flash are memory-mapped onto the CPU code space, supporting 8-bit, 16-bit and 32-bit access. Code execution from flash is supported.
- Up to 8 MB of external flash/SRAM memory are mapped onto the CPU data space, supporting 8-bit, 16-bit and 32-bit access. Data-read is supported on the flash and SRAM. Data-write is supported on the SRAM.

- Note that ESP32 chips with embedded flash do not support the address mapping between external flash and peripherals.

**Peripheral input/output:** The ESP32 offers a rich peripheral interface with DMA that includes:

- Capacitive touch
- ADCs (analog-to-digital converter)
- DACs (digital-to-analog converter)
- I²C (Inter-Integrated Circuit)
- UART (universal asynchronous receiver/transmitter)
- CAN 2.0 (Controller Area Network)
- SPI (Serial Peripheral Interface)
- I²S (Integrated Inter-IC Sound)
- RMII (Reduced Media-Independent Interface)
- PWM (pulse width modulation), and more.

**Security:**

- IEEE 802.11 standard security features all supported, including WFA, WPA/WPA2 and WAPI Secure boot
- Flash encryption
- 1024-bit OTP, up to 768-bit for customers
- Cryptographic hardware acceleration: AES, SHA-2, RSA, elliptic curve cryptography (ECC), random number generator (RNG)
- Developing applications for the ESP32
- The usual way to develop any embedded system is to first choose the proper microcontroller, or microcontroller module, that fits the desired hardware requirements and, just as importantly, has the proper software development support.
- A prototype hardware platform to test the application code is developed. Then, the application software development process can begin.
- Assuming that the ESP32 has been chosen as the microcontroller module, the next step is to actually set up an environment where application code can be developed and tested.
- Developing an application for an embedded system is an iterative process that usually requires a setup on a cross-development platform whereby the code can be written, complied, linked, and loaded into the processor.

- After hardware testing, the whole process is repeated until you achieve the final performance requirements.
- The entire process is usually carried out in an Integrated Development Environment (IDE) that, at a minimum, should provide the following: A text editor to write the application code; a compiler/linker/locater; and a loader to download the compiled binary code to the proper physical address segments in the target processor.

## 4.2 HC-SR04 ULTRASONIC SENSOR:


**Figure 4.2** *Diagram of the basic ultrasonic sensor operation*

### 4.2.1 HC-SR04 Ultrasonic Sensor – Overview

1. What is an ultrasonic?

- ULTRA = BEYOND
- SONIC = SOUND The sound beyond human hearing range(20000Hz) is known as ultrasonic.

2. What is Ultrasonic sensor?

- Ultrasonic sensors are sensors that convert ultrasound waves to electrical signals or vice versa.

An Ultrasonic sensor is a device that can measure the distance to an object by using sound waves. It measures distance by sending out a sound wave at a specific frequency and listening for that sound wave

to bounce back. By recording the elapsed time between the sound wave being generated and the sound wave bouncing back, it is possible to calculate the distance between the sonar sensor and the object.

The HC-SR04 ultrasonic sensor uses sonar to determine distance to an object like bats do. It offers excellent non-contact range detection with high accuracy and stable readings in an easy-to-use package. From 2cm to 400 cm or 1" to 13 feet. Its operation is not affected by sunlight or black material like sharp rangefinders are (although acoustically soft materials like cloth can be difficult to detect). It comes complete with ultrasonic transmitter and receiver module. Since it is known that sound travels through air at about 344 m/s (1129 ft/s), you can take the time for the sound wave to return and multiply it by 344 meters (or 1129 feet) to find the total round-trip distance of the sound wave. Round-trip means that the sound wave traveled 2 times the distance to the object. before it was detected by the sensor; it includes the 'trip' from the sonar sensor to the object AND the 'trip' from the object to the Ultrasonic sensor (after the sound wave bounced off the object). To find the distance to the object, simply divide the round-trip distance in half.

$$distance = \frac{speed\ of\ sound \times time\ taken}{2}$$

### 4.2.2 HC-SR04 Ultrasonic Sensor – Features

1. Power Supply: +5V DC
2. Quiescent Current: <2mA
3. Working Current: 15mA
4. Effectual Angle: <15°
5. Ranging Distance: 2cm – 400 cm/1″ – 13ft
6. Resolution: 0.3 cm
7. Measuring Angle: 30 degrees
8. Trigger Input Pulse width: 10uS


**Figure 4.2.2** *Ultrasonic sensor*

9. Dimension: 45mm x 20mm x 15mm

### 4.2.3 HC-SR04 Ultrasonic Sensor Pin Configuration

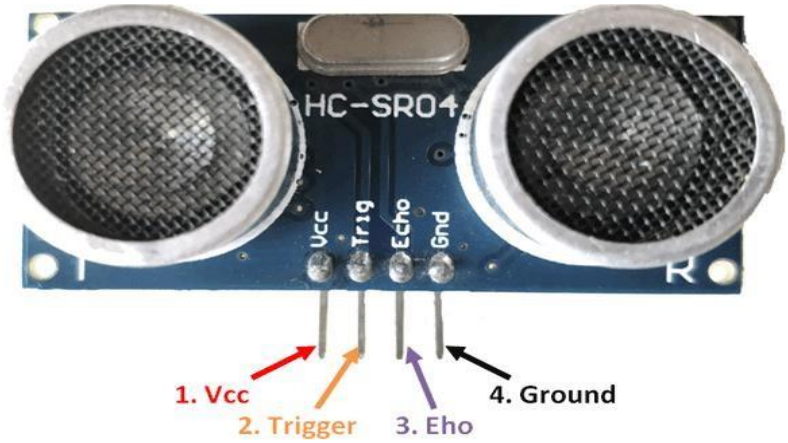



**Figure 4.2.3** *Ultrasonic Sensor Pins*

| Pin Number | Pin Name | Description |
|---|---|---|
| 1 | VCC | The VCC pin powers the sensor, typically with +5V |
| 2 | Trigger | Trigger pin is an Input pin. This pin has to be kept high for 10us to initialize measurement by sending US wave. |
| 3 | Echo | Echo pin is an Output pin. This pin goes high for a period of time which will be equal to the time taken for the US wave to return back to the sensor. |
| 4 | Ground | This pin is connected to the Ground of the system. |

**Table 4.2.3***: HC-SR04 Ultrasonic Sensor Pin Configuration*

## 4.2.4 HC-SR04 Ultrasonic Sensor – Work

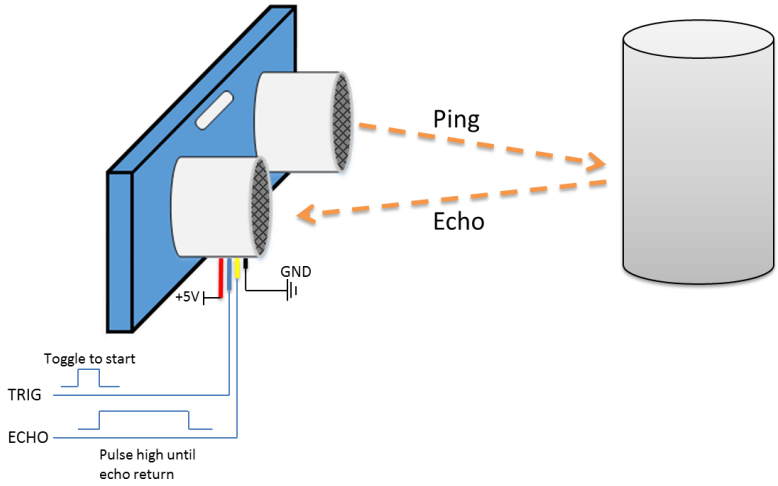



**Figure 4.2.4 (a)** *Block Diagram of Ultrasonic Sensor work*

The ultrasonic sensor uses sonar to determine the distance to an object. Here's what happens:

1. the transmitter (trig pin) sends a signal: a high-frequency sound
2. when the signal finds an object, it is reflected and
3. the transmitter (echo pin) receives it.

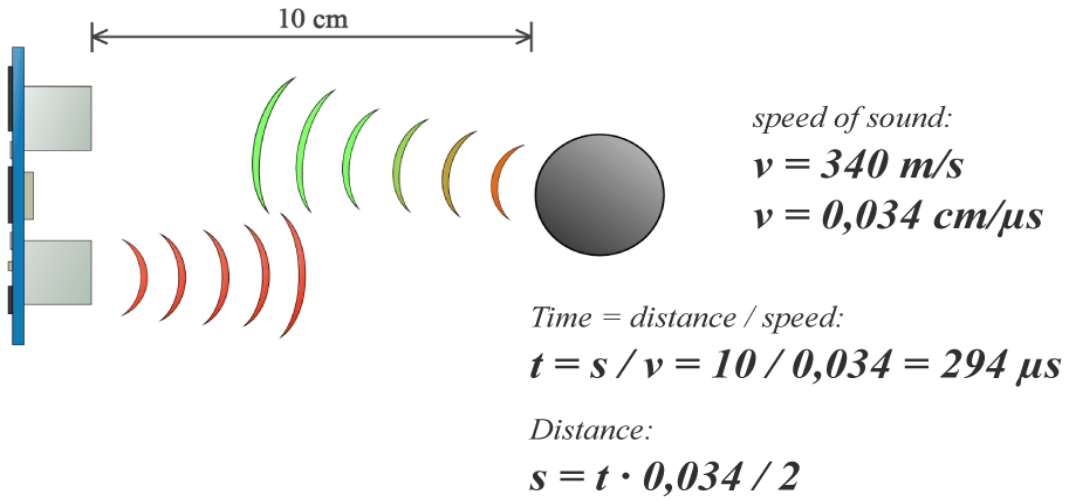



**Figure 4.2.4(b)**

For example, if the object is 10 cm away from the sensor, and the speed of the sound is 340 m/s or

0.034 cm/µs the sound wave will need to travel about 294 u seconds. But what we will get from the Echo pin will be double that number because the sound wave needs to travel forward and bounce backward. So, in order to get the distance in cm we need to multiply the received travel time value from the echo pin by 0.034 and divide it by 2 as shown above.

### 4.2.5 HC-SR04 Ultrasonic Sensor - Applications

1. Used to avoid and detect obstacles with robots like biped robot, obstacle avoider robot, path finding robot etc.
2. Used to measure the distance within a wide range of 2cm to 400cm
3. Can be used to map the objects surrounding the sensor by rotating it.
4. Depth of certain places like wells, pits etc. can be measured since the waves can penetrate through water.

## 4.3 30 RPM DC MOTOR:

A 30 rpm DC motor is a direct current motor with a rotational speed of approximately 30 revolutions per minute (RPM) at its rated voltage. These motors often incorporate a gearbox to reduce the high-speed output of the motor to this lower, more usable speed while significantly increasing the output torque.

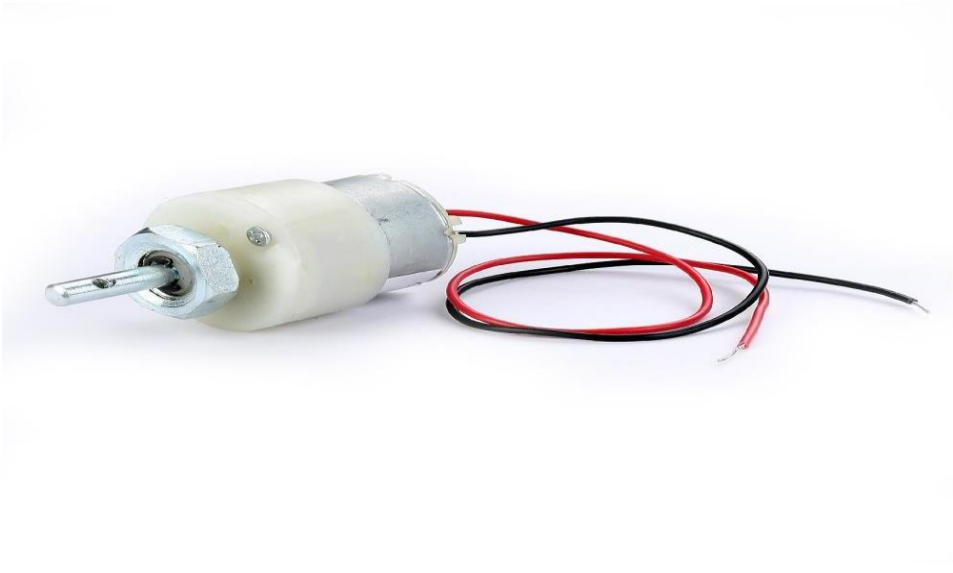


**Figure 4.3:** *30 RPM DC motor*

Here's a breakdown of what you should know about them:

**Key Characteristics:**

- **Speed:** The primary characteristic is its output speed of around 30 RPM. This makes it suitable for applications requiring slow and controlled movement.
- **Voltage:** They are available in various voltage ratings, commonly 6V, 12V, or 24V DC. You need to ensure your power supply matches the motor's rating.
- **Torque:** Due to the integrated gearbox, these motors provide significantly higher torque compared to a direct-drive DC motor running at 30 RPM. The exact torque depends on the gear ratio. Torque is crucial for applications involving load-bearing or requiring significant force.
- **Gearbox:** The gearbox is a critical component. It typically consists of a series of gears (spur, planetary, or worm) that reduce the motor's speed and amplify its torque. The gearbox material can be plastic or metal, with metal gears generally offering higher durability and load-bearing capacity.
- **Shaft:** The output shaft is what you connect to your application (e.g., wheels, levers). Common shaft types include plain, keyed, or D-shaped. Some geared motors also feature threaded shafts or internal holes for easier connections.
- **Size and Mounting:** 30 RPM DC gear motors come in various sizes and with different mounting options (brackets, flanges). Consider the space constraints and mounting requirements of your project.

### 4.3.1 Typical Specifications (can vary between models):

Based on the search results, here are some common specifications you might find for a 30 RPM DC motor:

- **Operating Voltage:** 6V - 12V DC (some can go up to 18V)
- **Rated Speed:** Approximately 30 RPM (at the specified voltage, e.g., 12V)
- **Torque:** Ranges from 5 kg-cm to 15 kg-cm or even higher (stall torque can be significantly higher, e.g., 18 kg-cm to 38 kg-cm)
- **No-load Current:** Typically in the range of 60 mA (max)
- **Load Current:** Can go up to 300 mA (max) or more depending on the load
- **Shaft Diameter:** Commonly 6mm with a possible internal hole or threading (e.g., M3 thread)

- **Gearbox Type:** Spur gears are common, and the material can be plastic or metal.
- **Motor Dimensions:** Varies depending on the model.
- **Weight:** Around 100 grams is typical for some smaller models.

### 4.3.2 Applications:

30 RPM DC gear motors are well-suited for applications requiring controlled, slow movement with significant force. Some common examples include:

- **Robotics:**
  - Driving wheels or tracks on robots.
  - Actuating robotic arms or other mechanisms requiring controlled speeds.
- **Automation:**
  - Small conveyor belts.
  - Positioning systems.
  - Automated door openers.
  - Turntables.
- **DIY Projects:**
  - Hobbyist robots and vehicles.
  - Model railroads.
  - Custom mechanical projects requiring slow and powerful motion.
- **Home Appliances:**
  - Some types of electric door locks or openers.
- **Industrial Applications:**
  - Vending machines.
  - Cleaning equipment.
  - Some types of actuators in machinery.
- **Voltage Variation:** Changing the input voltage will directly affect the motor's speed. Lower voltage results in lower speed.
- **Pulse Width Modulation (PWM):** This is a common and efficient method for speed control. By varying the duty cycle of a pulsed DC signal, you can effectively control the average voltage applied to the motor and thus its speed. You'll typically need a motor driver circuit (like an L298N) and a microcontroller (like an Arduino) to implement PWM control.

## 4.4 SIM900A GSM MODULE:

SIM900A GSM Module is the smallest and cheapest module for GPRS/GSM communication. It is common with Arduino and microcontroller in most of embedded application. The module offers GPRS/GSM technology for communication with the uses of a mobile sim. It uses a 900 and 1800MHz frequency band and allows users to receive/send mobile calls and SMS. The keypad and display interface allows the developers to make the customize application with it. Furthermore, it also has modes, command mode and data mode. In every country the GPRS/GSM and different protocols/frequencies to operate. Command mode helps the developers to change the default setting according to their requirements.

The SIM900A is integrated with the TCP/IP protocol; extended TCP/IP AT commands are developed to use the TCP/IP protocol easily, which is very useful for data transfer applications. It can communicate with controllers via AT commands (GSM 07.07,07.05, and SIMCOM enhanced AT Commands).

**Figure 4.4:** *SIM900A GSM Module*

### 4.4.1 SIM900A Pin Configuration

The Module SIM900A looks like a single chip but it has a bunch of features that can help to build almost many commercial applications. Although, there are a total of 68 pins on SIM900A and using these pins helps to build the applications.

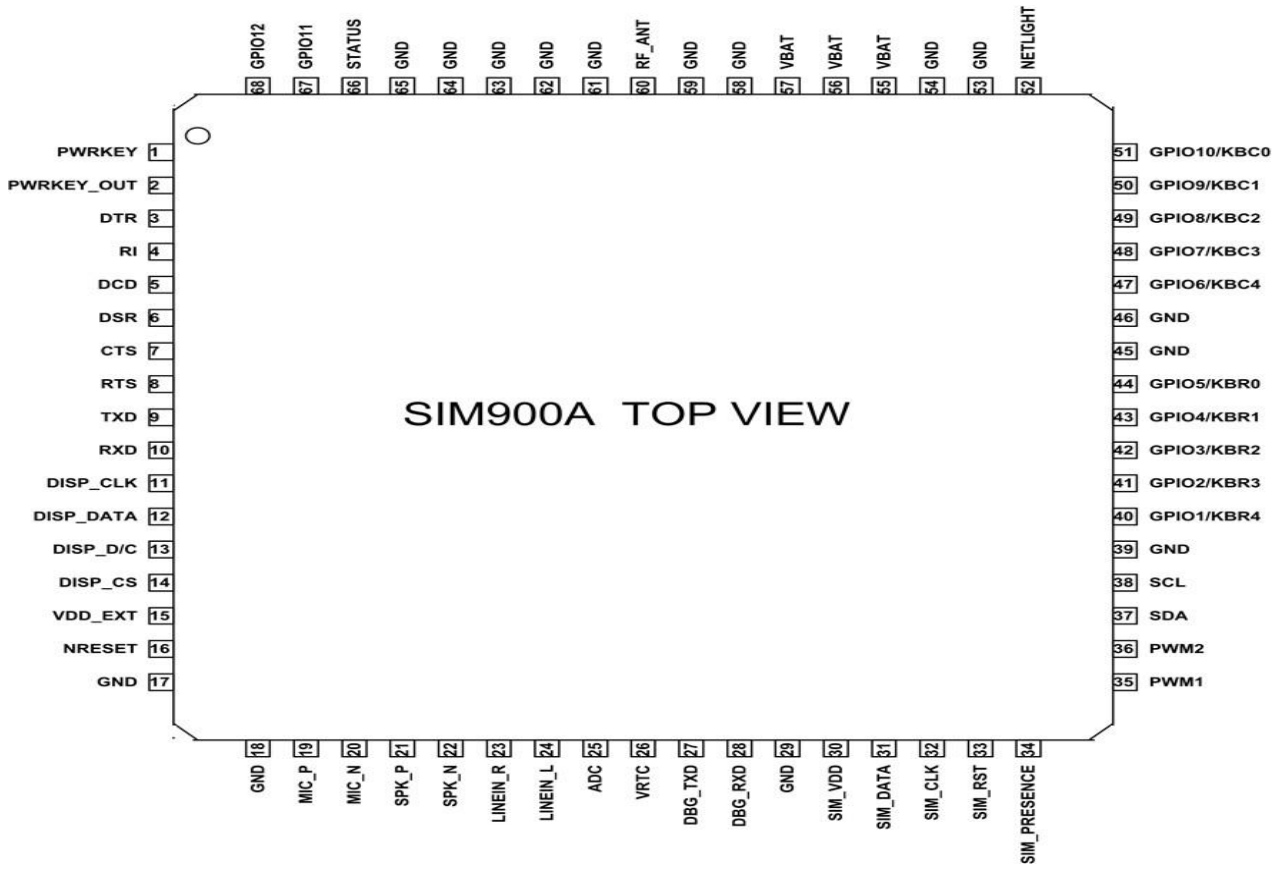



**Figure 4.4.1** *SIM900A Pin Configuration*

**SIM900A GSM Module Pin Configuration Description:**

**GPIO Pins**

The GPIO pins help to perform the simple and advance I/O function. All pins give the maximum output equal to the power supply which is useable to control most of the devices like sensors and other modules. All GPIO pins in SIM900A are:

- GPIO1 – Pin40
- GPIO2 – Pin41
- GPIO3 – Pin42

- GPIO4 – Pin43
- GPIO5 – Pin44
- GPIO6 – Pin47
- GPIO7 – Pin48
- GPIO8 – Pin49
- GPIO9 – Pin50
- GPIO10 – Pin51
- GPIO11 – Pin67
- GPIO12 – Pin68

**Status Pins**

The module has two status pins which help to indicate two different kinds of status. The first one is the working status of the module and the second for communication status. Net status means either the module is connecting to the network or other network functions, etc. Both these pins can't operate LED directly. They always act with a combination of a transistor.

- STATUS – Pin52
- NIGHTLIGHT – Pin66

**SIM900A Display Interface Pins**

The device offers a 4 pin display interface with itself. The display isn't necessary, it is only in case of requirement. The use of interface helps to get the visualization with the module and make it an application. All display pins are:

- DISP_DATA – Pin12 – For Display Data
- DISP_CLK – Pin11 – For Clock Input
- DISP_CS – Pin14 – To enable the display
- DISP_D/C – Pin13 – To select between data and command

**I2C Pins**

SIM900A has multiple kinds of communication and I2C is one of them due to is popularity. The module has a single I2C protocol pin, which helps to build the application with any module with that communication.

- SCL – Pin38
- SDA – Pin37

SDA for data and SCL for clock pulse.

**SIM900A GSM Module Keypad interface Pins**

The two-pin keypad is interfaceable with the module. The module will take the keypad data as a 2D matrix value from the KCB pins for each value. The keypad interface pins in the module are:

- KBR0~KBR4 (ROWS) – Pin40~Pin44
- KBC0~KBC4 (COLUMN) – Pin47~Pin51

**Serial Port**

The UART serial interface uses the two pins for proper data communication, which are RX and TX. Both pins have no independence on any other pins or modules. In SIM900A these pins are available but it also has some other pins for status/indication of data. By combining these pins, the serial port helps to generate the RS-232 connector too. All the serial pins are:

- RXD – Pin10 – To receive the data
- TXD – Pin 9- To send the data
- RTS – Pin8 – To send the request of data transmission
- CTS – Pin7 – To clear the send request
- RI – Pin4 – Ring indicator
- DSR – Pin6 – To indicate that data set ready
- DCD – Pin5 – To indicate data carry detect
- DTR – Pin3 – To indicate data terminal ready

**Debug Interface**

Debugging helps the developers to debug the module and update its firmware. In this module, there are sperate serial interface pins for debugging. Both pins are:

- DBG_TXD – Pin27 – For Data Transmission
- DBG_RXD – Pin28 – For Data receiving

**SIM Interface**

As we know that module SIM900A is a GPRS/GSM module. The module is dependent on some devices for some of its features. The most important one is the SIM. The SIM needs to connect with the module for GPRS/GSM functions to fully operate. All the sim interface of the module is:

- SIM_VDD – Pin30 – Power Supply of the SIM
- SIM_DATA – Pin31 – For data output
- SIM_CLK – Pin32 – For clock pulse
- SIM_RST – Pin33 – For reset

- SIM_PRESENCE – Pin34 – To detect the SIM

**SIM900A Analog to Digital converter Pins**

The module has only a single pin to detect and convert the analog signal to digital for SIM900A. The voltage range on the ADC pin is from 0 to 3 only.

- ADC – Pin25

**PWM Pins**

The PWM is mostly in microcontrollers for industrial applications but due to IoT, the module offers two PWM pins which helps to make the IoT and PWM based device without using any third interface.

- PWM1 – Pin35
- PWM2 – Pin36

**Audio Interface**

The audio interface will help to connect the mic and speaker with SIM900A. The connection of Line, Audio and Speaker will help to make the calls through the modules.

- MIC_P – Pin19
- MIC_N – Pin20
- SPK_P – Pin21
- SPK_N – Pin22
- LINEIN_R – Pin23
- LINE_L – Pin24

**Control Pin**

There is power on pins on the device, which helps to turn it on using external signals. There is two power on pins. The first one is PWRKEY which requires a LOW signal to power on/off the system. To do that, the pins require an input signal for a little bit long time. The second pin is PWRKEY_OUT, which gets short with the PWRKEY pin and turn on/off the device.

- PWRKEY – Pin1
- PWRKEY_OUT – Pin2

**Reset pins**

The device has an external LOW input signal reset pin to reset the device with the use of an external signal.

- NRESET – Pin16

**SIM900A GSM Module RF Antenna**

To extend the range of the SIM900A the antenna pin needs to connect with an external wire. The official antenna is also available for the module.

- RF_ANT – Pin60

**Power Pins**

The module SIM900A has multiple types of power pin. Some works as input and some as output. The most important one to understand is VRTC, which acts as a backup for the internal RTC of the device. All power and ground pins of the module are:

- VBAT(Input) – Pin55, Pin56, Pin57
- VRTC (Input/Output) – Pin26
- VDD_EXT(OUTPUT) – Pin15
- GND – Pin17, Pin18, Pin29, Pin39, Pin45, Pin46, Pin53, Pin54, Pin58, Pin59, Pin61, Pin62, Pin63, Pin64, Pin65

### 4.4.2 SIM900A GSM Module Block Diagram

The following diagram is describing the SIM900A internal structure of the module.

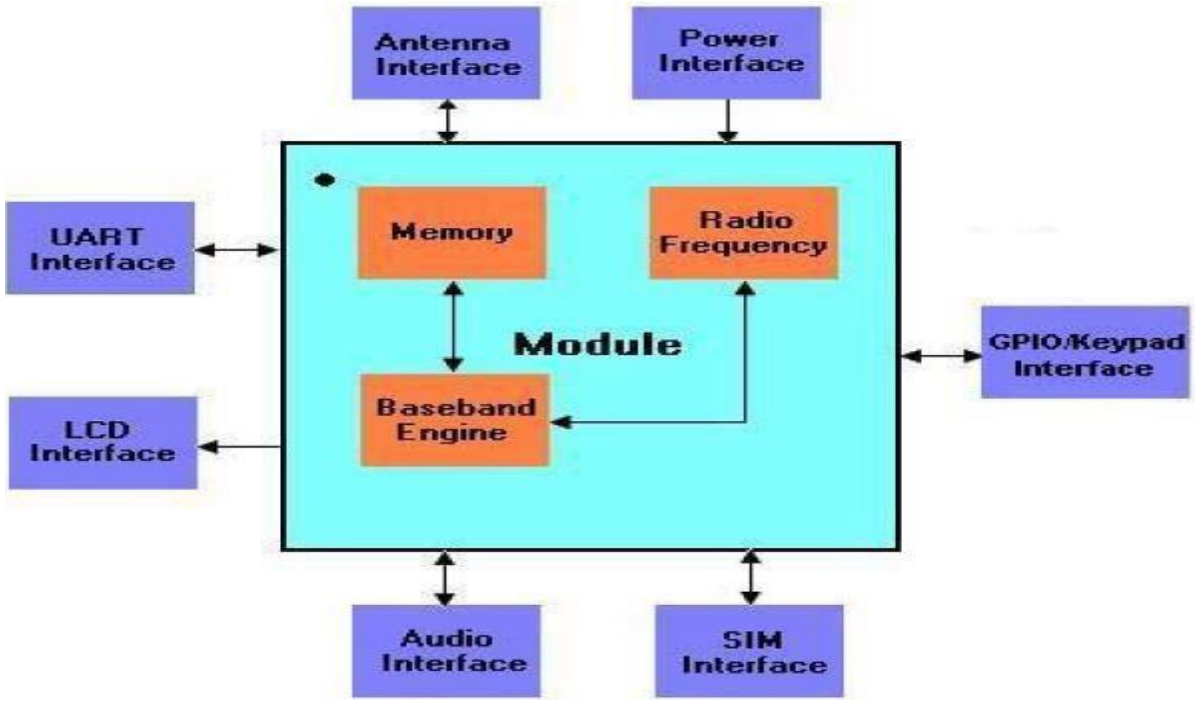



**Figure 4.4.2** *SIM900A GSM Module Block Diagram*

### 4.4.3 SIM900A GSM Module Main Features:

| FEATURES | DETAIL |
|---|---|
| Power Input | 3.4V to 4.5V |
| Operating Frequency | EGSM900 and DCS1800 |
| Transmitting Power Range | 2V for EGSM900 and 1W for DCS1800 |
| Data Transfer Link | Download: 85.6kbps, Upload:42.8kbps |
| SMS | MT, MO, CB, Text and PDU mode. |
| Antenna Support | Available |
| Audio Input/output | Available |
| Serial Port | I2C and UART |
| Serial Debug Port | Available |

**Figure 4.4.3** *SIM900A GSM Module Main Features*

**Applications**

- The module is the best application to design a graphic for Voice call and SMS application.
- Some IoT applications, mostly in an emergency have the module.
- The location trancing system also uses SIM900A.
- SIM900A can use for mobile communication.

## 4.5 PIR SENSOR:

A passive infrared sensor is an electronic sensor that measures infrared light radiating from objects. PIR sensors mostly used in PIR-based motion detectors. Also, it used in security alarms and automatic lighting

applications. The below image shows a typical pin configuration of the PIR sensor, which is quite simple to understand the pinouts. The PIR sensor consist of 3 pins:

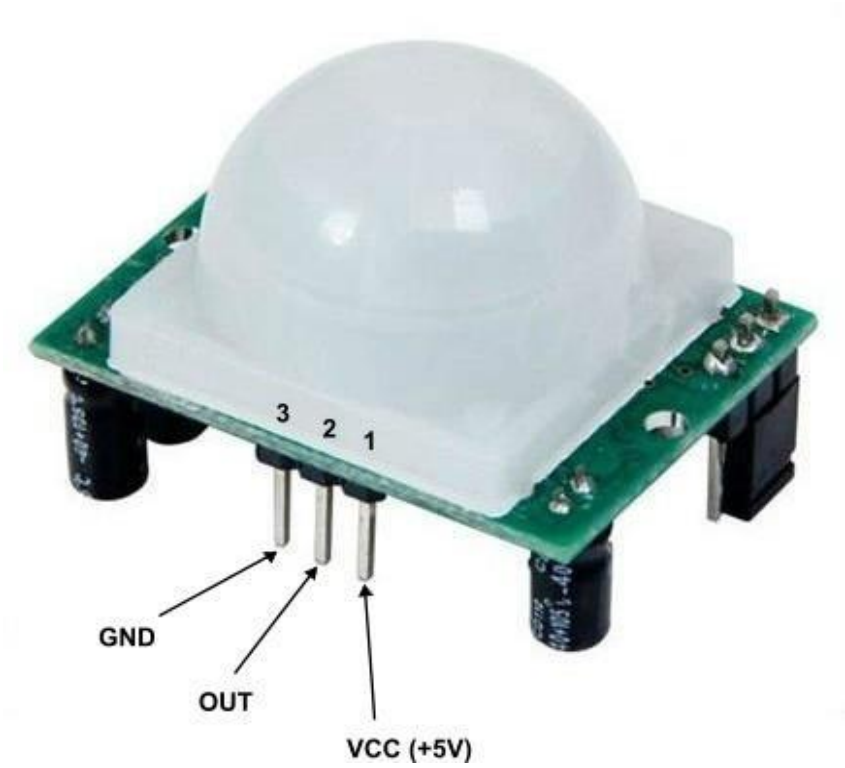



**Figure 4.5 (a)** *PIR Sensor*

- Pin1 corresponds to the drain terminal of the device, which connected to the positive supply 5V DC.
- Pin2 corresponds to the source terminal of the device, which connects to the ground terminal via a 100K or 47K resistor. The Pin2 is the output pin of the sensor. The pin 2 of the sensor carries the detected IR signal to an amplifier from the
- Pin3 of the sensor connected to the ground.

**What does a PIR Sensor detect?**

Generally, PIR sensor can detect animal/human movement in a requirement range. PIR is made of a pyroelectric sensor, which is able to detect different levels of infrared radiation. The detector itself does not emit any energy but passively receives it.

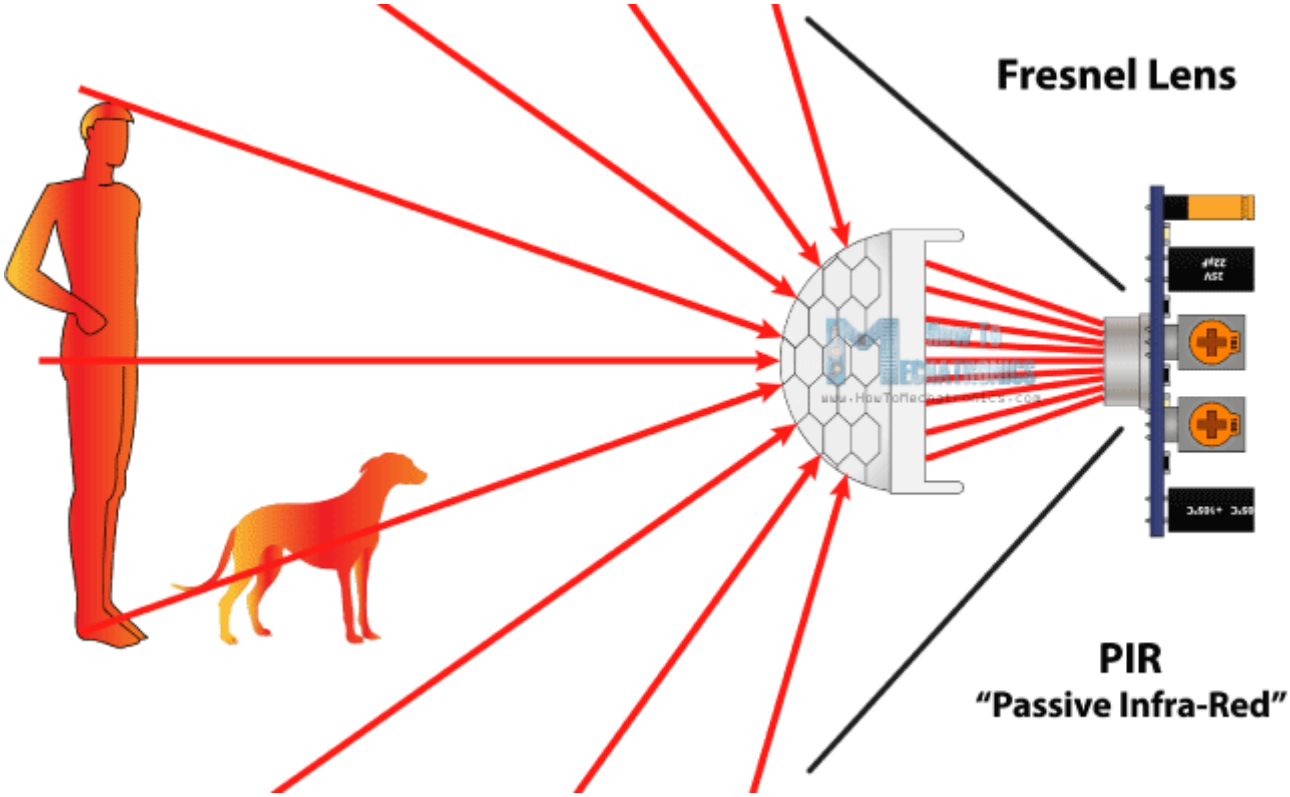



**Figure 4.5 (b)** *PIR Sensor*

It detects infrared radiation from the environment. Once there is infrared radiation from the human body particle with temperature, focusing on the optical system causes the pyroelectric device to generate a sudden electrical signal. Simply, when a human body or any animal passes by, then it intercepts the first slot of the PIR sensor. This causes a positive differential change between the two bisects. When a human body leaves the sensing area, the sensor generates a negative differential change between the two bisects.

### 4.5.1 PIR Sensor Working Principle :

The passive infrared sensor does not radiate energy to space. It receives the infrared radiation from the human body to make an alarm. Any object with temperature is constantly radiating infrared rays to the outside world. The surface temperature of the human body is between 36° C - 27 ° C and most of its radiant energy concentrated in the wavelength range of 8 um-12 um.

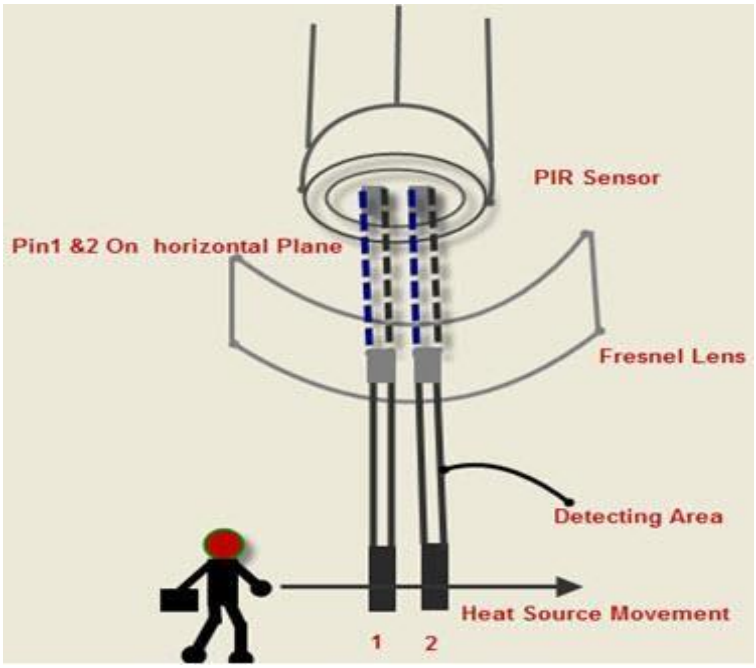



**Figure 4.5.1(a)** *PIR Sensor Working Principle*

Passive infrared alarms classified into infrared detectors (infrared probes) and alarm control sections. The most widely used infrared detector is a pyroelectric detector. It uses as a sensor for converting human infrared radiation into electricity. If the human infrared radiation is directly irradiated on the detector, it will, of course, cause a temperature change to output a signal. But in doing all this, the detection distance will not be more. In order to lengthen the detection distance of the detector, an optical system must be added to collect the infrared radiation. Usually, plastic optical reflection system or plastic Fresnel lens used as a focusing system for infrared radiation.

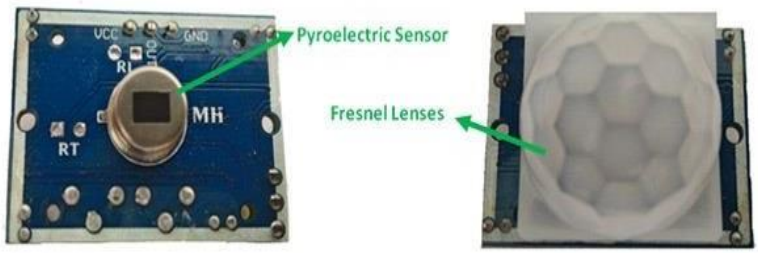



**Figure 4.5.1(b)** *PIR sensor*

In the detection area, the lens of the detector receives the infrared radiation energy of the human body through the clothing and focused on the pyroelectric sensor. When the human body moves in this

surveillance mode, it enters a certain field of view in sequence and then walks out of the field of view. The pyroelectric sensor sees the moving human body for a while and then does not see it, so the infrared radiation of human body constantly changes the temperature of the pyroelectric material. So that it outputs a corresponding signal, which is the alarm signal.

**Range of PIR Sensor:**

- Indoor passive infrared: Detection distances range from 25 cm to 20 m.
- Indoor curtain type: The detection distance ranges from 25 cm to 20 m.
- Outdoor passive infrared: The detection distance ranges from 10 meters to 150 meters.
- Outdoor passive infrared curtain detector: distance from 10 meters to 150 meters

## 4.6 L293D MOTOR DRIVER MODULE :

L293D is a typical Motor driver or Motor Driver IC which allows DC motor to drive on either direction. L293D is a 16-pin IC which can control a set of two DC motors simultaneously in any direction. It means that you can control two DC motor with a single L293D IC. Dual H-bridge Motor Driver integrated circuit (IC).

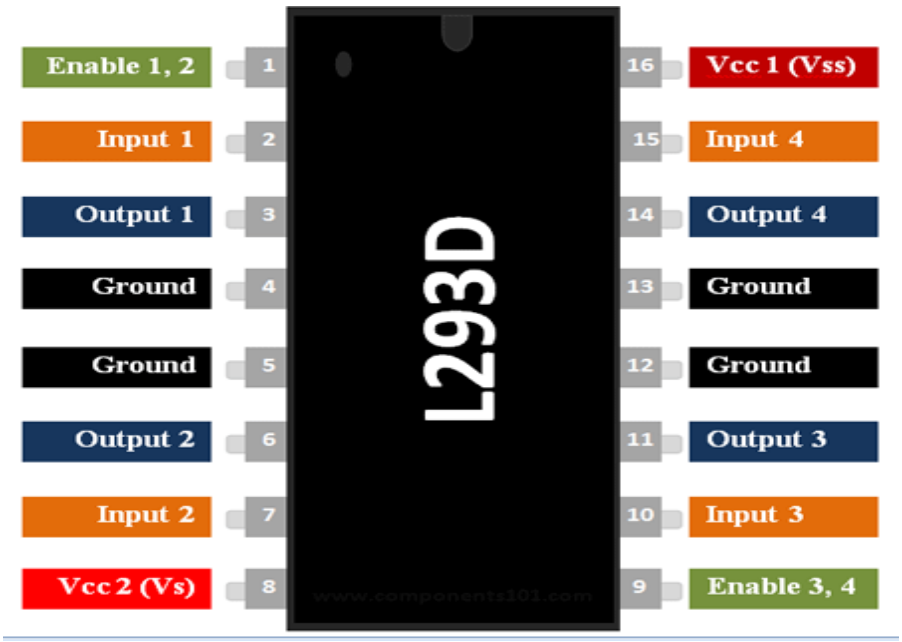



**Figure 4.6** *Motor Driver IC L293D Pinout*

**Features:**

- Can be used to run Two DC motors with the same IC.
- Speed and Direction control is possible

- Motor voltage Vcc2 (Vs): 4.5V to 36V
- Maximum Peak motor current: 1.2A
- Maximum Continuous Motor Current: 600mA
- Supply Voltage to Vcc1(vss): 4.5V to 7V
- Transition time: 300ns (at 5Vand 24V)
- Automatic Thermal shutdown is available
- Available in 16-pin DIP, TSSOP, SOIC packages

### 4.6.1 L293D IC Usage:

The L293D is a popular 16-Pin Motor Driver IC. As the name suggests it is mainly used to drive motors. A single L293D IC is capable of running two DC motors at the same time; also the direction of these two motors can be controlled independently. So if you have motors which has operating voltage less than 36V and operating current less than 600mA, which are to be controlled by digital circuits like Op-Amp, 555 timers, digital gates or even Microcontrollers like Arduino, PIC, ARM etc.. this IC will be the right choice for you. Using this L293D motor driver IC is very simple. The IC works on the principle of Half H- Bridge, let us not go too deep into what H-Bridge means, but for now just know that H bridge is a set up which is used to run motors both in clock wise and anticlockwise direction. As said earlier this IC is capable of running two motors at the any direction at the same time, the circuit to achieve the same is shown below.

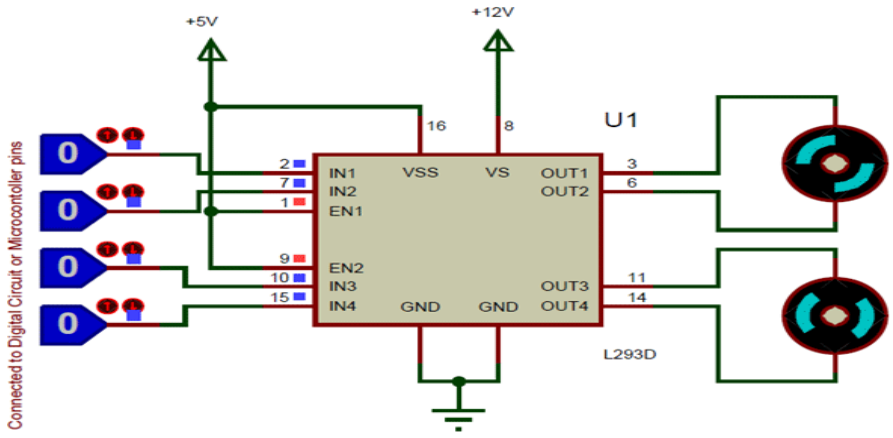



**Fig 4.6.1** *working of motor driver module*

All the Ground pins should be grounded. There are two power pins for this IC, one is the Vss (Vcc1) which provides the voltage for the IC to work, this must be connected to +5V. The other is Vs(Vcc2)

which provides voltage for the motors to run, based on the specification of your motor you can connect this pin to anywhere between 4.5V to 36V, here I have connected to +12V.

The Enable pins (Enable 1,2 and Enable 3,4) are used to Enable Input pins for Motor 1 and Motor 2 respectively. Since in most cases we will be using both the motors both the pins are held high by default by connecting to +5V supply. The input pins Input 1,2 are used to control the motor 1 and Input pins 3,4 are used to control the Motor 2. The input pins are connected to the any Digital circuit or microcontroller to control the speed and direction of the motor. You can toggle the input pins based on the following table to control your motor.

| | | |
|---|---|---|
| Input 1 = HIGH(5v) | Output 1 = HIGH | Motor 1 rotates in Clock wise Direction |
| Input 2 = LOW(0v) | Output 2 = LOW | |
| Input 3 = HIGH(5v) | Output 1 = HIGH | Motor 2 rotates in Clock wise Direction |
| Input 4 = LOW(0v) | Output 2 = LOW | |

| | | |
|---|---|---|
| Input 1 = LOW(0v) | Output 1 = LOW | Motor 1 rotates in Anti-Clock wise Direction |
| Input 2 = HIGH(5v) | Output 2 = HIGH | |
| Input 3 = LOW(0v) | Output 1 = LOW | Motor 2 rotates in Anti -Clock wise Direction |
| Input 4 = HIGH(5v) | Output 2 = HIGH | |

| | | |
|---|---|---|
| Input 1 = HIGH(5v) | Output 1 = HIGH | Motor 1 stays still |
| Input 2 = HIGH(5v) | Output 2 = HIGH | |
| Input 3 = HIGH(5v) | Output 1 = LOW | Motor 2 stays still |
| Input 4 = HIGH(5v) | Output 2 = HIGH | |

### 4.6.2 Applications:

- Used to drive high current Motors using Digital Circuits
- Can be used to drive Stepper motors.
- High current LED's can be driven
- Relay Driver module (Latching Relay is possible)

## 4.7 16 X 2 LCD DISPLAY:

The term LCD stands for liquid crystal display. It is one kind of electronic display module used in an extensive range of applications like various circuits & devices like mobile phones, calculators, computers, TV sets, etc. These displays are mainly preferred for multi-segment light-emitting diodes and seven segments. The main benefits of using this module are inexpensive; simply programmable, animations, and there are no limitations for displaying custom characters, special and even animations, etc.


**Fig 4.7** *16X2 LCD Display*

### 4.7.1 LCD 16×2 Pin Diagram:

The 16×2 LCD pinout is shown below.

- Pin1 (Ground/Source Pin): This is a GND pin of display, used to connect the GND terminal of the microcontroller unit or power source.
- Pin2 (VCC/Source Pin): This is the voltage supply pin of the display, used to connect the supply pin of the power source.

- Pin3 (V0/VEE/Control Pin): This pin regulates the difference of the display, used to connect a changeable POT that can supply 0 to 5V.
- Pin4 (Register Select/Control Pin): This pin toggles among command or data register, used to connect a microcontroller unit pin and obtains either 0 or 1(0 = data mode, and 1 = command mode).
- Pin5 (Read/Write/Control Pin): This pin toggles the display among the read or writes operation, and it is connected to a microcontroller unit pin to get either 0 or 1 (0 = Write Operation, and 1 = Read Operation).
- Pin 6 (Enable/Control Pin): This pin should be held high to execute Read/Write process, and it is connected to the microcontroller unit & constantly held high.
- Pins 7-14 (Data Pins): These pins are used to send data to the display. These pins are connected in two-wire modes like 4-wire mode and 8-wire mode. In 4-wire mode, only four pins are connected to the microcontroller unit like 0 to 3, whereas in 8-wire mode, 8-pins are connected to microcontroller unit like 0 to 7.
- Pin15 (+ve pin of the LED): This pin is connected to +5V
- Pin 16 (-ve pin of the LED): This pin is connected to GND.

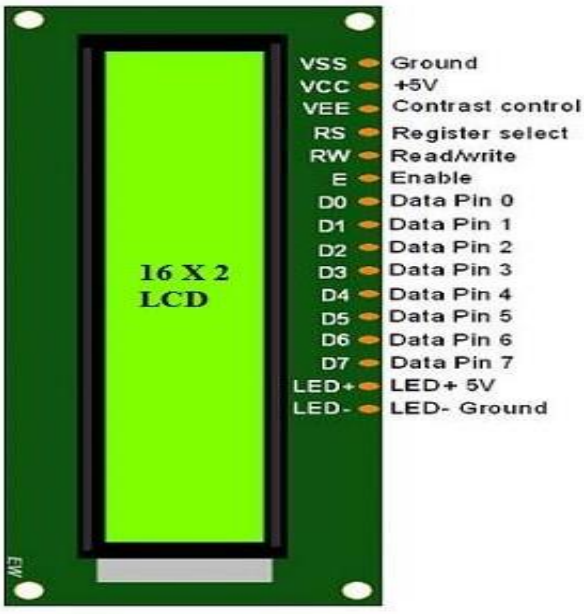



**Fig 4.7.1** *LCD-16×2-pin-diagram*

### 4.7.2 Registers of LCD:

A 16×2 LCD has two registers like data register and command register. The RS (register select) is mainly used to change from one register to another. When the register set is '0', then it is known as command register. Similarly, when the register set is '1', then it is known as data register.

- **Command Register :**

  The main function of the command register is to store the instructions of command which are given to the display. So that predefined tasks can be performed such as clearing the display, initializing, set the cursor place, and display control. Here commands processing can occur within the register.

- **Data Register :**

  The main function of the data register is to store the information which is to be exhibited on the LCD screen. Here, the ASCII value of the character is the information which is to be exhibited on the screen of LCD. Whenever we send the information to LCD, it transmits to the data register, and then the process will be starting there. When register set =1, then the data register will be selected.

### 4.7.3 Features of LCD 16x2 :

The features of this LCD mainly include the following.

- The operating voltage of this LCD is 4.7V-5.3V
- It includes two rows where each row can produce 16-characters.
- The utilization of current is 1mA with no backlight
- Every character can be built with a 5×8 pixel box
- The alphanumeric LCDs alphabets & numbers
- Is display can work on two modes like 4-bit & 8-bit
- These are obtainable in Blue & Green Backlight
- It displays a few custom generated characters

## 4.8 9V BATTERY POWER SUPPLY:

A 9V battery is a widely used power source in electronics, known for its compact size and ability to provide a stable direct current (DC) voltage. Typically rectangular in shape and equipped with snap connectors on top, it is most commonly available in alkaline, lithium, or rechargeable (NiMH) chemistries. The nominal voltage output is 9 volts, which makes it suitable for a variety of low-power applications such as smoke detectors, remote controls, multi meters, and educational electronic projects.

Alkaline 9V batteries generally offer a capacity between 400 to 600 mAh, while lithium versions provide a longer lifespan and better performance in extreme temperatures. Although convenient, 9V batteries are not ideal for high-current devices like motors or GSM modules, as their voltage tends to drop quickly under heavy load, leading to inconsistent performance or device resets. In embedded systems and microcontroller-based projects, the 9V battery is often used in combination with a voltage regulator to safely step down the voltage to 5V or 3.3V for components like the ESP32. Its portability and ease of use make it a go-to option for powering small circuits during prototyping or demonstrations, though for long- term or power-intensive applications, other power sources may be more appropriate.


**Fig 4.8** *9v Battery*

### 4.8.1 Applications:

- Telecommunication equipments, such as cordless phone, cellular phones, transceivers.
- Emergency lightings and security systems.
- Radio control toys.
- Audio and video devices, such as camcorders, walkmans, MDs, portable DVDs, VCDs, TVs.
- Information devices, such as notebook computers, PDAs, portable fax machines.
- Power tools.
- Electric bicycles 2 TYSONIC NI-MH 9V RECHARGEABLE BATTERY.

- Other applications, such as electric shavers, electric toothbrushes, massagers, portable vacuum cleaners, portable terminals.

### 4.9 BUZZER:

An audio signalling device like a beeper or buzzer may be electromechanical or piezoelectric or mechanical type. The main function of this is to convert the signal from audio to sound. Generally, it is powered through DC voltage and used in timers, alarm devices, printers, alarms, computers, etc. Based on the various designs, it can generate different sounds like alarm, music, bell & siren.


**Figure 4.9** *Buzzer Pin Configuration*

The pin configuration of the buzzer is shown below. It includes two pins namely positive and negative. The positive terminal of this is represented with the '+' symbol or a longer terminal. This terminal is powered through 6Volts whereas the negative terminal is represented with the '-'symbol or short terminal and it is connected to the GND terminal.

The specifications of the buzzer include the following.

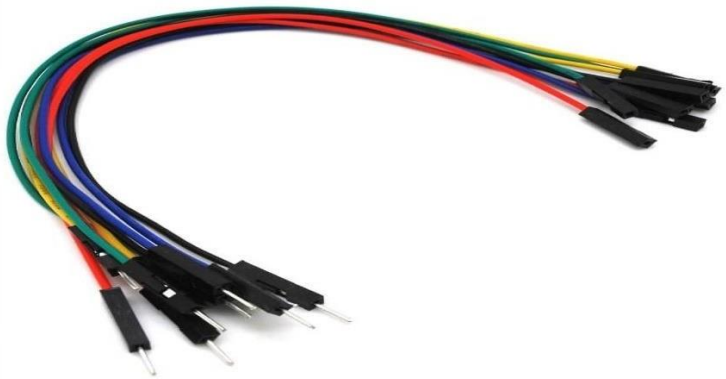
- Colour is black
- The frequency range is 3,300Hz
- Operating Temperature ranges from – 20° C to +60°C

. Operating voltage ranges from 3V to 24V DC

- The sound pressure level is 85dBA or 10cm
- The supply current is below 15mA

## 4.10 JUMPER WIRES:

Jumper wires are essential components used in electronics to establish connections between different points on a breadboard or between breadboards and other electronic components. These wires are typically made of copper for good conductivity and come with plastic insulation to prevent short circuits. Jumper wires are available in three main types based on their connectors: male-to-male, male-to-female, and female-to-female, allowing versatile connections depending on the setup. They are commonly used in prototyping and testing circuits without the need for soldering, making them reusable and highly convenient for hobbyists, students, and engineers. The wires come in various lengths and colors, which help in organizing and identifying connections easily. They are widely used with microcontrollers like Arduino, ESP32, and Raspberry Pi. Their flexibility and ease of use make jumper wires a fundamental tool in circuit design and



development, especially in educational and experimental electronic projects.

**Fig 4.10** *Jumper wires*

# CHAPTER 5 IMPLEMENTATION & WORKING OPERATION

This system uses an ESP32 microcontroller to integrate sensors, GSM communication, and deterrent mechanisms to detect animals near railway tracks and alert authorities.

## 5.1 COMPONENT ROLES:

- **ESP32**: Central controller for data processing and decision-making.
- **Ultrasonic Sensors**: Measure distance to detect static obstacles (e.g., animals on tracks).
- **PIR Sensor**: Detects motion to confirm the presence of live animals.
- **GSM SIM900A**: Sends SMS alerts to the station master via cellular networks.
- **L293D Motor Driver**: Activates a deterrent mechanism (e.g., noise/siren, flashing lights).
- **LCD Module**: Displays real-time system status (e.g., "Clear" or "Animal Detected").

## 5.2 WORKING OPERATION OF THE SYSTEM:

**Step 1: Continuous Monitoring**

- **Ultrasonic Sensors**: Emit sound waves to measure distance. If the distance falls below a predefined threshold (e.g., 1 meter), it signals a potential obstacle.
- **PIR Sensor**: Scans for infrared radiation (motion). If triggered, it confirms a moving object (likely an animal).

**Step 2: Obstacle Verification**

- The ESP32 cross-checks signals from both sensors to reduce false alarms:
  - **Condition**: Ultrasonic sensor detects proximity and PIR detects motion.
  - If both conditions are met, the system confirms an animal is present.

**Step 3: Alert Generation**

- **GSM Module Activation**:
  - ESP32 sends a command to the SIM900A module.
  - SIM900A transmits an SMS to the station master's phone with a predefined message: *"ALERT: Animal detected near Track [SMS ALERT] at [Time]."*

- **LCD Update**: Displays "Animal Detected! SMS Sent."

**Step 4: Deterrent Activation**

- **L293D Motor Driver**:
  - ESP32 triggers the motor driver to activate a connected deterrent (e.g., motorized siren, rotating lights).
  - Continues until sensors no longer detect the animal.

**Step 5: Reset**

- Once the animal moves away (sensors return to normal), the system:
  - Stops the deterrent mechanism.
  - Updates LCD to "Track Clear.

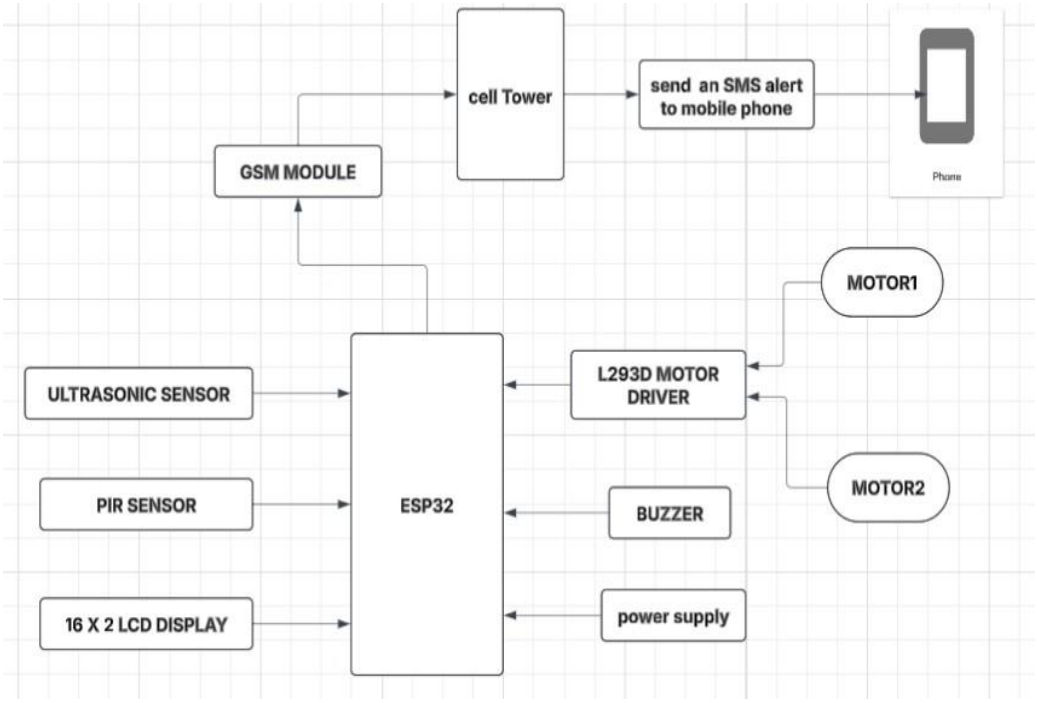



**Figure 5.2.** *Block Diagram of System*

## 5.3 HOW A MOBILE PHONE RECEIVES SMS THROUGH GSM MODULE (SIM800L)?

**1. GSM Network Communication:**

- The SIM800L module uses a GSM (Global System for Mobile Communications) network to send and receive data, including SMS.

- The mobile phone and the GSM module (SIM800L) communicate through cell towers operated by a cellular network provider.

**2. Incoming SMS to SIM800L:**

- When an SMS is sent to the mobile number associated with the SIM card inserted in the SIM800L:
  - The message is routed through the GSM network.
  - The SMSC (Short Message Service Center) handles the delivery.
  - The SMSC forwards the message to the SIM card inserted in the GSM module.

**3. SIM800L Processes Incoming SMS:**

- The SIM800L continuously monitors the network for incoming SMS using the AT+CNMI command.
  - AT+CNMI=2,2,0,0,0
  - 2,2 – Enables new message indication and sends the SMS content directly to the serial interface.
- When a message is received, it is immediately sent to the connected microcontroller (e.g., Arduino/ESP32).

**4. Microcontroller Reads SMS:**

- The microcontroller (e.g., Arduino) reads the SMS data via the serial interface.
- The SMS content is parsed using AT commands.
  - AT+CMGF=1 – Sets SMS mode to text mode.
  - AT+CMGR=<index> – Reads the SMS stored at a specific index.
- If AT+CNMI=2,2,0,0,0 is configured:
  - The message appears directly over the serial connection.

**5. SMS Display on Mobile**

- The GSM module does not send the SMS to a phone but receives it instead.
- If a mobile phone is connected to the same network, it receives the SMS normally through the GSM network.
- Mobile phones read SMS messages from the SIM card or the internal memory.

## 5.4 PROPOSED METHODOLOGY:

When the system is powered on, all connected components including ESP32, GSM SIM900A, ultrasonic sensors, PIR sensor, LCD module, and L293D motor driver are initialized. The LCD displays a welcome

message and system status (e.g., “System Ready”, “Monitoring…”). The PIR sensor continuously scans for motion of animals near the track. Simultaneously, the ultrasonic sensor monitors for any physical obstacles within a set distance threshold.

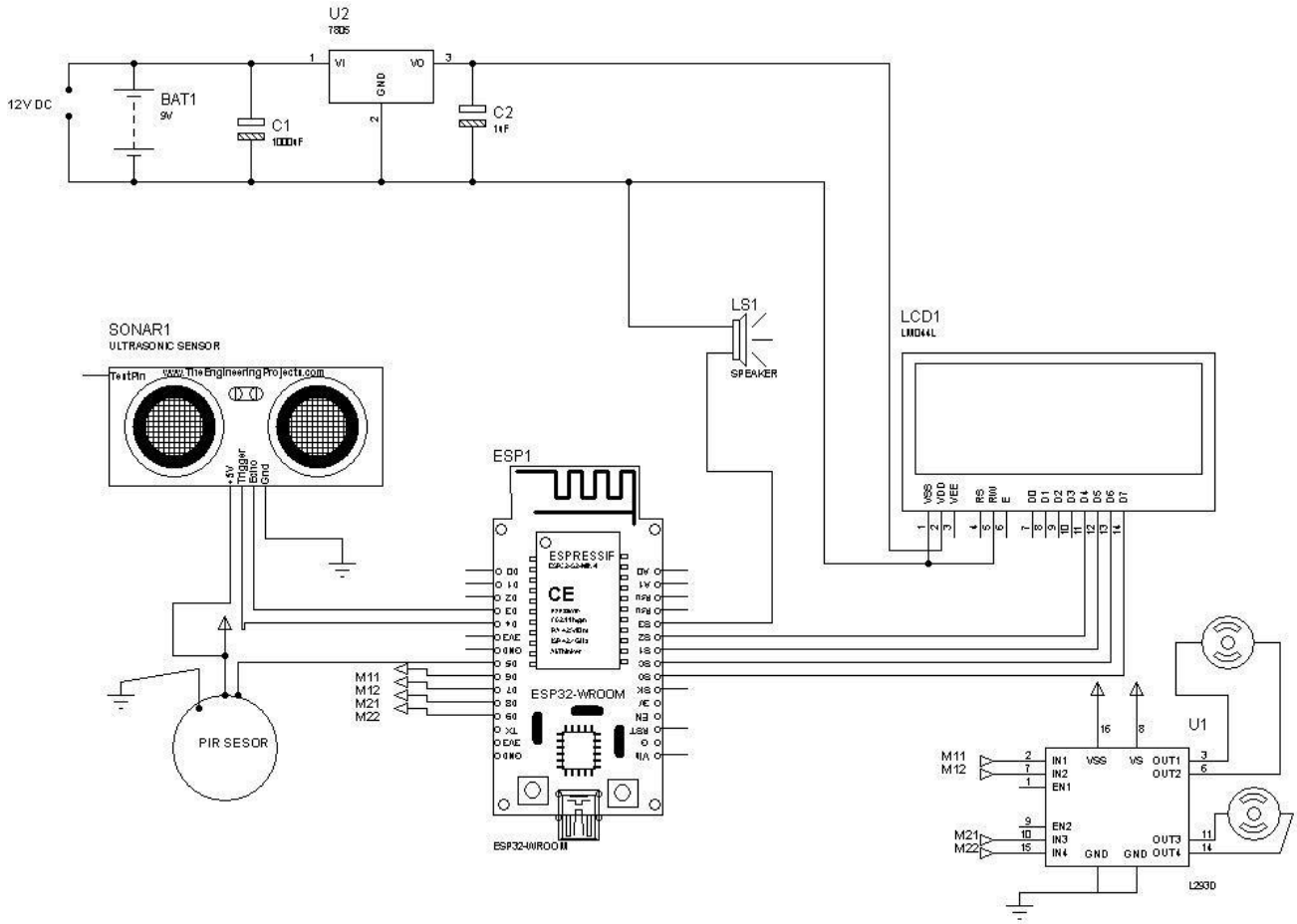



**Figure 5.3** *Block Diagram of Proposed System*

If either sensor (PIR or ultrasonic) detects an obstacle or motion. The sensor sends a digital signal to the ESP32 microcontroller. ESP32 processes the data and confirms the presence of an obstacle/animal on or near the railway track. Once detection is confirmed:

- ESP32 communicates with the GSM SIM900A module using AT commands over a serial interface.
- A pre-defined **SMS alert** is sent to the station master or relevant railway authority via the cellular mobile network.
- The SMS might read something like:

  *“Alert: Animal/Obstacle Detected Near Track Section 3 – Immediate Action Required.”*

Simultaneously, the LCD displays a message such as:

- "Obstacle Detected"
- "SMS Sent to Station Master"

The system also includes the physical deterrent (e.g., buzzer, motor to drive a barrier or scare device) The L293D motor driver module receives a signal from ESP32 and activates the motor to perform the desired action (e.g., rotating a barrier arm or activating a siren to scare animals away). After sending the SMS and executing the response actions, the system resets itself after a few seconds. It then returns to surveillance mode, continuously monitoring for further obstacles.

## 5.5 OUTPUT:

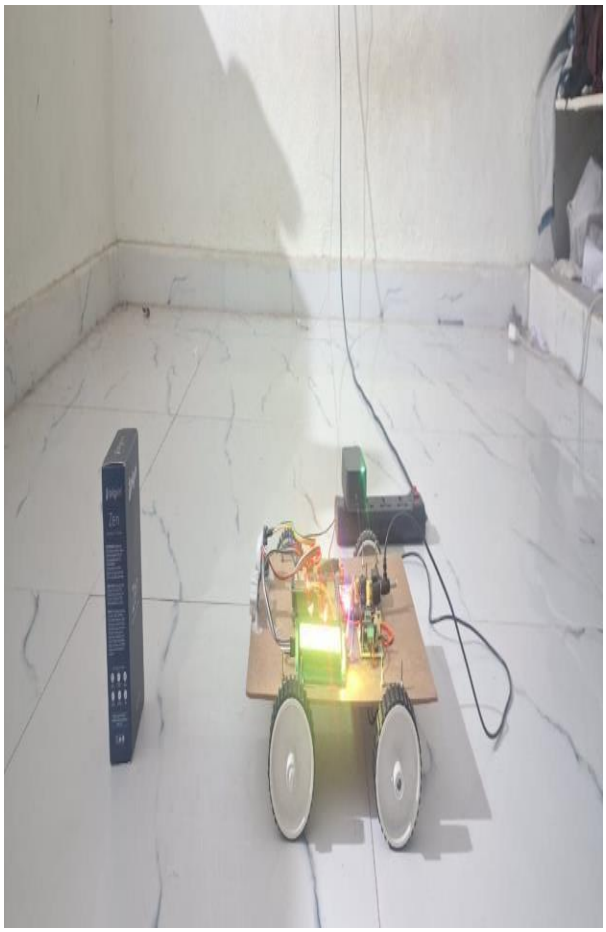


**Figure 5.5.1(a)**

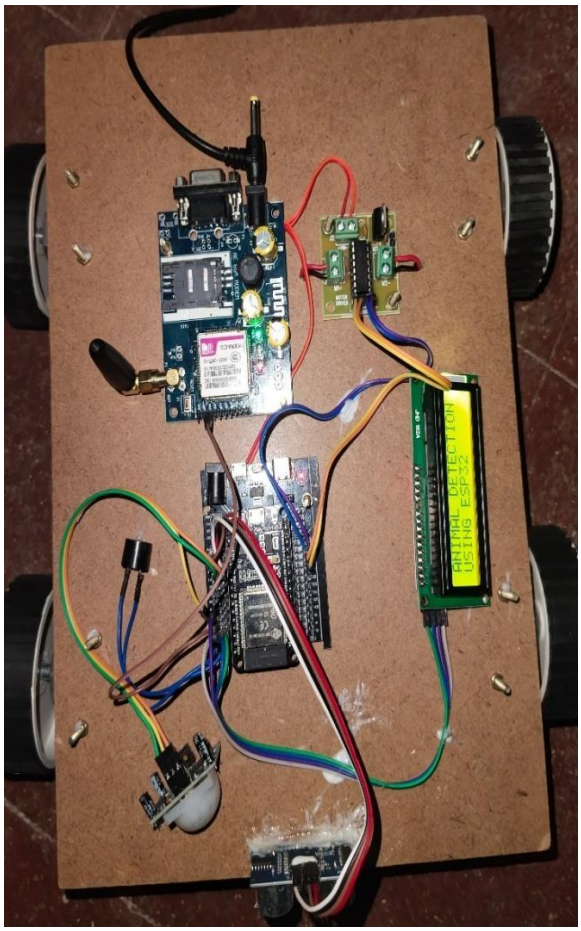



**Figure 5.5.1(b)**

**Message Alert:**

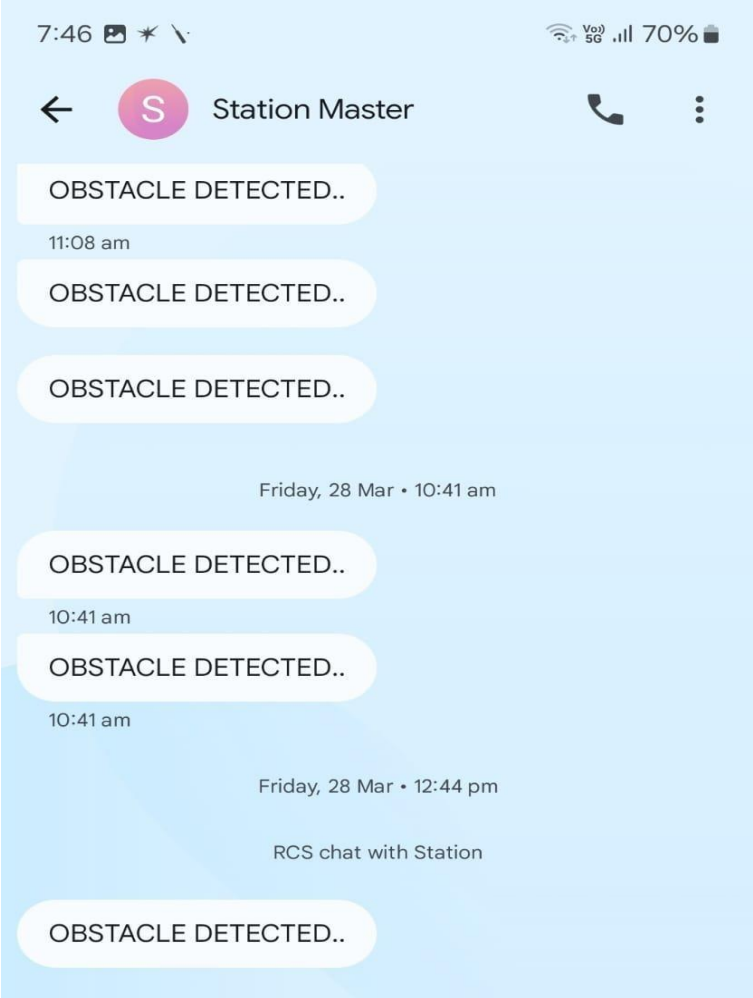



**Figure 5.5.2** *Sending Message Alert*

# Chapter 5

# Conclusion and Recommendation

## 5.1 CONCLUSION

The proposed system for creature detection was used for the creature detection on the rails. The proposed system was placed in the accident prone areas where the accidents occur due to the wild animals crossing the rail, vehicle accidents, falling down of trees etc. will be monitored. According to the system, we were verified the system performance in real condition. Using ultrasonic Sensor we identify the train and using PIR Sensor we identify the creature. This information will be helpful for the loco-pilot to stop the train and avoid accidents that harm the creature in the track. The future of surveillance for animal tracking promises significant advancements driven by technological innovation and evolving research methodologies. Miniaturization of tracking devices will lead to the development of smaller, lighter, and less invasive tags, expanding the range of species that can be monitored. Integration of multiple sensors, including GPS, accelerometers, and environmental sensors, will provide a more comprehensive understanding of animal behavior and habitat use. Data analytics and artificial intelligence techniques will enable efficient processing and analysis of large tracking datasets, uncovering patterns and insights that were previously inaccessible. Real-time monitoring capabilities, coupled with remote sensing technologies like satellite imagery and UAVs, will facilitate broad-scale monitoring of wildlife populations and habitats. Ethical considerations and animal welfare concerns will continue to guide research protocols, ensuring that tracking methods prioritize the well-being of study animals. Ultimately, these advancements will contribute to more effective wildlife conservation and management efforts in the face of global environmental challenges.

## 5.2 FUTURE RECOMMENDATIONS

- Adding a Camera: If the current project is interfaced with a camera (e.g. a Webcam) robot can be driven beyond line-of-sight & range becomes practically unlimited as networks have a very large range.
- We can extend this project with wireless technology by IR (or) RF (or) ZIGBEE.
- We can use the DTMF receiver by using the mobile phone.
- This robot can be used for pick and place the required object by giving directions to

the robot but ultrasonic sensor should be replaced depending upon the application.

## 5.3 FUTURE WORKS

To enable robots to be able to adapt to its environment is an important domain of robotics research. Whether this environment be underwater, on land, underground, in the air or in space. A fully autonomous robot vehicle has the ability to

- Work for an extended period of time without intervention from human or a need for power supply.
- Avoid situations that are harmful.
- Move either all or part of itself throughout its operating environment.

The most effective method to increase the accuracy of my robot is the inclusion of better sensors, although the project cost might increase but the accuracy will definitely increase as well as the problem space where the robot can be used. Better actuators will result in a faster and more efficient robot.

## 5.4 ADVANTAGES

- Collision control.
- It provides Safe Navigation.
- This is the basic of all robot and has a wide scope of extensions

## 5.5 APPLICATION

- **Wildlife Protection**: Prevent animal-train collisions, ensuring the safety of wildlife and reducing casualties.
- **Railway Safety**: Enhance operational safety by providing real-time alerts to train operators about animal presence.
- **Accident Prevention**: Use automated deterrents like lights or sounds to discourage animals from entering railway tracks.
- **Cost Efficiency**: Reduce railway operational costs associated with delays and repairs caused by animal collisions.

# REFERENCES


1. Vishruth Kumar, Yashas L, Viresh Kadole , Raghavendra C, Charanraj B“Railway Inspection System Using IoT” International Research Journal of Engineering and Technology (IRJET),2023.
2. Alan Shaju, Aravind Madhu, Doel Sam “Automated Railway Track Detection System” International Journal of Research in Engineering and Science (IJRES) ,2023.
3. Ramesh B E, Beena Sheril , Priyanka L S, Rakshitha U, Sreeja B H, Supraja C S “RAILWAY TRACK FAULT DETECTION USING ML & IOT” Journal of Emerging Technologies and Innovative Research (JETIR) , 2024 .
4. Satish Kumar, Dr. Narayanan.R, "Solar powered crack detector applicable for Railway tracks", International Journal of Scientific Research, Vol. 3 , 2019.
5. Jubayer Jalil and M. B. I. Reaz, “Accident detection and reporting system using GPS, GPRS and GSM technology”,in IRJET, October 2019.
6. Abhisekh Jain S, Arvind S, Balaji B.S, Ram Viyas N.P, "On board rail track safety monitoring system", International Conference on Advanced Communication Systems, Jan. 2021, GCT, Coimbatore.
7. SarathChandran.P and Karthika.M, “Track system requirements, Railway Group Standards, GC/RT5021”,in Railway Safety and Standard BoardLimited, London, October 2019.
8. N.Karthick, R.Nagarajan, S.Suresh and R. Prabhu, ”Implementation of Railway Track Crack Detection and Protection”, International JournalOf Engineering And Computer Science,2017.
9. RanuDewangan and Pratibhadevi Umesh, “Automatic Accident Control System on Railway Tracks”, International Journal of Advanced Researchin Electronics and Communication Engineering (IJARECE), July 2016.
10. Likhith Chandra1 and Harsha V, “IoT Based Railway Track Faults Detection and Localization using Acoustic Analysis and Signalling”, International Journal of Advanced Research in Science, Communication and Technology (IJARSCT),2024.

# DEPARTMENT
# OF
# ELECTRONICS AND COMMUNICATION ENGINEERING

## VISION

To achieve global standards of quality in technical education with the help of advanced resources and automated tools to bridge the gap between industry and academia.

## MISSION

1. Build the students technically competent on global arena through effective teaching learning process and world-class infrastructure.
2. Inculcate professional ethics, societal concerns, technical skills and life-long learning to succeed in multidisciplinary fields.
3. Establish competency center in the field of Artificial Intelligence and Machine Learning with the collaboration of industry and innovative research.

## PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

**The graduates of Computer Science and Engineering with Specialization in Artificial Intelligence & Machine Learning will be able to:**

**PEO 1: Domain Knowledge**: Impart strong foundation in basic sciences, Mathematics, Engineering and emerging areas by Advanced tools and Technologies.

**PEO 2: Professional Employment**: Develop Professional skills that prepare them for immediate employment in industry, government, entrepreneurship and R&D.

**PEO 3: Higher Degrees**: Motivation to pursue higher studies and acquire masters and research.

**PEO 4: Engineering Citizenship**: Communicate and work effectively, engage in team work, achieve professional advancement, exhibit leadership skills, and ethical attitude with a sense of social responsibility.

**PEO 5: Lifelong Learning**: Lead edge of the industrial engineering discipline and respond to challenges of an ever-changing environment with the most current knowledge and technology.

## PROGRAM OUTCOMES (POs)

**Program outcomes for the graduates of Electronics and Communications Engineering**

1. **Engineering Knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem Analysis**: Identify, formulate, review research literature, and analyse complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, and environmental considerations.
4. **Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society**: Apply reasoning informed by contextual knowledge to assess societal, health, safety, legal and cultural issues, and the consequent responsibilities relevant to professional engineering practice.
7. **Environment and sustainability**: Understand the impact of professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of engineering practice.

**9.** **Individual and teamwork**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

**10.** **Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective Presentations, and give and receive clear instructions.

**11.** **Project management and finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary Environments.

**12.** **Life-long learning**: Recognize the need for and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

## PROGRAM SPECIFIC OUTCOMES (PSOs)

| | |
|---|---|
| **PSO 1** | Analyze, design and implement specific engineering problems in the areas of VLSI and Embedded systems. |
| **PSO 2** | Apply the knowledge of domain specific skill set for analysis of Signal Processing and Communications. |
| **PSO 3** | Analyze and solve the complex engineering problems using state of the art hardware and software tools. |
| **PSO 4** | Develop proficiency in innovative technologies to sustain with the dynamic industry challenges. |

## COURSE OUTCOMES (COs) FOR PROJECT WORK

| CO No. | Course Outcome Statement | Taxonomy Level |
|---|---|---|
| **C425.1** | Identify challenging practical problems, solutions of Electronics and Communication Engineering field. | APPLY |
| **C425.2** | Analyse the various methodologies and technologies and discuss with team for solving the problem. | ANALYZE |

| | | |
|---|---|---|
| **C425.3** | Apply technical knowledge and project management skills for solving the problem. | APPLY |
| **C425.4** | Design and Development of technical projects as an individual or in a team | CREATE |
| **C425.5** | Prepare the project reports and give proper explanation during the presentation and demonstration. | CREATE |

## Correlation Levels

| | |
|---|---|
| Substantial/High | 3 |
| Moderate Medium | 2 |
| Slight/Low | 1 |
| No correlation | |

## CO-PSO Correlation Matrix

| **COs** | **PSOs** | | |
|---|---|---|---|
| | **PSO1** | **PSO2** | **PSO3** |
| **CO1** | 1 | 3 | 2 |
| **CO2** | 1 | 3 | 3 |
| **CO3** | 2 | 3 | 3 |
| **CO4** | 1 | 3 | 3 |
| **CO5** | 1 | 2 | 3 |
| **CO** | 1 | 3 | 3 |

### CO-PO Correlation Matrix

| **COs** | **POs** | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** |
| **CO1** | 3 | 3 | 2 | 3 | 2 | 1 | 1 | 1 | 2 | 1 | 1 | 2 |
| **CO2** | 3 | 3 | 2 | 3 | 2 | 1 | 1 | 1 | 2 | 2 | 2 | 2 |
| **CO3** | 3 | 3 | 2 | 3 | 3 | 1 | 1 | 1 | 3 | 2 | 3 | 3 |
| **CO4** | 2 | 2 | 3 | 2 | 3 | 2 | 1 | 2 | 3 | 3 | 3 | 3 |
| **CO5** | 1 | 2 | 2 | 2 | 3 | 2 | 1 | 3 | 2 | 3 | 3 | 3 |
| **CO** | 3 | 3 | 2 | 3 | 3 | 1 | 1 | 1 | 3 | 3 | 3 | 3 |

## APPENDIX - A

```
#include <LiquidCrystal_I2C.h>
int lcdColumns = 16;
int lcdRows = 2;
LiquidCrystal_I2C lcd(0x27, lcdColumns, lcdRows);
const int bz = 23;
const int m11 = 13;
const int m12 = 12;
const int m21 = 14;
const int m22 = 27;
const int pir = 16;
const int trigPine_in= 2;
const int echoPine_in= 4;
long duration, distance,cm_in,inches_in;
int x = 0;
void setup() {
  Serial.begin(9600);              //Initialize serial
  lcd.begin();
  lcd.backlight();
  lcd.setCursor(0,0);
  lcd.print("ANIMAL DETECTION");
  lcd.setCursor(0,1);
  lcd.print("USING ESP32");
  delay(2000);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("PIR:");
  lcd.setCursor(0,1);
  lcd.print("DIS:");
  pinMode(bz,OUTPUT);
  pinMode(pir,INPUT);
  pinMode(m11,OUTPUT);
  pinMode(m12,OUTPUT);
```

```
  pinMode(m21,OUTPUT);
  pinMode(m22,OUTPUT);
  digitalWrite(m11,LOW);
  digitalWrite(m12,LOW);
  digitalWrite(m21,LOW);
  digitalWrite(m22,LOW);
  digitalWrite(bz,HIGH);
  pinMode(trigPine_in, OUTPUT);
  pinMode(echoPine_in, INPUT);
  run();
  delay(2000);
  stop();
  digitalWrite(bz,LOW);
  gsm_init();
}
void loop() {
x = digitalRead(pir);
UltrasonicSensor(trigPine_in, echoPine_in);
cm_in = distance;
//Serial.println(distance);
lcd.setCursor(0,1);
lcd.print("DIS:");
lcd.print(distance);
lcd.print("  ");
if((distance < 20)||(x == HIGH))
{
lcd.setCursor(8,0);
lcd.print("STOP ");
stop();
}
else
{
lcd.setCursor(8,0);
lcd.print("MOVING  ");
```

```
run();
}
lcd.setCursor(0,0);
lcd.print("PIR:");
if(x == HIGH)
{
 lcd.print("YES ");
 digitalWrite(bz,HIGH);
 send_sms("ANIMAL DETECTED ");
 delay(1000);
}
else
{
lcd.print("NO  ");
digitalWrite(bz,LOW);
}
if(distance < 20)
{
lcd.setCursor(8,0);
lcd.print("STOP ");
digitalWrite(bz,HIGH);
stop();
send_sms("OBSTACLE DETECTED..");
digitalWrite(bz,LOW);
}
}
void UltrasonicSensor(int trigPin, int echoPin)
{
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
```

```
  distance = (duration / 2) / 29.1;
}
void run()
{
  digitalWrite(m11,HIGH);
  digitalWrite(m12,LOW);
  digitalWrite(m21,HIGH);
  digitalWrite(m22,LOW);
  digitalWrite(bz,LOW);
}
void stop()
{
digitalWrite(m11,LOW);
digitalWrite(m12,LOW);
digitalWrite(m21,LOW);
digitalWrite(m22,LOW);
digitalWrite(bz,LOW);
}
void gsm_init()
{
  Serial.println("AT");
  delay(1000);
  Serial.println("AT+CNMI=2,2,0,0,0");
  delay(500);
  Serial.println("AT+CMGF=1");
}
void send_sms(String str)
{
    Serial.println("AT");
    delay(500);
    Serial.println("AT+CMGF=1");
    delay(500);
    Serial.println("AT+CMGS=\"+917601075424\"\r");//917601075424
    delay(1000);
```

```
    //Serial.println("TROUBLE AT https://www.google.com/maps/place/15.4340,80.0467");
    //Serial.print("OBSTACLE DETECTED");
    Serial.print(str);
    delay(2000);
    Serial.println((char)26); // ASCII code of CTRL+Z
    delay(1000);
}
```