##### **A Reverse Guide System for Automobiles Using CAN Protocol**

##### **Noor Ayesha\*, Ganesh Kumar\*, Gireesh A\*, Ajith H R\*, Darshan B S\***

\* Department of Electronics and Communication Engineering, P.E.S. College of Engineering,

Mandya, Karnataka, India

***Abstract*-** This paper presents a low-cost, Arduino-based reverse guidance system for automobiles, utilizing the Controller Area Network (CAN) protocol for reliable communication. Designed for challenging reverse parking scenarios due to limited visibility, the system incorporates three HC-SR04 ultrasonic sensors, two MCP2515 CAN controller modules, two Arduino UNO boards, a 0.96-inch OLED display, and a buzzer. Sensors detect rear obstacles, sending proximity data via CAN frames to the display unit, providing both visual and auditory feedback to the driver. The communication relies on standard CAN frames with firmware-based encoding and decoding logic. A functional prototype was successfully implemented, with the complete design and source code available on GitHub:<https://github.com/Amazing-Stardom/arduino-reverse-car-can-system>. This solution offers an affordable and practical reverse guidance system for entry-level vehicles.

***Index Terms***- CAN Protocol, Embedded Systems, HC-SR04, MCP2515, Reverse Parking Assistance, Vehicle Safety

1. Introduction

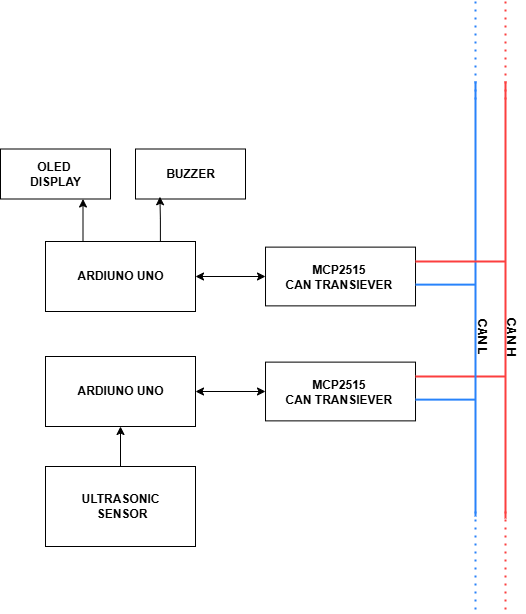
Reverse parking remains a significant driver challenge in congested urban areas. While modern luxury vehicles offer advanced parking assist systems using radar or cameras, budget vehicles often lack these features due to cost constraints. This project addresses the issue by proposing a practical and affordable reverse guidance system for entry-level vehicles. Using CAN protocol—a proven solution in automotive systems for its robustness and real-time capability [2]—the project connects a sensor node and a display node over a twisted-pair CAN bus. Ultrasonic sensors detect distance, which is classified into five levels and displayed visually, accompanied by dynamic buzzer alerts. The solution is cost-effective, uses open-source tools, and can be integrated into existing vehicle dashboards [3][5].

1. LITERATURE REVIEW AND DESIGN REQUIREMENTS

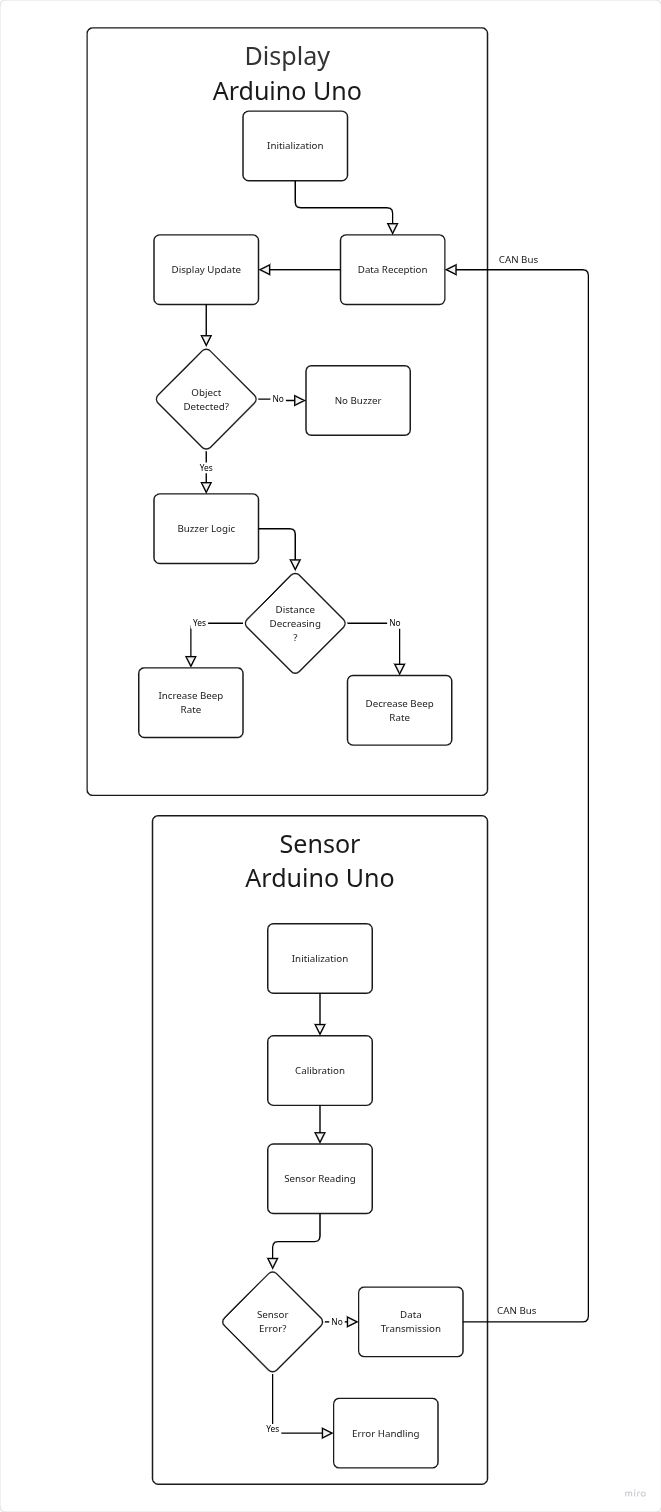
Previous works have successfully utilized HC-SR04 ultrasonic sensors for obstacle detection, with an accurate range up to 400 cm [3][5]. Kumar et al. demonstrated a basic system with real-time display, while Joshi et al. implemented ultrasonic-based parking systems with Arduino integration [7]. In the domain of vehicle communication systems, Raut and Deshmukh discussed the reliability and wiring simplicity offered by CAN in multi-node automotive applications [4]. The CAN protocol was initially developed by Bosch and has become a global standard in automotive embedded communication for its low error rate, multi-master access, and message prioritization [2].

Security limitations of CAN were highlighted in works by Lin and Sangiovanni-Vincentelli, who pointed out the lack of message authentication [1]. This concern is addressed in academic solutions such as the counter-based replay protection technique proposed by Szilagyi and Koopman [2] and TESLA-based authentication mechanisms [6]. These insights shaped the foundation of our system, focusing on modularity, fault tolerance, and expandability for low-cost vehicles.

1. SYSTEM IMPLEMENTATION

The envisioned reverse guidance system is based on a two-node architecture with two Arduino UNO boards, each connected to an MCP2515 CAN controller. The Sensor Node conducts obstacle detection through three HC-SR04 ultrasonic sensors, and the Display Node offers real-time visual and auditory feedback through an OLED screen and buzzer. Interaction between the two nodes is conducted through a CAN bus at 125 kbps.

**Figure 1:** Block Diagram of Reverse Guide System

Sensor data is grouped into five levels of proximity, stored in bytes, and sent in SPI-based CAN frames. These are mapped to decreasing thresholds of distance from an obstacle, hence facilitating dynamic warnings. The data loop takes 10 milliseconds to complete, which guarantees low latency and responsiveness.

All the hardware parts were mounted and tested in breadboards. Both ends of the CAN bus were terminated with 120 Ω termination resistors for the correct line impedance matching. The firmware development involved the Arduino IDE, using the mcp\_can.h library for CAN management and Adafruit SSD1306 for rendering OLED displays.

1. METHODOLOGY

The operational workflow of the system is illustrated in Figure 2, which outlines the behavior of both the Sensor Node and the Display Node through dedicated flowcharts.

The **Sensor Node** is responsible for detecting the proximity of objects using ultrasonic sensors. Upon system power-up, it initializes all essential modules, including I/O pins, SPI, and the CAN interface. Once initialized, the sensors undergo a calibration process to ensure consistent and accurate distance measurements. Following calibration, the sensors begin continuous distance sensing. If any irregular or invalid data is detected, an error-handling routine is triggered to maintain system reliability. When valid readings are obtained, they are converted into defined proximity levels and transmitted to the Display Node via the CAN bus. The Sensor Node operates independently, focusing solely on sensing and data transmission, while all display and alert mechanisms are handled separately.

The **Display Node** begins by initializing the OLED display, the CAN interface, and the buzzer pin configurations. It listens for incoming CAN frames from the Sensor Node, which are then decoded to extract the proximity levels. These values are used to update the OLED screen, showing visual representations of obstacle distances. Simultaneously, the system checks whether any object is detected within the defined safety threshold. If so, the buzzer is activated to alert the driver. The rate of the buzzer beeps is dynamically controlled based on the proximity of the detected object—faster beeps indicate closer obstacles.

This separation of sensing and display functions allows for modular debugging, efficient resource allocation on the Arduino boards, and reliable real-time feedback essential for vehicle guidance systems.

**Figure 2:** System Flowchart for Sensor and Display Nodes.

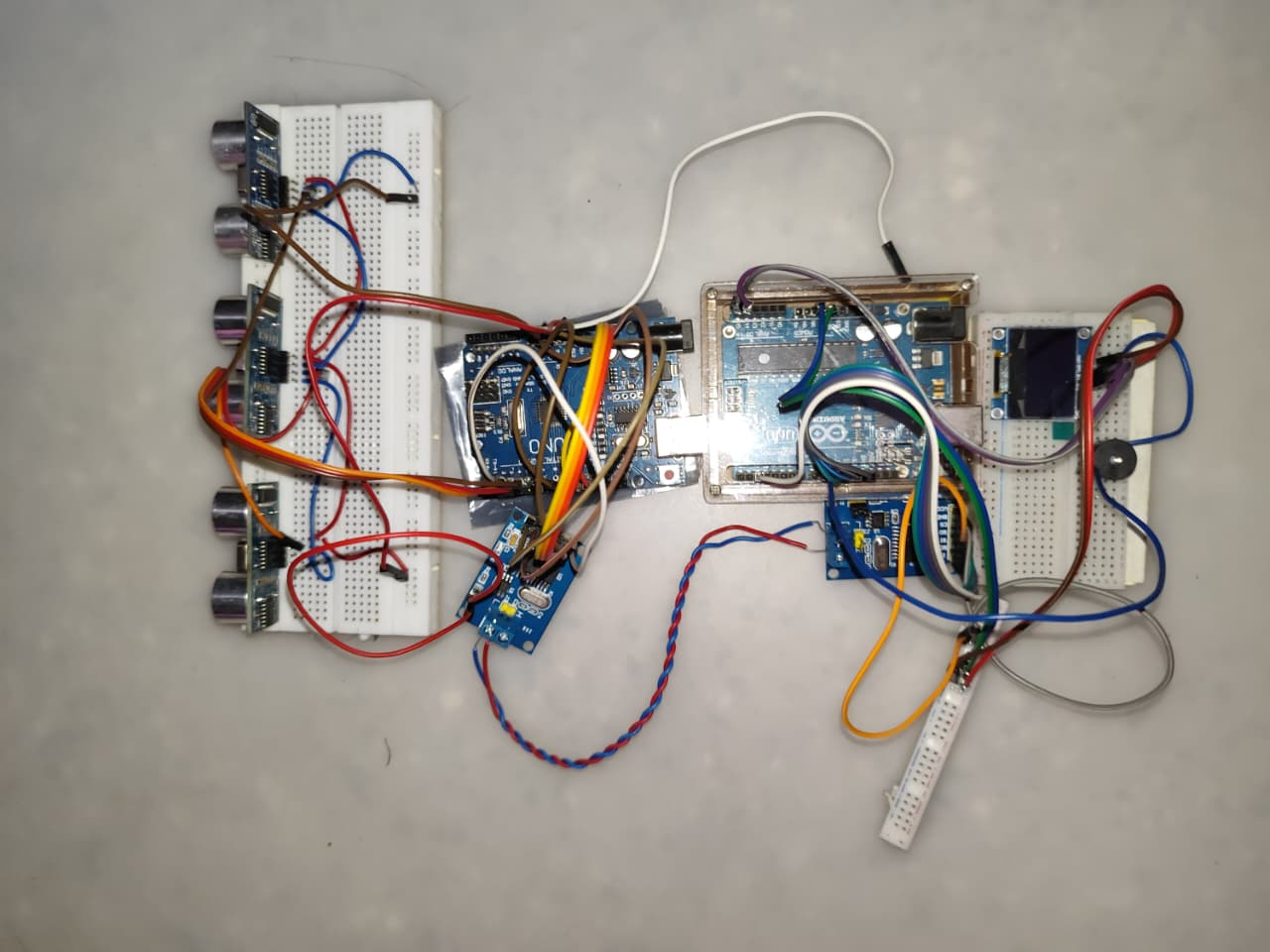
1. RESULTS

The proposed reverse guidance system was successfully implemented using two Arduino UNO boards communicating via the CAN protocol. The prototype was divided into two modules: a **Sensor Node** with four ultrasonic sensors and a **Display Node** with an OLED screen and buzzer.

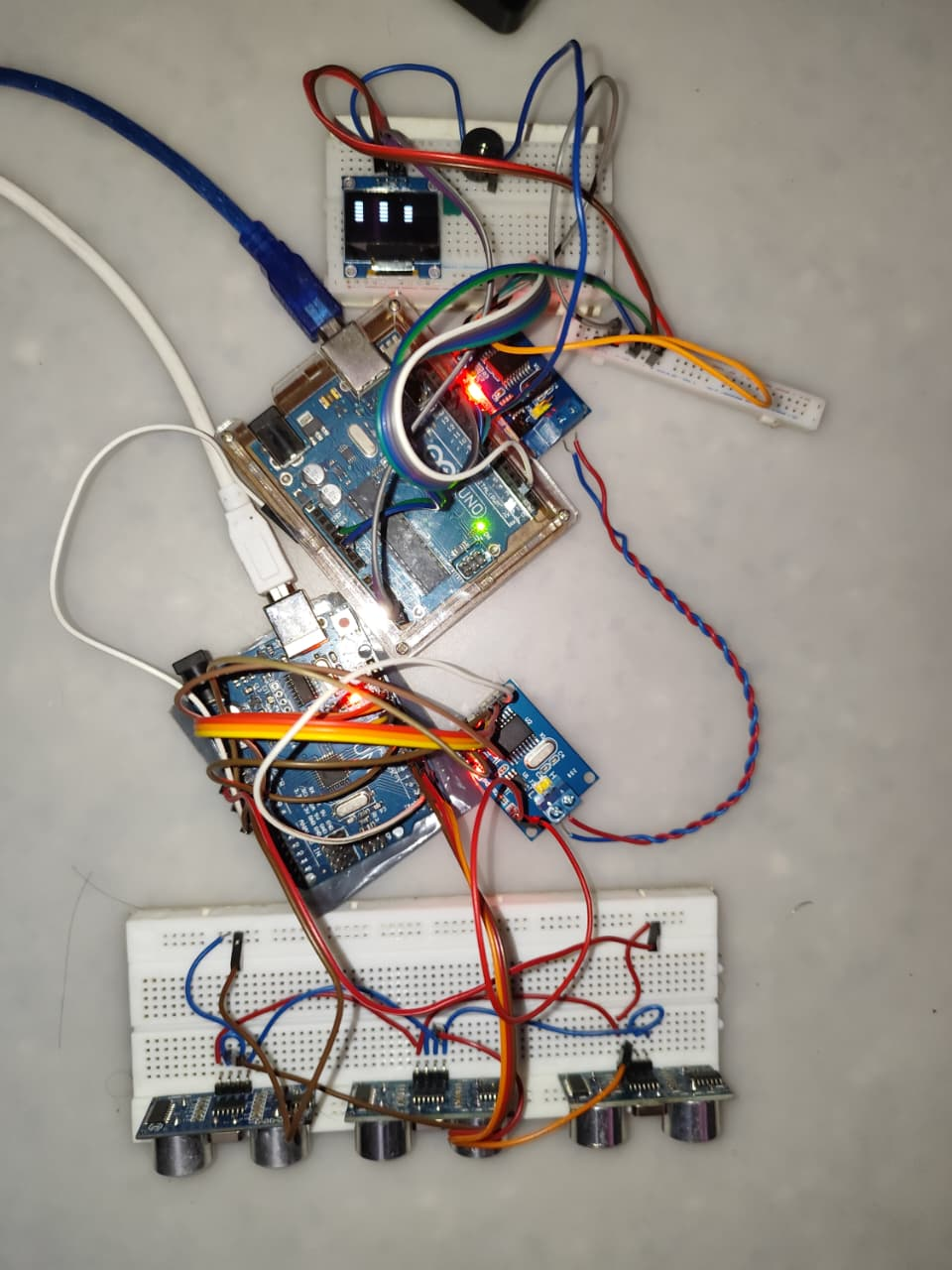
The Sensor Node accurately detected obstacle distances and transmitted encoded levels (1–5) via the MCP2515 CAN controller at 125 kbps. The Display Node received the data in real time, visualizing it as vertical bars on the OLED and providing proximity-based beeps through the buzzer.

Stable CAN communication was maintained using a twisted pair with 120 Ω termination resistors, ensuring low-latency, error-free transmission.

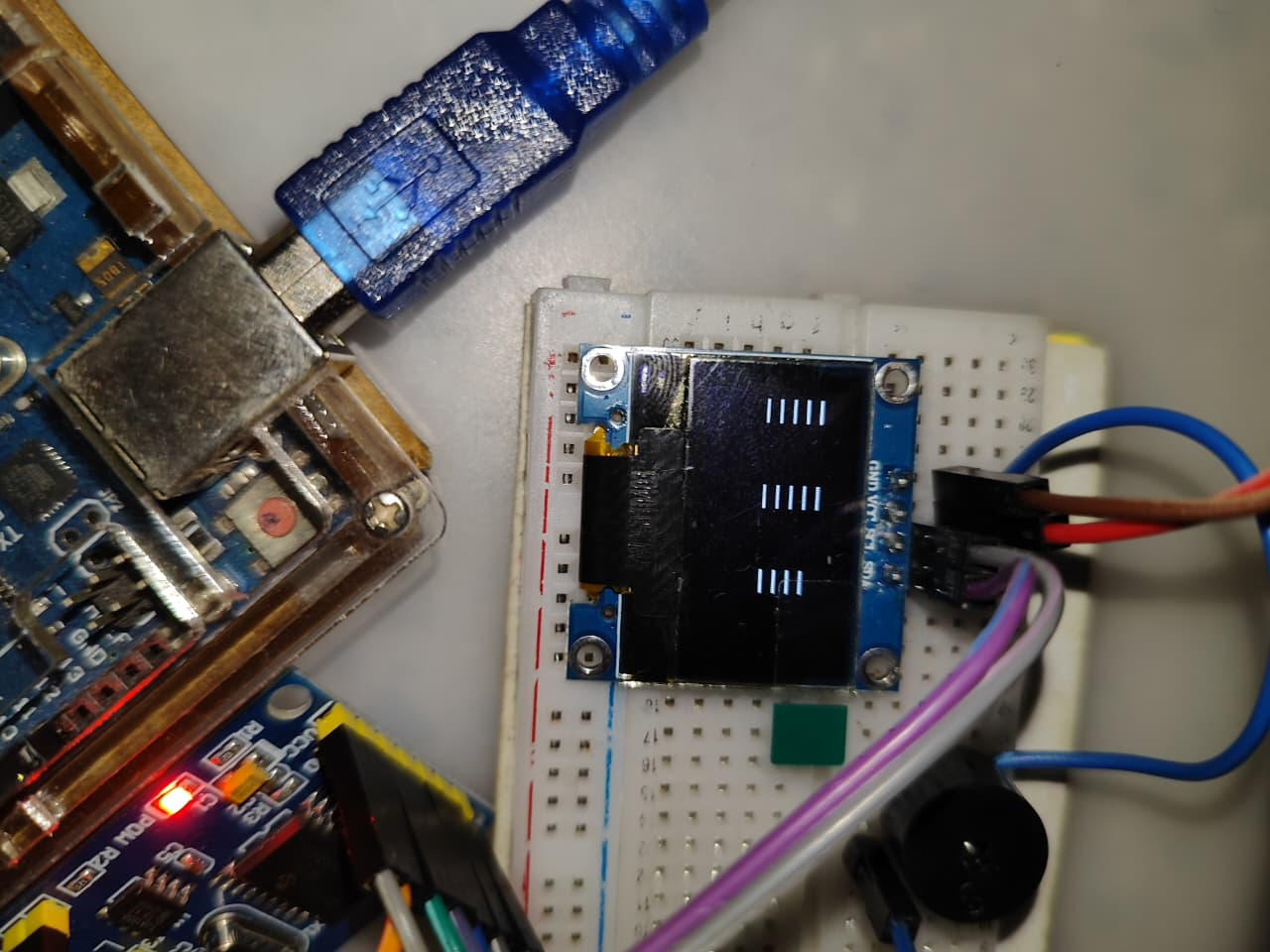
Figures 3,4 and 5 show the functional prototype, confirming real-time operation and system reliability.



**Figure 3:** Complete prototype setup on a breadboard integrating sensors and display.



**Figure 4:** Powered-on system displaying real-time distance bars on the OLED screen.



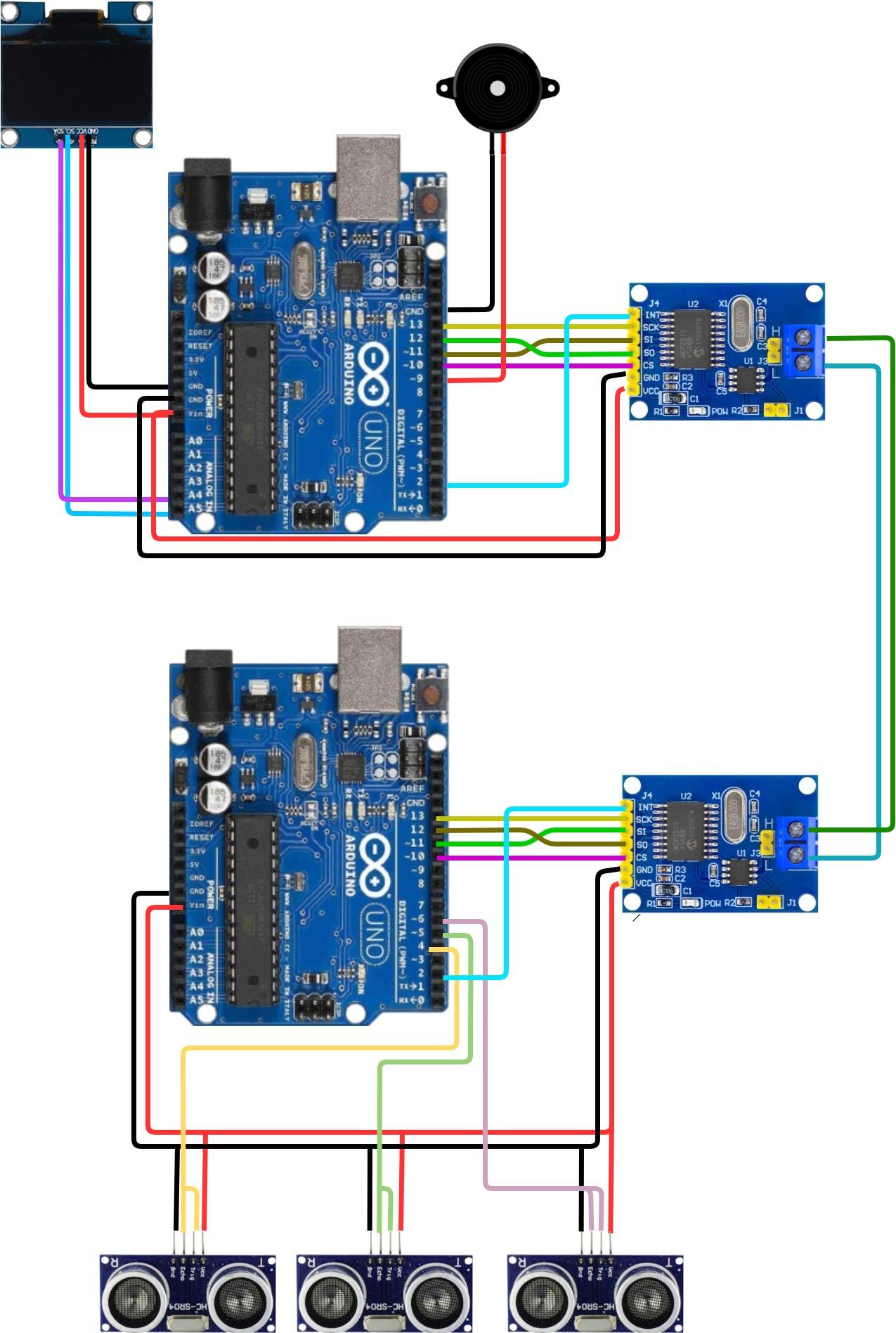
**Figure 5:** Close-up view of the OLED showing proximity indicators.

1. CONCLUSION

This research demonstrates that a reverse parking guidance system with real-time feedback can be developed using low-cost, off-the-shelf components. The combination of ultrasonic sensors and CAN communication proves to be effective and scalable. While the current system focuses on reverse proximity, future expansions may include side sensors, LCD dashboard integration, or wireless alerts. Security mechanisms such as message authentication codes (MACs) and time-stamped counters can be integrated without overloading the CAN bus [1][2][6]. The complete project offers a functional, open-source, and modular approach to enhancing safety in budget vehicles.

Appendix

**Circuit Schematic:**

  
**Figure 6:** Schematic Diagram of the Reverse Parking Assistance System

**In Figure 6** shows the schematic diagram of the Reverse Parking Assistance System, highlighting the Sensor Node and Display Node built using Arduino UNO boards and MCP2515 CAN controller modules. Both nodes communicate via a twisted pair CAN bus with 120 Ω termination resistors at each end.

The Sensor Node includes three HC-SR04 ultrasonic sensors connected to the Arduino’s digital I/O pins, with data sent through the CAN bus using the SPI-connected MCP2515 module. The Display Node features a 0.96-inch OLED display (via I2C), a buzzer for audio alerts, and another MCP2515 module for receiving sensor data.

The system operates at 125 kbps CAN bus speed and is powered through USB or a 5V supply. The schematic and code are available at:  
<https://github.com/Amazing-Stardom/arduino-reverse-car-can-system>

Acknowledgment

The authors thank the Department of Electronics and Communication Engineering, PESCE Mandya\*, for providing laboratory access and guidance. Special appreciation is extended to Mrs. Noor Ayesha for her mentorship and encouragement throughout the project.

References

1. C.-W. Lin and A. Sangiovanni-Vincentelli, “Cyber-Security for the Controller Area Network (CAN) Communication Protocol,” in *Proc. IEEE CyberSecurity Conf.*, 2012.
2. C. Szilagyi and P. Koopman, “A Flexible Approach to Embedded Network Multicast Authentication,” *Workshop on Embedded System Security*, 2008.
3. E. T. Kumar, “Ultrasonic-Based Obstacle Detection,” *Int. J. Adv. Eng. Manag.*, vol. 7, no. 3, pp. 448–455, 2023.
4. U. Raut and V. Deshmukh, “Design of Vehicle Safety Systems Using CAN Protocol,” *Int. J. Eng. Tech.*, vol. 67, no. 3, pp. 12–18, 2019.
5. N. Kumar, V. Joshi, and P. Ramesh, “Arduino-Based Parking Assistance System with Ultrasonic Sensors,” *J. Autom. Auto. Eng.*, vol. 1, no. 2, pp. 1–5, 2023.
6. A. Perrig, R. Canetti, D. Song, and D. Tygar, “Efficient Authentication and Signing of Multicast Streams,” in *IEEE Symp. Security and Privacy*, 2000.
7. A. Verma, S. Patel, and R. Ahmed, “Smart Car Parking System Based on IoT,” *ResearchGate*, 2021.

Authors

**Mrs. Noor Ayesha** – Assistant Professor, Electronics and Communication Engineering, PESCE Mandya. Email: [ayesha.asiya@gmail.com](mailto:ayesha.asiya@gmail.com)

**Ganesh Kumar** – Pre-final year B.E. student, Electronics and Communication Engineering, PESCE Mandya. Email:[ganeshkumar6120@gmail.com](mailto:ganeshkumar6120@gmail.com)  
**Gireesh A** – Pre-final year B.E. student, Electronics and Communication Engineering, PESCE Mandya. Email: [drshngrsh@gmail.com](mailto:drshngrsh@gmail.com)  
**Ajith H R** – Pre-final year B.E. student, Electronics and Communication Engineering, PESCE Mandya. Email: [ajithgowda059@gmail.com](mailto:ajithgowda059@gmail.com)  
**Darshan B S** – Pre-final year B.E. student, Electronics and Communication Engineering, PESCE Mandya. Email: [sanjeevdarshan073@gmail.com](mailto:sanjeevdarshan073@gmail.com)