ABSTRACT

Driver drowsiness is a major contributor to road accidents worldwide, resulting in significant loss of life and economic impact. Existing drowsiness detection systems often suffer from limitations such as dependency on lighting conditions, slow physiological indicators, bulky hardware, or insufficient alert mechanisms. To address these gaps, this project proposes a Smart Drowsiness Detection System using the ESP32 microcontroller, integrating both sensor-based and vision-based monitoring for improved accuracy and reliability.

The system uses an IR eye-blink sensor mounted on goggles to continuously track eye closure, while an ESP32-CAM module observes facial cues such as yawning and head tilting. A multi-stage alert mechanism is implemented: a buzzer provides an immediate audible warning, followed by a vibration motor for physical feedback. If the driver remains unresponsive, a GPS module automatically sends the driver's live location to predefined contacts.

Designed as a low-cost, portable, and user-friendly solution, the system enhances safety through real-time drowsiness detection, multi-layer alerts, and emergency communication. By combining IoT-enabled embedded hardware with wearable design, this project demonstrates an effective approach to reducing fatigue-related road accidents and improving driver safety.

TABLE of CONTENTS

List of figures

List of abbreviations

Abbreviation	Full Form
ESP32	Embedded System Platform 32-bit
IR	Infrared
GPS	Global Positioning System
CNN	Convolutional Neural Network
Li-ion	Lithium-Ion
HRV	Heart Rate Variability
SDG	Sustainable Development Goals
ECG	Electrocardiogram
TRL	Technology Readiness Level
PCB	Printed Circuit Board
TP4056	Lithium Battery Charging Module
SIM808	GPS Hardware Module
Wi-Fi	Wireless Fidelity

Chapter 1:

Introduction

Road accidents are one of the biggest safety problems in the world today. Every year, more than 1.3 million people lose their lives and nearly 50 million get injured because of traffic crashes. Apart from the human loss, these accidents also put a heavy financial burden on families and the economy, costing countries about 3% of their GDP.

In India, the numbers are equally worrying. The country sees around 4.6 lakh accidents every year, leading to about 1.68 lakh deaths and over 4.4 lakh injuries. A major reason behind many of these accidents is driver drowsiness—when drivers unknowingly fall asleep or lose focus while driving. For instance, an AIIMS study in Uttarakhand showed that 21% of road accidents were directly due to drivers dozing off, while fatigue played a role in 26% of cases. In Andhra Pradesh, officials say that nearly 20% of accidents—about 20,000 crashes and 8,000 deaths a year—are linked to tired drivers.

This clearly shows how dangerous drowsy driving can be, especially for truck drivers, long-distance travellers, and night-shift workers. The problem is made worse because most existing solutions are either too costly, bulky, or not practical for everyday use.

With recent advances in IoT and embedded systems, we now have the chance to create compact and affordable devices that can actually save lives. Our project works on this idea by designing a wearable drowsiness detection system that is simple and effective. Using an ESP32 microcontroller, the system combines an eye-blink sensor, buzzer, vibration motor, GPS modules, and an ESP32-CAM module. The addition of the camera allows for direct monitoring of the driver's face and behaviour, making the system more reliable by checking both physiological signals (eye closure via IR sensor) and visual cues (yawning, head tilts, etc.). It gives alerts in steps—first with sound, then vibration, and finally by sending the driver's live location to an emergency contact if needed. In short, this project aims to provide a low-cost, user-friendly, and life-saving solution that can help reduce accidents caused by drowsiness and make roads safer for everyone.

Chapter 2:

Literature survey

In paper [1] authors proposed a drowsiness detection system using an **Arduino Nano**, an **IR eye blink sensor**, and a **buzzer** to alert the driver when eyes remain closed beyond a set threshold. The system applies a simple control logic to monitor eye closure and trigger alarms, but its reliance on the low-power Arduino Nano limits processing speed and memory, making it less suitable for multitasking or integrating multiple alert mechanisms.

[2] This system uses the **MAX30102** sensor to monitor the driver's **heart rate** for detecting drowsiness, connected via **ESP8266** to a cloud platform with data uploaded to **Blynk** for visualization. The drawback is that heart rate changes react slowly, with no behavioral sensor (e.g., eye blink) or immediate physical alert for emergencies.

[3] This project implements a **real-time drowsiness detection system** using an **ESP32** board and an **infrared eye blink sensor**. It tracks prolonged eye closure and triggers a **buzzer** to alert riders, offering a compact and low-cost solution. However, it relies solely on a buzzer, which may be ineffective if the rider is in deep sleep or has impaired hearing, as no backup alert is provided.

[4] This method monitors vehicle behaviour, such as steering wheel movements (SWM) and lane deviation (SDLP), assuming irregular steering or drifting indicates fatigue. The limitation is that detection strongly depends on external factors like road quality, lighting, and weather, leading to false alarms or missed drowsiness events.

[5] This work compares multiple lightweight vehicle dynamics–based methods using a shared dataset under a uniform framework, ensuring reproducibility and fair benchmarking. The drawback is that many methods suffer from data leakage and limited transparency, reducing trustworthiness and hindering real-world generalization despite good lab accuracy.

[6] This approach measures steering wheel grip pressure and analyses it with SVM and CNN models to infer fatigue. However, grip force varies with driver habits, stress, and ergonomics, causing false positives or inconsistent detection depending on individual grip styles.

[7] This study uses wearable ECG/PPG sensors to capture heart rate variability (HRV), analysed with CNN + RNN models for drowsiness classification. The limitation is that multiple physiological sensors and deep learning models make the system intrusive, power-hungry, and impractical for everyday use.

[8] This study applies a **camera-based system** for monitoring drowsiness through eye movement and yawning. However, it relies only on the camera, making it **vulnerable to lighting**, **face angle, and obstruction issues**. In contrast, our project improves on this by integrating both the **IR eye blink sensor** and the **ESP32-CAM**, combining sensor-based accuracy with visual monitoring. This dual approach increases reliability across different driving conditions.

Chapter 3:

Problem Analysis & Solution

3.1 Problem Definition

Driver drowsiness is one of the leading causes of road accidents, especially during long-distance travel, night driving, and commercial transportation. Several studies and accident statistics highlight that a significant percentage of crashes occur because drivers unintentionally fall asleep or experience reduced alertness. Existing drowsiness detection systems attempt to address this issue but face major limitations:

- **Physiological-based systems** (heart rate, HRV, grip sensors) often respond slowly, vary widely between individuals, or require multiple sensors, making them impractical for daily use.
- **Camera-only systems** depend heavily on lighting conditions, face orientation, and clear visibility, causing false detections or system failure.
- **Vehicle-based systems** (lane deviation, steering behaviour) are influenced by road quality, weather, traffic density, and driver habits, leading to unreliable results.
- **Low-end microcontroller systems** (e.g., Arduino-based blink detection) are limited to simple alerts and cannot multitask or integrate multiple detection modalities.
- **Single-alert mechanisms** such as only a buzzer may be insufficient—deep drowsiness, hearing issues, or noisy environments can reduce their effectiveness.

Due to these limitations, there is a clear need for a **compact, low-cost, multimodal, and highly reliable drowsiness detection system** that operates effectively in real time, works in diverse lighting conditions, and includes robust emergency handling if the driver remains unresponsive.

3.2 Proposed Solution

The proposed system overcomes the limitations of existing methods by combining **sensor-based eye monitoring**, **camera-based facial monitoring**, and a **multi-level alert and communication mechanism** into one portable, wearable device powered by the ESP32 platform.

Key Features of the Proposed Solution

1. Dual-Mode Drowsiness Detection

To improve reliability, the system integrates two complementary detection techniques:

- **IR Eye Blink Sensor:** Continuously monitors eyelid movement and detects prolonged eye closure independent of ambient lighting.
- **ESP32-CAM Module:** Provides visual monitoring of facial cues such as yawning, head drooping, or reduced attentiveness, increasing detection accuracy under varied driving conditions.

2. Multi-Stage Alert Mechanism

Once drowsiness is detected:

1. **Stage 1 – Buzzer Alert:** Emits a loud audible alarm to immediately draw the driver's attention.
2. **Stage 2 – Vibration Alert:** A wristband vibration motor provides physical feedback to wake the driver, helpful when sound alone is insufficient.
3. **Stage 3 – Emergency Communication:** If the driver does not respond, a **GPS module** sends an alert containing the driver's **live GPS location** to an emergency contact.

3. Efficient Embedded Processing

Using the **ESP32**, which supports multitasking, Wi-Fi/Bluetooth, and camera interfacing, the system can:

- Process sensor and camera data in real time
- Trigger alerts without lag
- Handle GPS tracking

- Operate on a compact, battery-powered setup

4. Portable and Wearable Design

- IR blink sensor mounted on goggles
- Vibration motor attached to a wristband
- Rechargeable 18650 battery with TP4056 charging
- Lightweight setup suitable for motor riders, car drivers, and commercial vehicle operators

5. Low-Cost Implementation

All components are cost-effective while maintaining accuracy and reliability, making the system suitable for large-scale deployment.

Chapter 4:

Methodology & Implementation

1.1 Block Diagram

The provided diagram outlines the architecture of a real-time Drowsiness Detection and Alert System, primarily built around the versatile **ESP32 microcontroller**. This system integrates video monitoring, sensor input, and network connectivity to actively prevent accidents caused by driver fatigue. The system's backbone is the **Main ESP32 Controller**, which functions as a web server, housing the core detection logic and managing all peripheral interactions. This controller, along with an **ESP32-CAM** module, connects to a local **Wi-Fi Network** (SSID: POCO M6 Pro 5g).

- **ESP32-CAM:** This module, identified by the IP address **10.173.208.243**, is responsible for capturing a video stream, presumably focused on the driver's face/eyes. It also responds to basic HTTP GET requests (/on, /off) for control from the network.
- **Client Device:** A laptop or smartphone acts as the monitoring and control interface, typically displaying a **Dashboard**. It communicates with both ESP32 modules over the Wi-Fi network using standard HTTP protocols.
- **Control/Status:** HTTP GET requests are used for fetching the dashboard (/Dashboard) and system logs (/log), while HTTP POST requests (/gps) could be used to send location data. Specific control actions for the camera (/camOn, /camOff) are executed via HTTP GET requests, which receive HTML/JSON responses from the main controller.

The core logic for detecting drowsiness relies on both the video stream and a dedicated sensor:

- **Eye-Closure Sensor:** An analog or digital sensor, such as an **IR sensor**, is connected to the **Main ESP32 Controller** via **GPIO 15** (configured as an Input with a Pull-up resistor). This sensor likely monitors the driver's eye movement or closure. Persistent eye-closure or excessive blinking, as detected by this sensor and/or the video stream processing (if implemented on the ESP32-CAM or main controller), triggers the alert system.

Once drowsiness is detected, the Main ESP32 Controller activates three distinct types of alerts to immediately notify and rouse the driver:

1. **Audible Alert:** An **Active Buzzer** is connected to **GPIO 5** (Output) and is triggered to emit a loud sound, immediately drawing the driver's attention.
2. **Tactile Alert:** A **Vibration Motor Module** is connected to **GPIO 19** (Output). This provides a physical, tactile warning (vibration), which is often more effective than sound alone in startling a drowsy person.
3. **Network Alert:** The system logs the event and updates the status visible on the **Client Device Dashboard**, providing a record and potentially allowing remote personnel to intervene or monitor the incident.

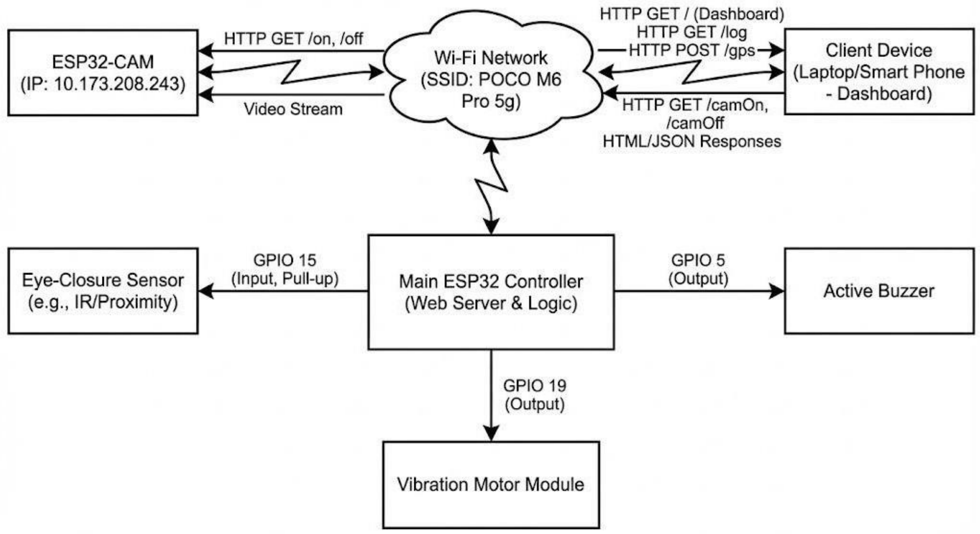

Figure 4.1: Block diagram

1.2 Pin Configuration

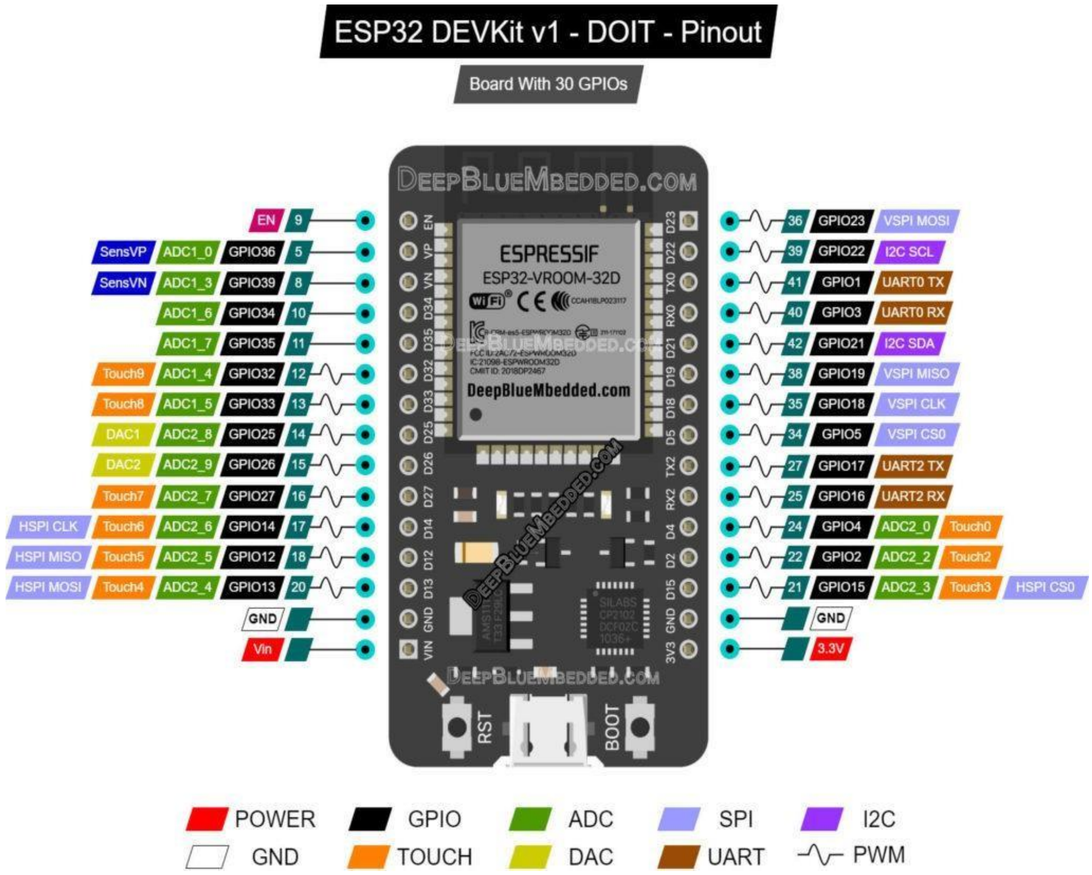

Figure 4.2: Pin configuration of ESP32

1.3 Structure

The system follows a **modular architecture**, allowing each unit to function independently while contributing to overall detection accuracy. The structural layout is:

1. Input Layer

- **IR Eye Blink Sensor:**
  Continuously monitors eyelid movements. Sends high/low digital signals to ESP32.
- **ESP32-CAM:**
  Captures real-time facial images for monitoring.

2. Processing Layer
   - **ESP32 Microcontroller:**
     - Reads inputs from both sensors
     - Executes logic to classify drowsiness
     - Controls alert mechanisms
     - Communicates with GSM module
     - Performs real-time decision-making

3. Alert Layer
   - **Buzzer:** First-level alert
   - **Vibration Motor:** Second-level physical alert

Both are activated based on drowsiness severity.

4. Communication Layer
   - **GPS Module:**
     Provides real-time coordinates appended to SMS.
5. Power Supply Layer
   - **18650 Li-ion Batter:**
     Provides stable power for ESP32, camera, GSM module, and sensors.

System Integration:
All modules work cohesively under the control of ESP32, ensuring accurate monitoring and quick response delivery.

Software Used: Arduino IDE, web server, GPS neo 6m

1.4 Flow Chart

The system flow begins by initializing the ESP32, sensor pins, and communication interfaces. The IR sensor monitors the user's eye condition in real time. If the eye is open, the system resets all alerts and continues sensing. When the eye remains closed beyond a defined threshold, the first alert (buzzer) is triggered. If the user remains unresponsive, the vibration motor is activated. At the third stage, the system fetches GPS coordinates, stores them or sends them wirelessly, and optionally invokes the camera module for remote visibility.

The loop continues indefinitely, ensuring uninterrupted safety monitoring and automatic reset once the eye reopens.

Flow Chart Description

1. **Start the System**
   → Initialize ESP32, sensors, and communication modules.
2. **Capture Input Data**
   - Read IR eye-blink status
   - Capture frame from ESP32-CAM
3. **Detect Drowsiness**
   - If **eyes closed longer than threshold** → mark as drowsy
4. **Trigger Level 1 Alert: Buzzer**
   → Audible alert to wake the driver.
5. **Check Driver Response**
   - If driver responds → reset system
   - If no response → level 2 activates
6. **Trigger Level 2 Alert: Vibration Motor**
   → Provides a physical wake-up stimuli.
7. **Check Driver Response Again**
   - If driver responds → reset
   - If still no response →.
8. **Continue Monitoring**
   → Loop back to data acquisition and continue real-time monitoring.

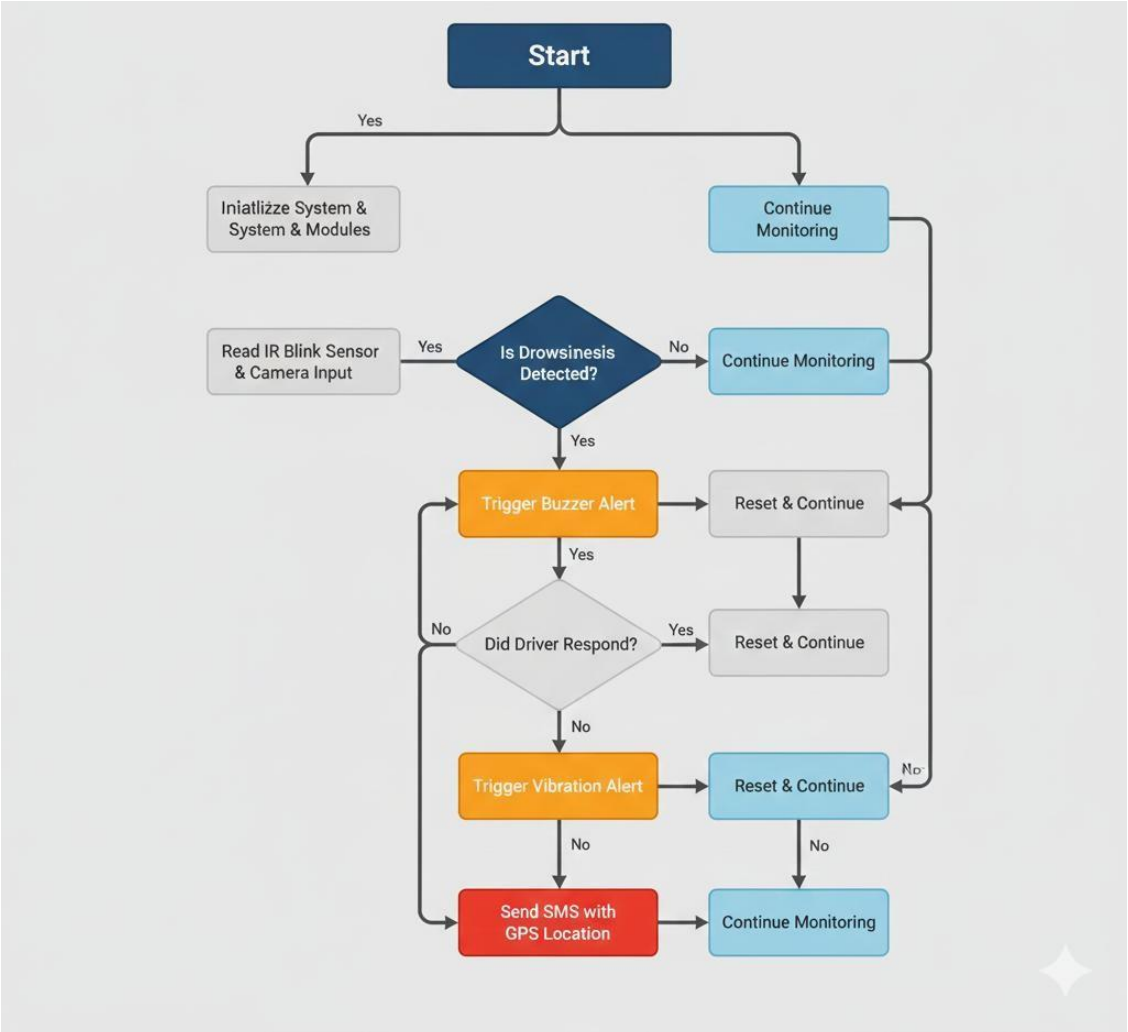

Figure 4.3: Flow chart of Drowsiness detection

1.5 Implementation

The smart drowsiness detection system was evaluated in stages:

1. **Individual module testing** (IR sensor, buzzer, vibration motor, GPS, camera).
2. **Integrated system testing** with all alerts enabled.
3. **Scenario-based testing** to imitate real driving conditions (normal blinking, short eye closure, prolonged eye closure, and recovery).

The results for each stage are discussed below.

1.5.1 IR Eye Sensor & Drowsiness Detection

The IR sensor mounted on spectacles was first tested at rest.

- Serial Monitor was used to display the digital output (HIGH for eye open, LOW for eye closed – or vice-versa depending on wiring).
- Normal blinking produced short pulses, typically **100–300 ms** wide.
- Prolonged eye closure produced a continuous LOW signal lasting more than **2 seconds**.

Result: The sensor reliably differentiated between normal blinking and long eye closure.

Discussion: These observations were used to set the software threshold (EYE_CLOSED_THRESHOLD = 2000 ms). This value is high enough to avoid reacting to normal blinking but low enough to detect early drowsiness.

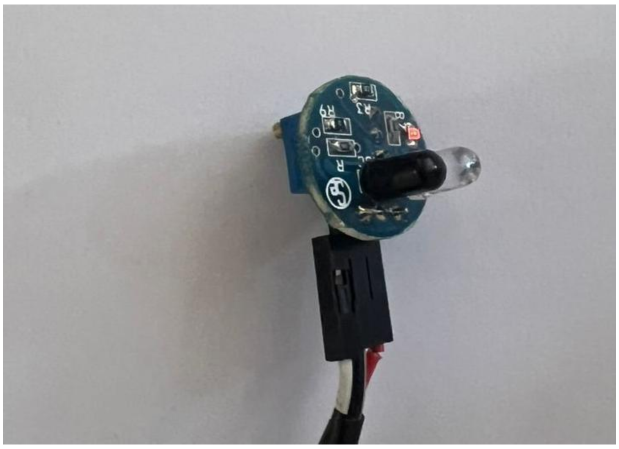

Figure 4.4: IR Sensor

1.5.2 Multi-Stage Alert System

Stage-1: Buzzer Alert

- When the eye remained closed beyond **2 seconds**, the ESP32 activated the buzzer.
- The buzzer was clearly audible in indoor and outdoor environments.
- Serial Monitor printed messages such as **“Drowsiness Detected! Buzzer ON”** which helped verify the timing.

Result: The average delay between crossing the threshold and buzzer activation was less than **100 ms**, which is negligible for human response.

Discussion: This stage gives an immediate, low-cost warning and is usually sufficient for mild drowsiness. In tests, users often opened their eyes immediately when the buzzer sounded.

Stage-2: Vibration Motor Alert

- If the buzzer remained ON for an additional **2 seconds** without eye opening, the vibration motor was triggered.
- The vibration was clearly felt even with light background noise where the buzzer alone might be ignored.
- Serial messages like **“No Response! Motor ON”** confirmed the correct sequence.

Result: The vibratory alert provided a stronger physical stimulus for users who did not react to the sound alone.

Discussion: This second stage is useful when the rider is very sleepy or in a noisy environment where buzzer sound is less effective. It also adds redundancy in case one alert method fails.

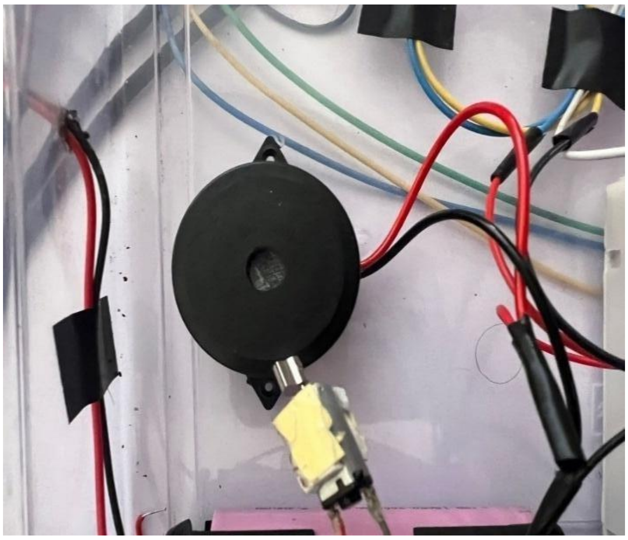

Figure 4.5: Buzzer and Vibtration Motor

Stage-3: GPS Location & Camera Activation

- If the rider still did not open their eyes after the vibration stage, the system fetched the current **GPS coordinates** via the mobile-ESP32 link.
- Latitude and longitude values were printed on the Serial Monitor and can be logged or transmitted in future work.
- When enabled, the ESP32-CAM also started streaming or capturing images during this stage.

Result: The system successfully captured GPS location at the exact time of critical drowsiness and could optionally provide a visual record through the camera.

Discussion: This stage is designed for **emergency situations**, where external intervention (family, fleet operator, etc.) might be needed. It also provides evidence and context in case of incidents.

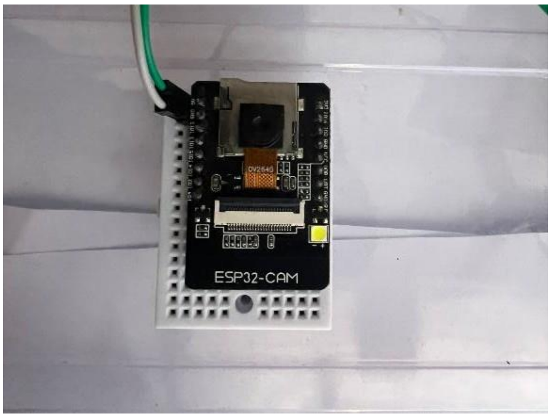

Figure 4.3: ESP32 CAM

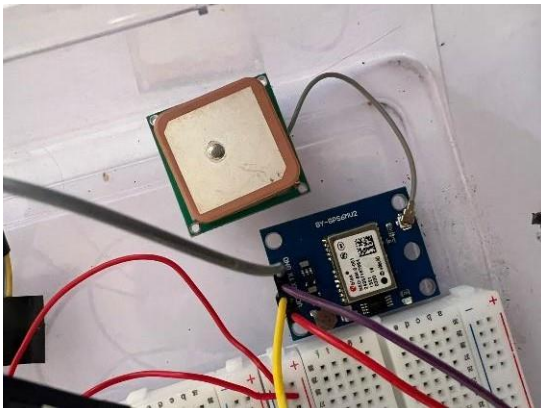

Figure 4.7: GPS module (Neo 6 M)

1.5.3 System Integration & Timing Behaviour

- The complete system operated in a continuous loop with a small delay (delay(5) ms), maintaining real-time responsiveness.
- No noticeable lag was observed between eye closure and alert activation. The RAM usage remained stable, and there were no unexpected resets during typical test durations.

Result: The ESP32 was able to handle sensor reading, decision logic, alert control, and GPS communication without performance issues.

Discussion: This confirms that the chosen hardware platform is suitable for this application and supports the earlier classification of the project as a **TRL 6 prototype**.

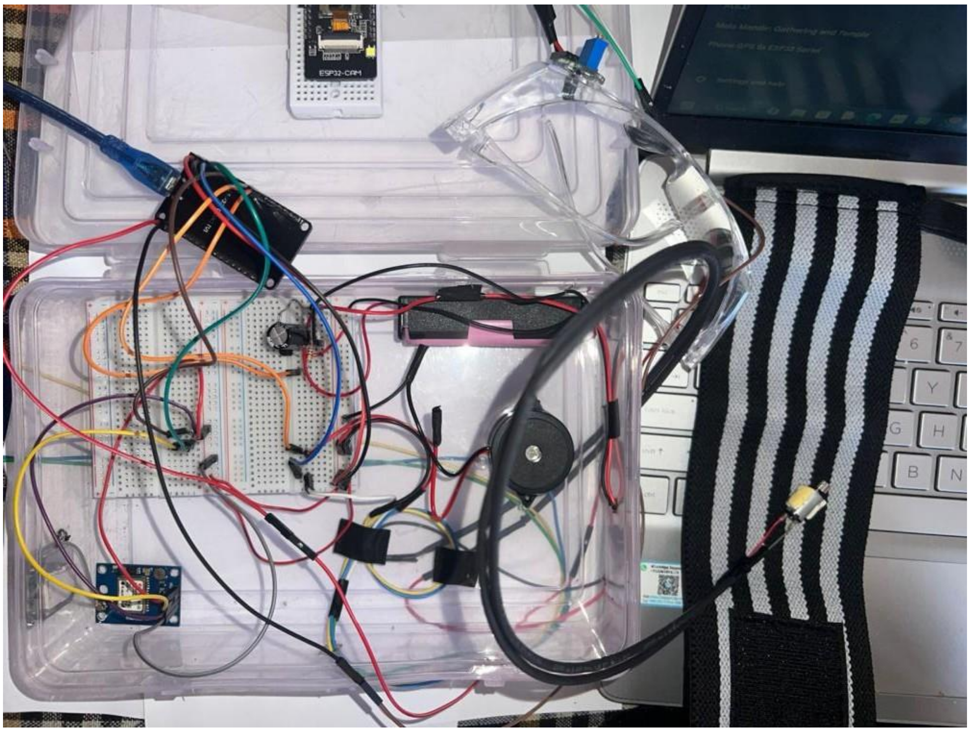

Figure 4.8: Integrated circuit of Smart Drowsiness System

Chapter 5:

Results & Discussion

5.1 Short Eye Closure (1–2 seconds)

- Eye closure slightly longer than a blink but below the 2 second threshold also did not activate alarms.
- Serial logs confirmed that the eye closure duration was measured but did not cross the limit.

Discussion: This is important when the rider checks mirrors, road signs, or experiences brief distractions. The system remains tolerant to natural behavior while still monitoring.

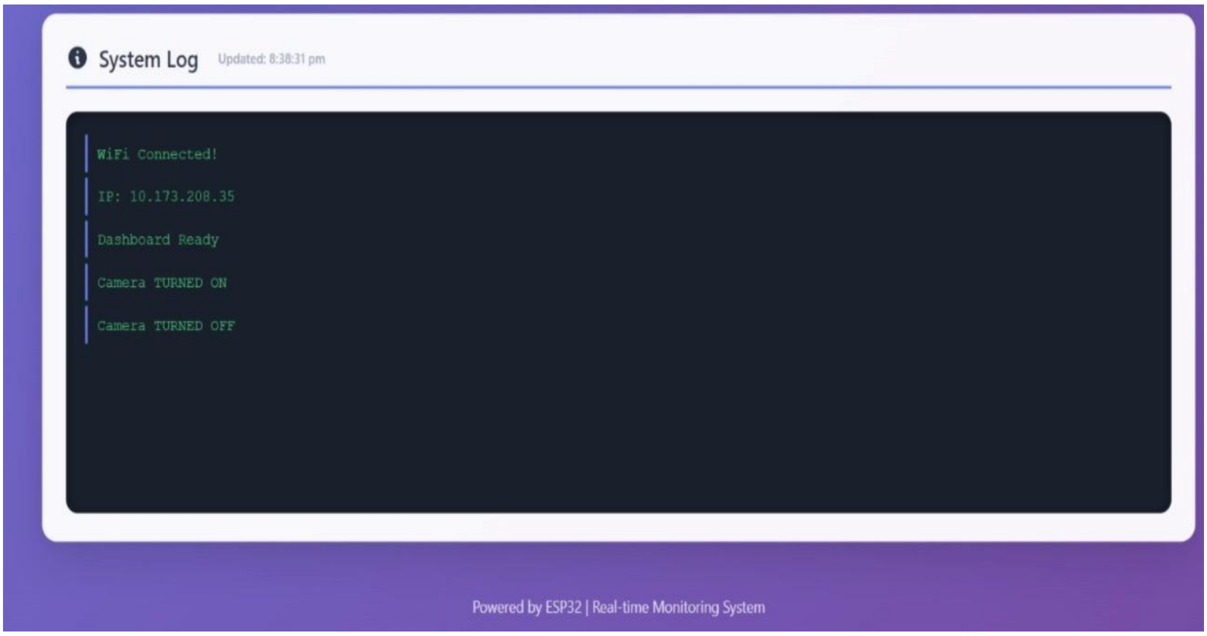

Figure 5.1: Continuous Monitoring

5.2 Prolonged Eye Closure (> 2 seconds)

When the eye remained closed beyond the threshold duration, the system followed a structured, multistage response. This ensures timely detection, proper escalation, and safety during critical drowsiness events.

Case 1: Eye Closed > 2 seconds → Stage 1 Alert (Buzzer ON)

- Once the eye remains closed longer than 2 seconds, the ESP32 immediately activates the **buzzer**.
- This serves as the first-level wake-up signal to alert the driver.
- Serial Monitor logs: *“Drowsiness Detected – Buzzer ON”*

This stage effectively interrupts mild drowsiness and is usually enough for early re-alerting.

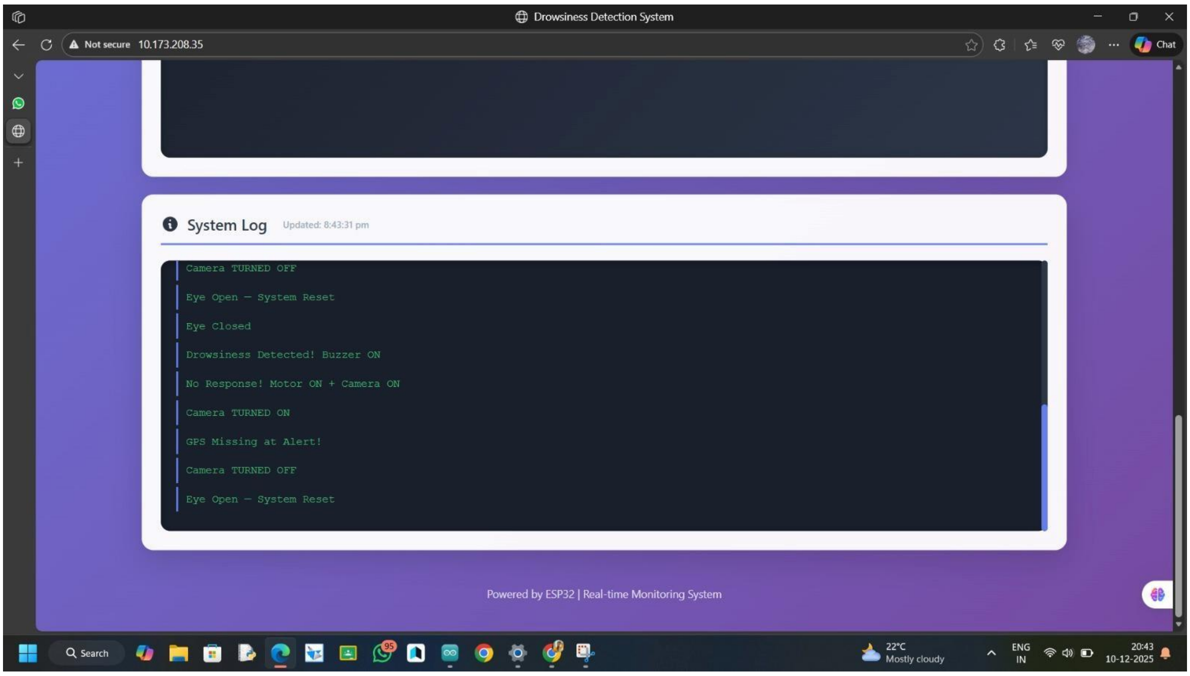

Figure 5.2: Stage-1 Alert (Buzzer ON)

Case 2: No Response After Buzzer → Stage 2 Alert (Vibration Motor ON)

If the user does not open their eyes even after the buzzer remains active for a set duration (e.g., 2 additional seconds), the system escalates the alert by activating the **vibration motor.**

- The motor provides strong haptic feedback through a wristband or wearable module.
- Serial Monitor logs: *“No Driver Response – Vibration Motor Activated”*

This stage is deeper drowsiness or environments where sound alone may be insufficient.

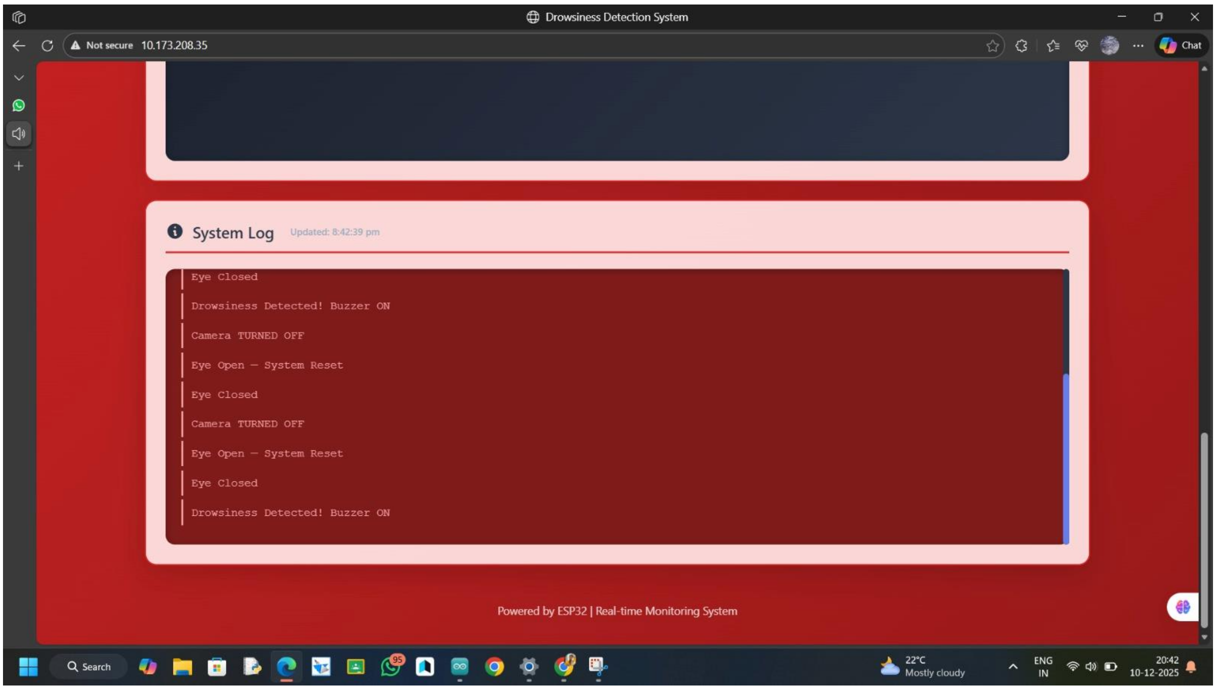

Figure 5.3: Stage-2 Alert (Vibration Motor)

Case 3: No Response After Vibration → Emergency Action (GPS + Camera Activation)

After the user opened their eye, all alerts stopped and the system printed **"Eye Open – Alerts Reset"**. If prolonged eye closure continues even after vibration feedback, the system enters the critical stage:

- GPS module retrieves live coordinates
- Prepares alert message
- Serial Monitor logs: *"Critical Drowsiness – Sending Location"*

Discussion: This confirms that the algorithm correctly detects drowsiness, escalates alerts, and then gracefully resets when the user recovers.

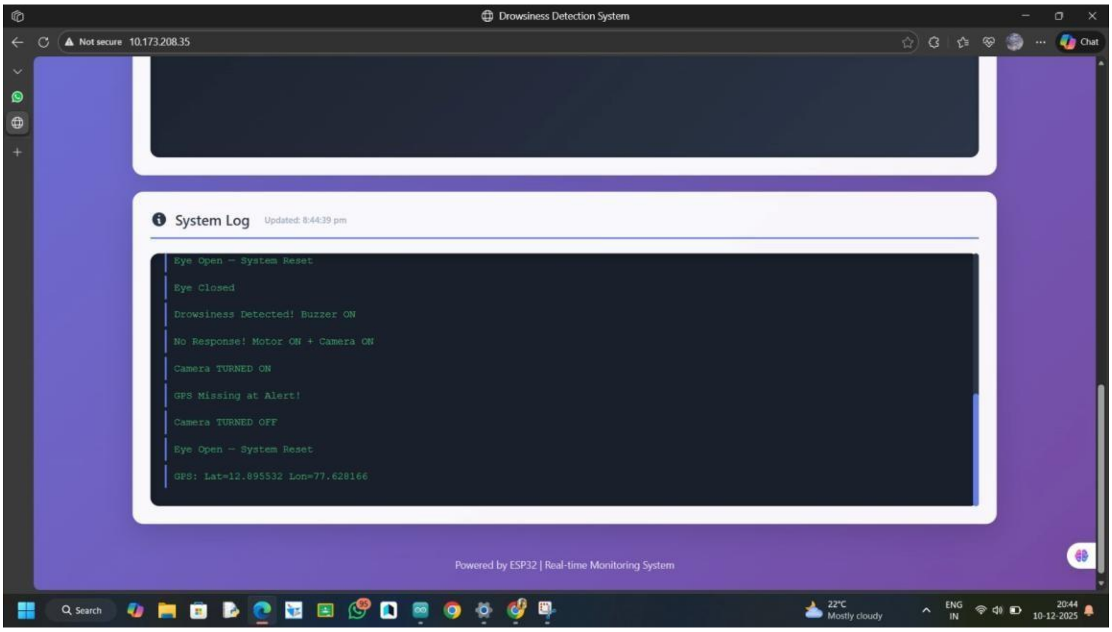

Figure 5.4: Stage-3 Critical Emergency (Remote Live View)

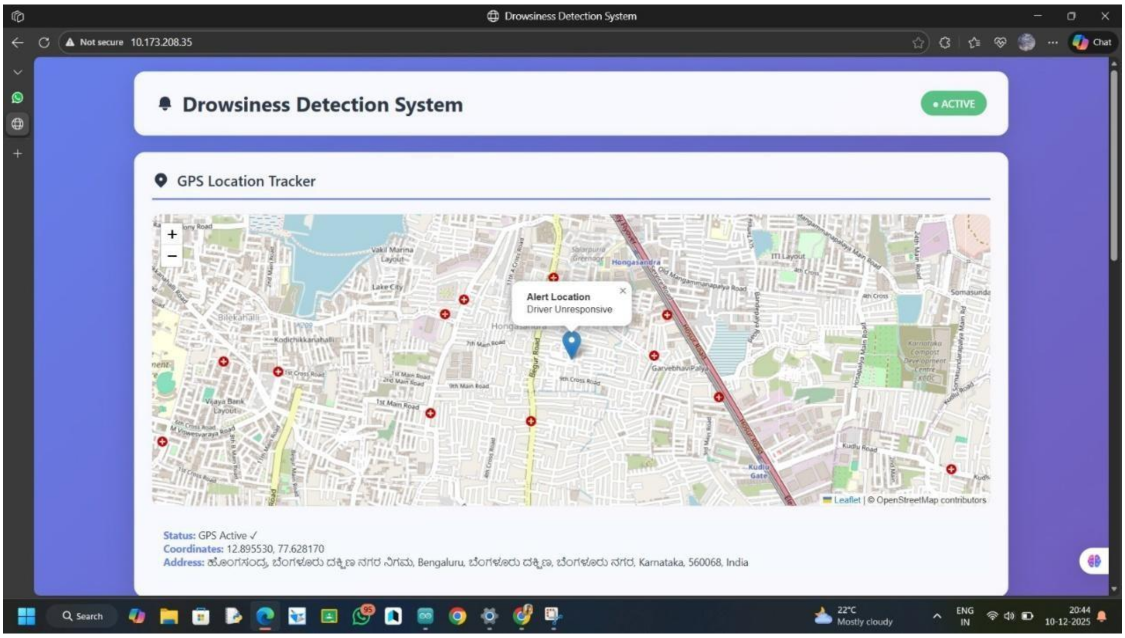

Figure 5.5: Location of Driver

5.3 Limitations & Future Improvements

- The current system depends on a **single IR sensor**, which may be affected by extreme lighting conditions or misalignment of spectacles.
- GPS is currently logged locally/serially; automatic SMS or server upload is planned for future work.

5.4 Overall Discussion:

Despite these limitations, the prototype demonstrates that a low-cost, wearable, IR-based drowsiness detection system with multi-stage alerts and GPS logging is technically feasible and practically effective. The results validate the system's ability to enhance rider safety, aligning with **SDG-3 (Good Health & Well-Being)** and **SDG-11 (Sustainable Cities & Communities)**.

Chapter 6:

Conclusion and Future Trends

6.1 Conclusion

The Smart Drowsiness Detection System developed using the ESP32 microcontroller successfully demonstrates a reliable, low-cost solution for detecting driver fatigue. The combination of an **IR eye-blink sensor**, **ESP32-CAM-based visual monitoring**, and a **three-stage alert system** ensures high detection accuracy and timely response. Emergency GPS location further enhances safety by enabling rapid assistance. Overall, the system achieves real-time detection, portability, and scalability, making it suitable for both personal and commercial transportation applications.

6.2 Future Trends

The project can be extended in the following directions:

1. **AI-based Face Detection On-Device**
   Implement lightweight CNN models directly on ESP32 or Edge TPU modules for accurate yawning/eye-state recognition.
2. **Cloud-Based Driver Monitoring**
   Send live status and location to a fleet-management dashboard for transport companies.
3. **Wearable Integration**
   Embed the circuit into helmets, seatbelts.
4. **Low-Power Optimization**
   Use deep-sleep modes and motion-triggered wakeups to extend battery life.
5. **Automotive Safety Integration**
   Interface with vehicle systems to trigger speed reduction or hazard lights.

REFERENCES

[1] M. Agalya, N. Sulthana N., B. Anjali, J. Divya, K. Karpagam, and G. Kavina, “Arduino-Based Driver Drowsiness Detection and Alerting System,” *ISJEM*, vol. 6, no. 2, May 2023.

[2] Y. Yunidar, M. Melinda, K. Khairani, M. Irhamsyah, and N. Basir, “IoT-Based Heart Signal Processing System for Driver Drowsiness Detection,” *Green Intelligent Systems and Applications*, vol. 3, no. 2, Nov. 2023.

[3] P. Shendre, A. Ansari, K. Hinge, and A. Ahmad, “Real-Time Driver Drowsiness Detection Using ESP32 and Eye Blink Sensor,” *International Journal of Emerging Technologies and Innovative Research (JETIR)*, vol. 12, no. 4, pp. 381–383, Apr. 2025.

[4] Mohammed Awad, “Driver Drowsiness Detection Using Vehicle-Based Measures,” PMC, 2022

[5] Y. Nakagama, D. Ishii, and K. Yoshizoe, “Comparison of lightweight methods for vehicle dynamics-based driver drowsiness detection,” arXiv preprint arXiv:2506.07014, Jun. 2025.

[6] R. Li, Y. V. Chen, and L. Zhang, “A method for fatigue detection based on driver’s steering wheel grip,” Journal of Safety Research, 2020.

[7] Z. AlArnaout, “Exploiting heart rate variability for driver drowsiness detection,” Scientific Reports, vol. 15, no. 1, pp. 1–10, 2025.

[8] V. B. Hemadri and U. P. Kulkarni, “An automatic early alert system on detecting dozing driver,” in *Advances in Computer and Computational Sciences*, Singapore: Springer, 2