Vehicle Accident Control System

Suzan Vadgave, Saniya Kamble , Sneha Koravi , Roshani Kamble, Mr. Pravin Chavan
Students, Department of Electronics and Telecommunication Engineering
Guide, Department of Electronics and Telecommunication Engineering
Sanjay Ghodawat Institute, Atigre, India

Abstract: *Road accidents are a leading global concern, with human error—specifically drowsiness and impaired driving—being a primary cause. This project proposes the development of a Vehicle Accident Control System, a cost-effective and efficient solution to prevent accidents by monitoring the driver's condition in real time. The system uses an integrated combination of hardware and software components to detect unsafe driving behaviors and take immediate corrective actions. This system is built around an Arduino Nano, a compact and versatile microcontroller that processes data from multiple sensors and drives the vehicle's control mechanisms. The design includes: Eye Blink Sensor: Monitors the driver's eye movements to detect drowsiness. By analyzing the blink frequency and duration, the system identifies fatigue-related behaviors and triggers preventive measures. MQ3 Alcohol Sensor: Detects the presence of alcohol in the driver's breath. If the measured alcohol concentration exceeds a predefined threshold, the system assumes impairment and halts the vehicle. L298N Motor Driver: Controls two 100 RPM DC gear motors that simulate vehicle wheels. The motor driver receives commands from the Arduino Nano to start, stop, or control the vehicle's movement based on sensor inputs. 16x2 LCD Display: Provides real-time feedback to the user by displaying status updates and alerts such as "Alcohol Detected" or "Drowsiness Detected." Buzzer and Red LED: Serve as alert mechanisms to warn both the driver and nearby individuals about potential hazards. PCB and Jumper Wires: Facilitate compact and organized connections between components to ensure reliability and ease of assembly.*

Keywords: Road accidents

I. INTRODUCTION

Road safety is a growing global concern as accidents continue to claim lives and cause injuries. According to the World Health Organization (WHO), millions of people are involved in road traffic accidents annually, with a significant portion attributed to driver error. Among these errors, driver fatigue and alcohol consumption are two critical factors that lead to impaired driving ability.

In modern times, technology plays an essential role in addressing safety challenges. The integration of electronics, sensors, and microcontrollers in vehicles has paved the way for accident prevention systems. The Vehicle Accident Control System designed in this project utilizes such advanced technologies to monitor driver behavior in real-time, identify unsafe conditions, and take immediate corrective actions to prevent accidents.

II. REVIEW OF LITERATURE

The rise of vehicular accidents has prompted extensive research into preventive measures and technologies. Automated accident control systems have gained significant attention as they can actively monitor drivers and intervene to prevent potential accidents. This chapter reviews existing research, technologies, and systems that have addressed driver safety through fatigue and alcohol detection, vehicle control, and alert mechanisms. The gaps in these solutions that this project seeks to address are also highlighted.

III. METHODOLOGY

The project employs a systematic approach to achieve its objectives:

System Design:

- Identify hardware components required for vehicle motion simulation and driver monitoring.
- Create a logical flow of operations, from sensor data acquisition to motor control.

Hardware Implementation:

- Integrate components such as the eye blink sensor, MQ3 alcohol sensor, DC motors, and motor driver.
- Establish connections using a PCB and jumper wires to ensure compactness and reliability.

Software Development:

- Program the Arduino Nano to process sensor data and control motor operations.
- Display real-time feedback and alerts on the LCD. Testing and Optimization:
- Simulate various driving conditions, such as normal driving, drowsiness, and alcohol
- Optimize the system for accurate detection and reliable performance.

IV. WORKING

The system operates in real-time, continuously monitoring the driver's condition.

Driver Monitoring: The eye blink sensor checks for irregular blink patterns, such as prolonged eye closure, which indicate drowsiness. Similarly, the MQ3 alcohol sensor evaluates the alcohol content in the driver's breath.

Data Processing: Data from the sensors are transmitted to the Arduino Nano, which processes the inputs against predefined thresholds for drowsiness and alcohol impairment.

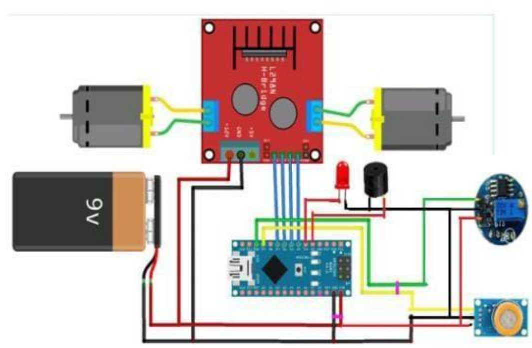
Control Actions:

If unsafe conditions are detected, the Arduino sends signals to the L298N motor driver to halt the DC motors, effectively stopping the vehicle.

The system activates the buzzer and LED to alert nearby individuals and enhance safety.

The LCD displays the reason for the intervention, such as "Driver Drowsy" or "Alcohol Detected."

Circuit Diagram



Main Component

- Arduino Nano
- L298N Motor Driver
- Eye Blimk Sensor
- MQ3 Alcohol Sensor
- Buzzer and Gear Motor
- 2DC Gear Motor
- PCB and Jumper W

Application

Prototyping of IoT devices

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.67
Volume 5, Issue 7, March 2025

Low power battery operated applications
Network projects
Projects requiring multiple I/O interfaces with Wi-Fi and Bluetooth functionalities
NodeMCU is a low-cost open source IoT platform. It initially included firmware which runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware which was based on the ESP-12 module. Later, support for the ESP32 32-bit MCU was added.
NodeMCU is an open source firmware for which open source prototyping board designs are available. The name "NodeMCU" combines "node" and "MCU" (micro-controller unit).The term "NodeMCU" strictly speaking refers to the firmware rather than the associated development kits.[citation needed]
Both the firmware and prototyping board designs are open source. The firmware uses the Lua scripting language. The firmware is based on the eLua project, and built on the Espress if Non-OS SDK for ESP8266. It uses many open source projects, such as lua-cjson and SPIFFS. Due to resource constraints, users need to select the modules relevant for their project and build a firmware tailored to their needs. Support for the 32-bit ESP32 has also been implemented
The prototyping hardware typically used is a circuit board functioning as a dual in-line package (DIP) which integrates a USB controller with a smaller surface-mounted board containing the MCU and antenna. The choice of the DIP format allows for easy prototyping on breadboards. The design was initially based on the ESP-12 module of the ESP8266, which is a Wi-Fi SoC integrated with a Tensilica Xtensa LX106 core, widely used in IoT applications

Code

```
#include <LiquidCrystal.h>
// Pin Definitions
#define EYE_BLINK_SENSOR_PIN 2     // Digital pin for Eye Blink Sensor
#define MQ3_SENSOR_PIN A0  // Analog pin for MQ3 Alcohol Sensor
#define MOTOR1_PIN1 3          // Motor 1 control pin 1
#define MOTOR1_PIN2 4          // Motor 1 control pin 2
#define MOTOR2_PIN1 5          // Motor 2 control pin 1
#define MOTOR2_PIN2 6          // Motor 2 control pin 2
#define BUZZER_PIN 7           // Buzzer pin #define LED_PIN 8 // Red LED pin

// MQ3 Alcohol Sensor Threshold
const int ALCOHOL_THRESHOLD = 400; // Adjust based on calibration

// Eye Blink Threshold (Adjust based on your sensor's output)
const int EYE_BLINK_THRESHOLD = HIGH;

// LCD Pins (RS, EN, D4, D5, D6, D7)
LiquidCrystal lcd(9, 10, 11, 12, A1, A2);

// Initialize Pins pinMode(EYE_BLINK_SENSOR_PIN, INPUT); pinMode(MQ3_SENSOR_PIN, INPUT); pinMode(MOTOR1_PIN1, OUTPUT); pinMode(MOTOR1_PIN2, OUTPUT); pinMode(MOTOR2_PIN1, OUTPUT); pinMode(MOTOR2_PIN2, OUTPUT); pinMode(BUZZER_PIN, OUTPUT); pinMode(LED_PIN, OUTPUT);

// Start with everything off stopMotors(); digitalWrite(BUZZER_PIN, LOW); digitalWrite(LED_PIN, LOW);
}

void loop() {
```

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.67
Volume 5, Issue 7, March 2025

```
// Read sensors
int eyeBlinkStatus          = digitalRead(EYE_BLINK_SENSOR_PIN);
int alcoholLevel = analogRead(MQ3_SENSOR_PIN);
// Check for drowsiness
if (eyeBlinkStatus == EYE_BLINK_THRESHOLD) { handleDrowsiness();
}
// Check for alcohol detection
else if (alcoholLevel > ALCOHOL_THRESHOLD) { handleAlcoholDetection();
void setup() {
// Initialize LCD lcd.begin(16, 2);
lcd.print("System Initializing"); delay(2000);
lcd.clear();
lcd.print("Monitoring Driver");
}
// Normal operation else {
handleNormalOperation();
}
delay(200); // Add a delay for stability
}

void handleDrowsiness() { stopMotors(); digitalWrite(BUZZER_PIN, HIGH); digitalWrite(LED_PIN, HIGH);
lcd.clear();
lcd.print("Drowsiness Detected");

digitalWrite(MOTOR2_PIN2, LOW);
}

}
void handleAlcoholDetection() { stopMotors(); digitalWrite(BUZZER_PIN, HIGH); digitalWrite(LED_PIN,
HIGH); lcd.clear();
lcd.print("Alcohol Detected");
}
void handleNormalOperation() { startMotors(); digitalWrite(BUZZER_PIN, LOW); digitalWrite(LED_PIN,
LOW); lcd.clear();
lcd.print("Monitoring Driver");
}
void stopMotors() { digitalWrite(MOTOR1_PIN1, LOW); digitalWrite(MOTOR1_PIN2, LOW);
digitalWrite(MOTOR2_PIN1, LOW); digitalWrite(MOTOR2_PIN2, LOW);
}
void startMotors() { digitalWrite(MOTOR1_PIN1, HIGH); digitalWrite(MOTOR1_PIN2, LOW);
digitalWrite(MOTOR2_PIN1, HIGH);
```

Advantages

Real-Time Monitoring:

The system continuously monitors the driver’s condition, ensuring prompt detection of drowsiness or alcohol impairment.

Accident Prevention:

By halting the vehicle in unsafe driving conditions, the system significantly reduces the likelihood of accidents caused by human error.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.67
Volume 5, Issue 7, March 2025

Cost-Effective:
The system is built using readily available and affordable components, making it accessible for widespread implementation.

Easy Integration:
The system can be integrated into existing vehicles with minimal modifications.

Compact Design:
The use of an Arduino Nano and other lightweight components allows for a compact and portable design.

User-Friendly Alerts:
The combination of an LCD display, buzzer, and LED ensures that warnings are easily noticeable by both the driver and surrounding individuals.

Disadvantages

Sensor Accuracy Limitations:
The system's performance depends on the calibration and sensitivity of the sensors. False positives or negatives may occur.

Environmental Sensitivity:
The MQ3 alcohol sensor's readings may be influenced by environmental factors such as temperature or other volatile substances.

Limited Range:
The eye blink sensor requires precise positioning to effectively monitor the driver's eye movements.

Not Foolproof:
The system may fail to account for other accident- causing factors such as mechanical failures or external hazards.

Dependency on Thresholds:
Fixed thresholds for alcohol detection and drowsiness may not cater to individual differences among drivers.

Prototype Limitation:
The system, in its current form, is a prototype and may require further refinement for commercial use.

V. FUTURE SCOPE

The Vehicle Accident Control System has immense potential for further development and integration into modern vehicles. Future enhancements could include:

Improved Sensor Accuracy:

- Advanced eye-tracking technologies, such as infrared cameras, can enhance drowsiness detection.
- Enhanced alcohol sensors with better environmental stability.

Integration with IoT:

- Connecting the system to cloud platforms for real-time monitoring and data analysis.
- Enabling remote alerts to emergency services or family members in case of unsafe driving conditions.

AI and Machine Learning:

- Incorporating machine learning algorithms to adapt thresholds based on individual driver behavior and environmental factors.
- Detecting additional unsafe conditions such as stress, distraction, or health emergencies.

Expanded Features:

- Adding GPS tracking and accident location reporting. Including a speed limiter to further enhance safety.

Enhanced Display and Alerts:

- Using larger and more sophisticated displays for improved communication.

VI. CONCLUSION

The project demonstrates a practical and effective approach to reducing accidents caused by driver fatigue and alcohol impairment. The system's successful operation showcases the potential of integrating affordable, easily available components with modern microcontroller technology to enhance vehicle safety.

Key highlights:

- Real-time monitoring and immediate response to unsafe driving conditions.
- Cost-effective and compact design for easy implementation in various vehicle types.
- Enhanced safety through a combination of sensory inputs, decision-making algorithms, and control mechanisms.
- This prototype serves as a proof of concept, highlighting the feasibility of such systems in real-world applications.

REFERENCES

[1]. K. L. James, Microcontroller Systems for Embedded Applications, Cengage Learning, 2016.
[2]. A P. Godse and D. A. Godse, Arduino Programming for Beginners, Technical Publications, 2021.
[3]. T. Martin and K. Barret, Introduction to Embedded Systems: Using Arduino, Springer, 2019.
[4]. https://www.arduino.cc/en/Guide/ArduinoNano Description: Official documentation for setting up and programming the Arduino Nano.
[5]. https://www.st.com/resource/en/datasheet/l298.pdf
[6]. Description: Datasheet for the L298N motor driver module, detailing its functionality and wiring.
[7]. https://components101.com/sensors/mq3-alcohol- gas-sensor
[8]. Description: Detailed technical specifications, working principles, and interfacing instructions for the MQ3 sensor.
[9]. https://robu.in/product/eye-blink-sensor-module Description: Product guide and specifications for the eye blink sensor module used in the project.
[10]. https://www.arduino.cc/en/Tutorial/HelloWorld Description: Official Arduino .