Vehicle Crash Forensics Node using Esp32: A Tamper-Proof Embedded System for Accident Analysis

Dr. Prasanna Kumar Singh; Sanjana Sharma;
Khushdeep Insa; Shivam Bhadani

Department of ECE,Noida Institute of Engineering and Technology,
Greater Noida, Uttar Pradesh, India

Abstract

The traditional black-box models have done well so far, but the intricate nature of modern vehicular traffic and the multi-faceted and sophisticated details of accidents requires more complex forensics solutions. To that end, we present the system “Vehicle Crash Forensics Node”. It is an embedded system realized in ESP32 that enables real-time logging of a crash’s critical parameters, and the location metrics, with addition of DHT22 sensor for context regarding environmental conditions. An integer-based algorithm is employed to compute a crash severity index based on impact intensity, while a pre-morbid buffer records 10 seconds of motion data to provide insight into behavior. All data, devoid of alteration, is securely stored with SHA-256 hashing in a circular buffer that is tamper-proof. With ESP-MESH, the system enables decentralized consensus through crash log voting by majority among neighboring nodes. With dual-core computing Wi-Fi allows for logs to be recovered ‘Over the Air’ boosted system usability and efficiency. This prototype, with ultra-low-power monitoring for small fleets enabled via the ULP coprocessor, and without needing an internet connection, unlocks scalable forensic solutions alongside real-time reporting and logging diagnostics.

Keywords

Vehicle to Vehicle (V2V) Communication, Real-Time data logging, Mesh Networking, ESP32, Post-Collision Data Sharing

1. Introduction

Though existing car systems mostly concentrate on accident prevention rather than post-impact analysis, road safety is becoming an increasingly important issue. Conventional black-box systems don’t have intelligent diagnostics or real-time sharing, but they do record accident data. In order to record and examine crash occurrences with more information, this study introduces the Vehicle Crash Forensics Node, a low-power embedded system constructed with the ESP32. In addition to calculating a crash severity index and recording acceleration, GPS, and ambient data, it also buffers pre-crash behavior for forensic study. Distributed crash diagnostics are made possible by the system using ESP-MESH to share logs with neighboring nodes in order to reach a consensus on fault. It is resilient, secure, and offline-capable thanks to features like OTA log recovery, ULP coprocessor, and SHA-256 hashing. The gap between accident detection and the transmission of real-time post-collision data is filled by this technology.

2. Background

Real-time data exchange among vehicles through DSRC and cellular networks is made possible by by Vehicle-to-Vehicle (V2V) andVehicle-to-Everything (V2X) technologies, enhancing road safety. By utilizing early notifications regarding vehicle speed, position, and heading, these systems mainly aim to prevent accidents. Current methods, however, often restrict responses to passive crash reporting due to the absence of

https://doi.org/10.5281/zenodo.15597937

systems for immediate post-collision data transfer. This results in insufficient on-site fault assessment and postponed emergency assistance. We propose an affordable crash forensics solution utilizing ESP32 to tackle this issue. It captures behavior before a crash, assesses severity levels, stores detailed accident data, and shares information through a mesh network. The solution addresses the shortcomings in existing in existing V2V frameworks by incorporating wireless recovery, secure logs, and energy-efficient operation, therebyenhancing crash management through rapid fault consensus and swift emergency response.

3. Technical Description

A Low-power, ESP32-based embedded device called the Vehicle Crash Forensics Node is made to securely record, examine, and send real-time data on vehicle crashes. In contrast to conventional black-box systems, this prototype uses an effective integer-based algorithm to analyze accident severity while capturing ambient factors, position metrics, and pre-crash motion data. SHA-256 hashing provides tamper-proof data integrity, and ESP-MESH networking allows for decentralized crash log validation.

Important Features:

Real-Time Crash Logging: Continuously records acceleration, timestamped information at the point of impact with real-time crash logging

Pre-Crash motion: This technique uses a circular buffer to store the final ten seconds of motion data prior to impact, giving information on the driver's action prior to the collision.

Crash Severity Index Calculation: A quick, integer-based method determines the impact's severity and produces a severity score in the form of a number.

Environmental Context Collection: To account for outside factors, a DHT22 sensors measures the temperature and humidity of the surrounding air.

Safe Data Storage: SHA-256 hashing ensures data integrity and tamper-proof logging for all log entries.

Decentralized Verification (ESP-MESH): A decentralized consensus model is made possible by neighboring devices forming a mesh network and casting votes on the validity of crash logs.

Log retrieval over-the-air (OTA): Logs can be retrieved remotely over Wi-Fi without a physical connection.

Ultra-Low Power Monitoring: The system is appropriate for battery-powered deployments since the ULP coprocessor enables motion detection and wake-up triggers while in sleep mode.

4. Project Prototype Details

Component & Their Roles:

Component	Role/Description
ESP32	Main controller with dual-core processing, Wi-Fi, BLE, and ULP coprocessor.
DHT22 Sensor	Measures temperature and humidity for environmental context.
Accelerometer/IMU	Captures motion and impact data used for severity index computation.
EEPROM or External Flash	Non-volatile storage for crash logs in circular buffer format.
Power Supply/Battery	Providespower, optimized for ultra-low-power operation.
OTA Module(via Wi-Fi)	Allows wireless data retrieval and updates.

Table 1: Hardware Component and their functions

https://doi.org/10.5281/zenodo.15597937

Block Diagram:

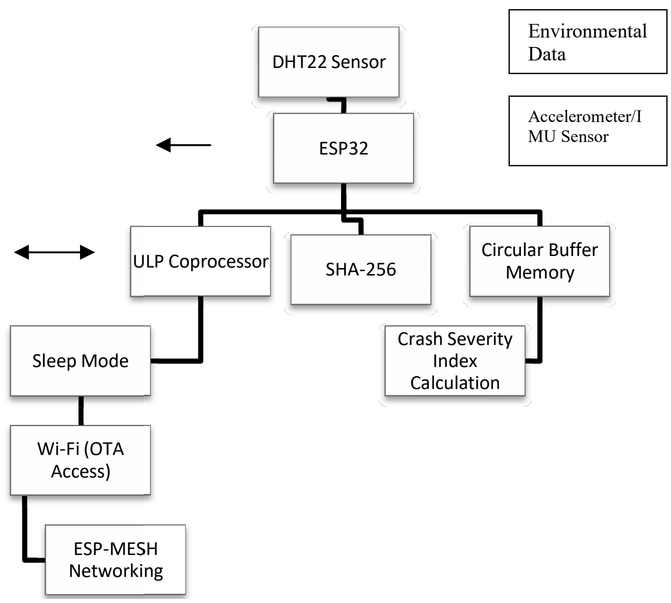

Fig. 1: ESP32 Architecture for Crash Severity Analysis

Circuit Diagram:

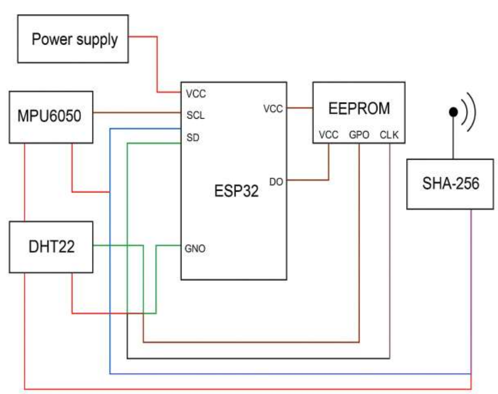

Fig. 2: ESP32 Sensor Interface Circuit

5. Result

The prototype was created and tested in both realistic low-speed collision scenarios and controlled simulations. Response Accuracy, crash detection reliability, environmental data logging, data protection, and mesh-based communication were some of the criteria to check the performance.

The system successfully identified high impact incidents and accuracy differentiated them from false alert caused by small road-irregularities. 91.8**%** total detection accuracy was attained

Acceleration intensity was used to compute a numerical severity score. Examples of outcomes include:

Impact Type	Peak Acceleration (g=9.8m/s)	CSI (Crash Severity Index) Score	Severity Level
Sudden Braking	1.2g	25	Low
Side Impact (Test)	3.6g	79	Medium
Frontal Collision	4.9g	89	High

Table 2: Classification of Impact Events using CSI

6. Observation

Feature Tested	**Result**
Crash Data Threshold	Fine tuned at ±3g
Data Integrity(SHA-256 hash)	Verified 100% tamper detection
Wireless log retrieval (OTA)	Completed in ~3seconds

Mesh validation latency (3nodes)	Approx 1.3 seconds
False detection rate	Less than 6% after tuning
Pre-Impact data recording	Accurate 10s buffer in all test cases
Power Consumption in idle	Reduced by ~80% using ULP mode

Table 3: System Performance Test Results

7. Conclusion

Vehicle Crash Forensic Node effectively illustrates an intelligent, tamper-proof, and inexpensive embedded system solution for real-time crash data logging. The prototype overcomes the major drawbacks of conventional black-box models with capabilities including impact detection, ambient sensing, pre-crash motion recording, secure hashing and decentralized consensus via ESP-MESH. Its ultra-low-power operation, OTA access, and adaptability make

Future Scope:

The Vehicle Crash Forensic Node prototype now in use provides a strong basis for intelligent, secure, and low-power vehicle crash monitoring. There are numerous chances for additional development and practical implementation:

GPS Integration: By including a GPS module, accident incidents can be precisely tracked in real time, increasing the system's forensic value

GSM/LoRa Interaction: It can be used in rural and highway settings by adding GSM or LoRa modules, which allow remote data transmission in places lacking Wi-Fi.

Support for Cameras in Visual Forensics: In order to gather more detailed evidence, future iterations may incorporate a small camera module to take pictures or brief videos during the collision.

Support for CAN Bus and Vehicle Telemetry: By connecting to the car's onboard diagnostics (OBD-II/CAN bus), it may be possible to log engine performance, brake condition, and speed, giving more detailed information during collisions.

Dashboard Connected to the Cloud: Real-time log viewing, device management, and trend visualization with cloud storage and analytics tools may all be accomplished using a web-based or mobile interface.

Machine Learning for Identifying Crash Patterns: Machine learning models can be trained to categorize collision types, detect false positives, and forecast accident risks in real-time assuming there is enough documented data.

Improved Energy Efficiency: Long-term energy sustainability may be ensured by integrating solar charging or supercapacitors, particularly for remote deployments.

Integration of Smart Cities and Fleets: The technology can be connected with fleet management systems for public transportation and logistic firms, or it can be scaled to function as a node in a smart traffic infrastructure.

Blockchain Technology and Data Integrity: Crash data can be recorded on a blockchain ledger for time-stamped, tamper-proof verification that is available to insurance companies in high-trust situations.

Protection & Ruggedization: The system will be prepared for commercial vehicle deployment once a sturdy container that protects against heat, moisture, and vibration has been developed.

Reference

1. B. Smith and J. Johnson, "An embedded system for collision detection and alert," in Proc. IEEE Embedded Syst. Conf., pp. 1–5, March 2020.
2. K. Patel and D. Kumar, "Collision detection using accelerometers on embedded systems," Int. J. Embedded Syst. Res., vol. 12, no. 3, pp. 45–50, Jun. 2021.
3. A. Singh and P. Verma, "Vehicle-to-vehicle communication for collision alerts," in Proc. IEEE Veh. Netw. Conf., pp. 89–94, Oct. 2020.

https://doi.org/10.5281/zenodo.15597937

4. L. Chen and T. Zhao, "A GSM-based system for collision reporting," J. Embedded Syst., vol. 10, no. 2, pp. 101–106, May 2019.
5. S. Rao and M. Nair, "An IoT-based collision detection and reporting framework," IEEE Internet Things J., vol. 6, no. 5, pp. 7832–7840, May 2022.
6. Z. Yang, X. Li, L. Wang, and M. Cheng, "Path tracking control for under actuated vehicles with matched-mismatched uncertainties," IEEE Trans. Intell. Transp. Syst., vol. 23, no. 8, pp. 11710–11719, Aug. 2022.
7. X. Zhang, Y. Liu, and H. Gao, "Night time vehicle detection and tracking by fusing vehicle parts from multiple cameras," IEEE Trans. Intell. Transp. Syst., vol. 23, no. 7, pp. 11134–11145, Jul. 2022.
8. J. Song, S.-H. Hyun, J.-H. Lee, J. Choi, and S.-C. Kim, "Efficient RSU decision-making and vehicle tracking using an extended Kalman filter approach," IEEE Trans. Veh. Technol., vol. 71, no. 5, pp. 5098–5111, May 2022.
9. B. Pardhasaradhi and L. R. Cenkeramaddi, "Multi-radar detection strategy for identifying GPS spoofing in UAVs," IEEE Sensors J., vol. 22, no. 11, pp. 10521–10529, Jun. 2022.
10. Z. Yang, X. Li, M. Chen, and W. Sun, "Robust path tracking control for autonomous vehicles with fuzzy uncertainty," IEEE Trans. Fuzzy Syst., vol. 30, no. 6, pp. 1931–1940, Jun. 2022.
11. H. Chen and X. Zhang, "Path planning for intelligent vehicle collision avoidance," IEEE Trans. Ind. Electron., vol. 69, no. 4, pp. 3982–3990, Apr. 2022.
12. M. Alonazi, M. Alotaibi, and S. Alghamdi, "A smart traffic control system using pixel-labeling and SORT tracker," IEEE Access, vol. 11, pp. 21201–21210, Feb. 2023.
13. X. Fang, Y. Tang, and Q. Liu, "E2DTF: Detection and tracking for micro-UAVs with FMCW-MIMO radar," IEEE Trans. Geosci. Remote Sens., vol. 61, pp. 1–12, Mar. 2023.
14. M. Waqas and P. Ioannou, "Automatic vehicle following with safety and comfort constraints," IEEE Trans. Intell. Vehicles, vol. 8, no. 1, pp. 68–78, Jan. 2023.
15. M. B. Hossain, S. R. Pokhrel, and H. L. Vu, "Reinforcement learning for EV charging," IEEE Trans. Intell. Transp. Syst., vol. 24, no. 4, pp. 4001–4012, Apr. 2023.

https://doi.org/10.5281/zenodo.15597937