Autonomous Vehicle and Smart Traffic Signal System

Using Dual ESP32 for Intelligent Vehicle-Infrastructure Communication

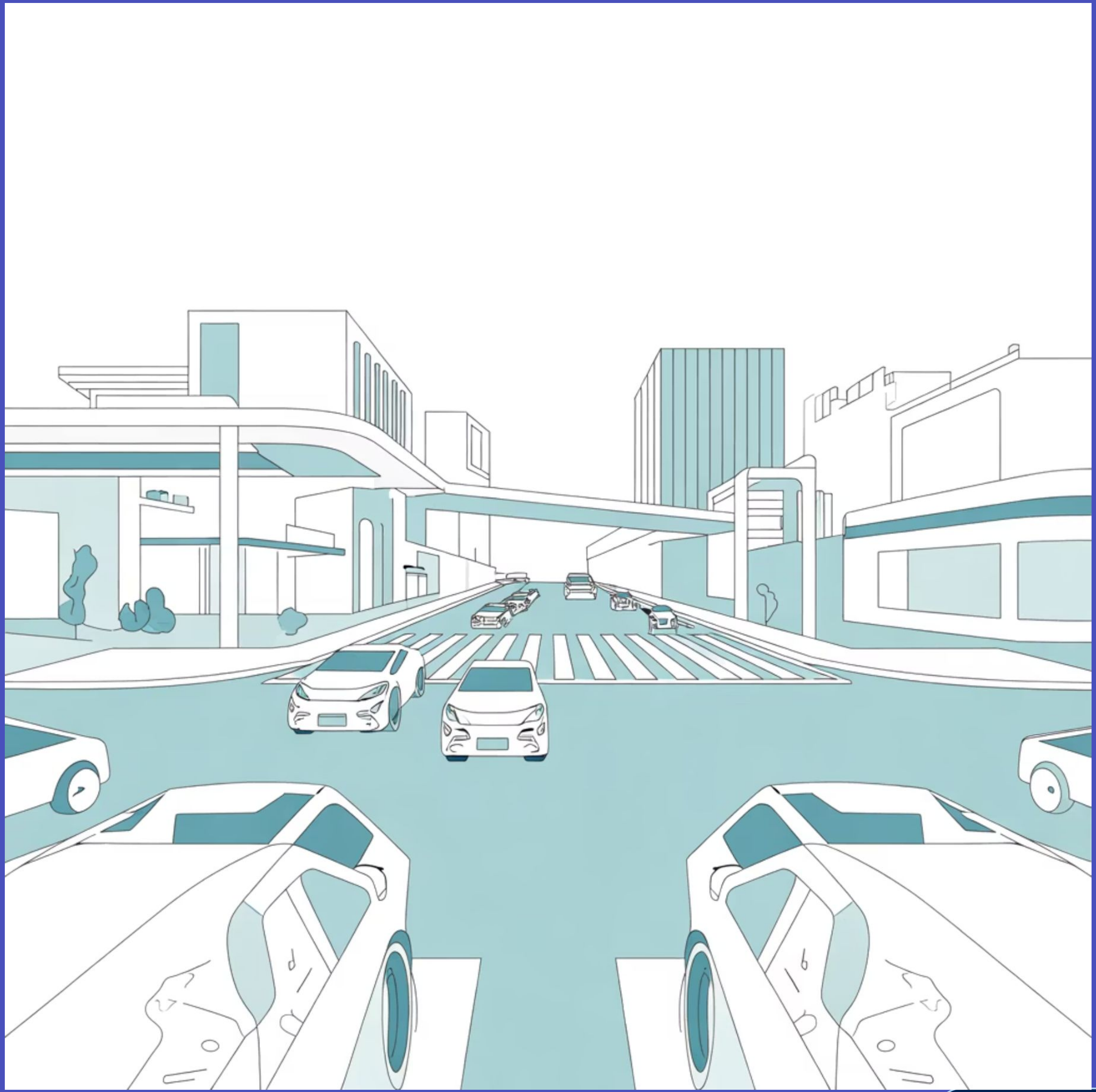
Automotive Embedded System
19RO2003

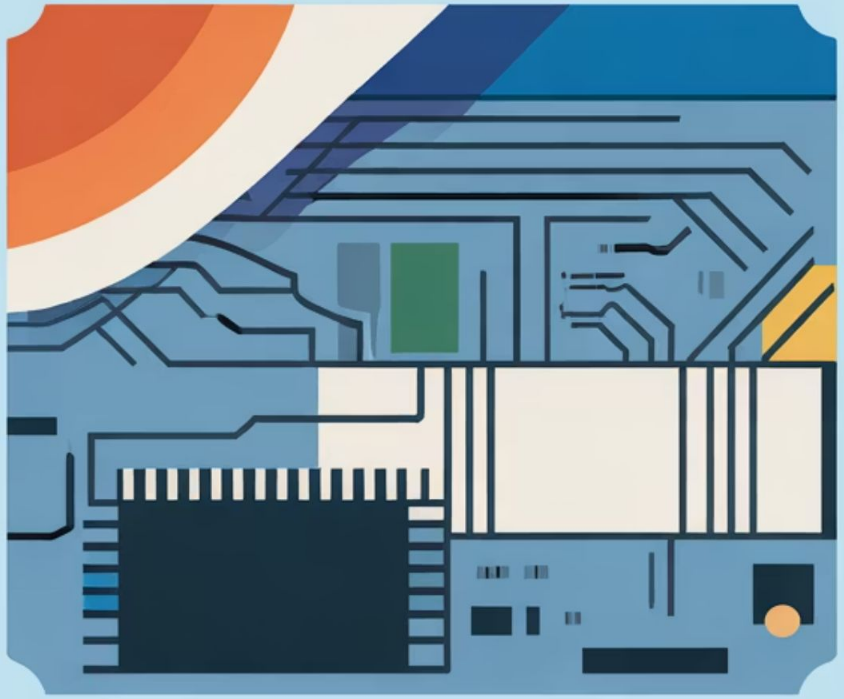
Aashiq Ahmed A URK22RA2022
Sukesan EG URK22RA2005
Krishnasamy P URK22RA1010

The Future of Smart Transportation

As urban populations grow, traffic congestion and road safety become critical challenges. Intelligent traffic management systems integrated with autonomous vehicles offer a transformative solution for safer, more efficient roads.

This project demonstrates real-time wireless communication between smart infrastructure and autonomous vehicles, showcasing how connected systems can coordinate seamlessly to reduce accidents and optimize traffic flow in modern cities.





Project Concept: Dual ESP32 Communication

Our system employs two ESP32 microcontrollers working in tandem to create an intelligent vehicle-infrastructure ecosystem. The architecture demonstrates how Internet of Things technology enables smart cities.

Traffic Signal Controller

First ESP32 manages LED signals (red, yellow, green) and broadcasts real-time status wirelessly to nearby vehicles using Wi-Fi or Bluetooth protocols.

Vehicle Control Unit

Second ESP32 receives signal data, processes traffic light status, and commands motor drivers to stop, slow down, or proceed accordingly.

Working Principle

Signal Transmission

Traffic signal ESP32 continuously broadcasts current light color status via wireless communication protocol to all vehicles in range.

Data Processing

Vehicle ESP32 receives and processes the signal data in real-time, determining appropriate vehicle response based on current traffic light state.

Motor Command

L298N motor driver receives control signals from ESP32, adjusting DC motor speed and direction to stop, slow, or move the vehicle.

Signal States

- **Red Light:** Vehicle stops completely, motors disabled
- **Yellow Light:** Vehicle decelerates gradually, preparing to stop
- **Green Light:** Vehicle proceeds at normal speed

Hardware Components

Microcontrollers

- 2× ESP32 Development Boards
- Dual-core processors
- Built-in Wi-Fi/Bluetooth
- Multiple GPIO pins

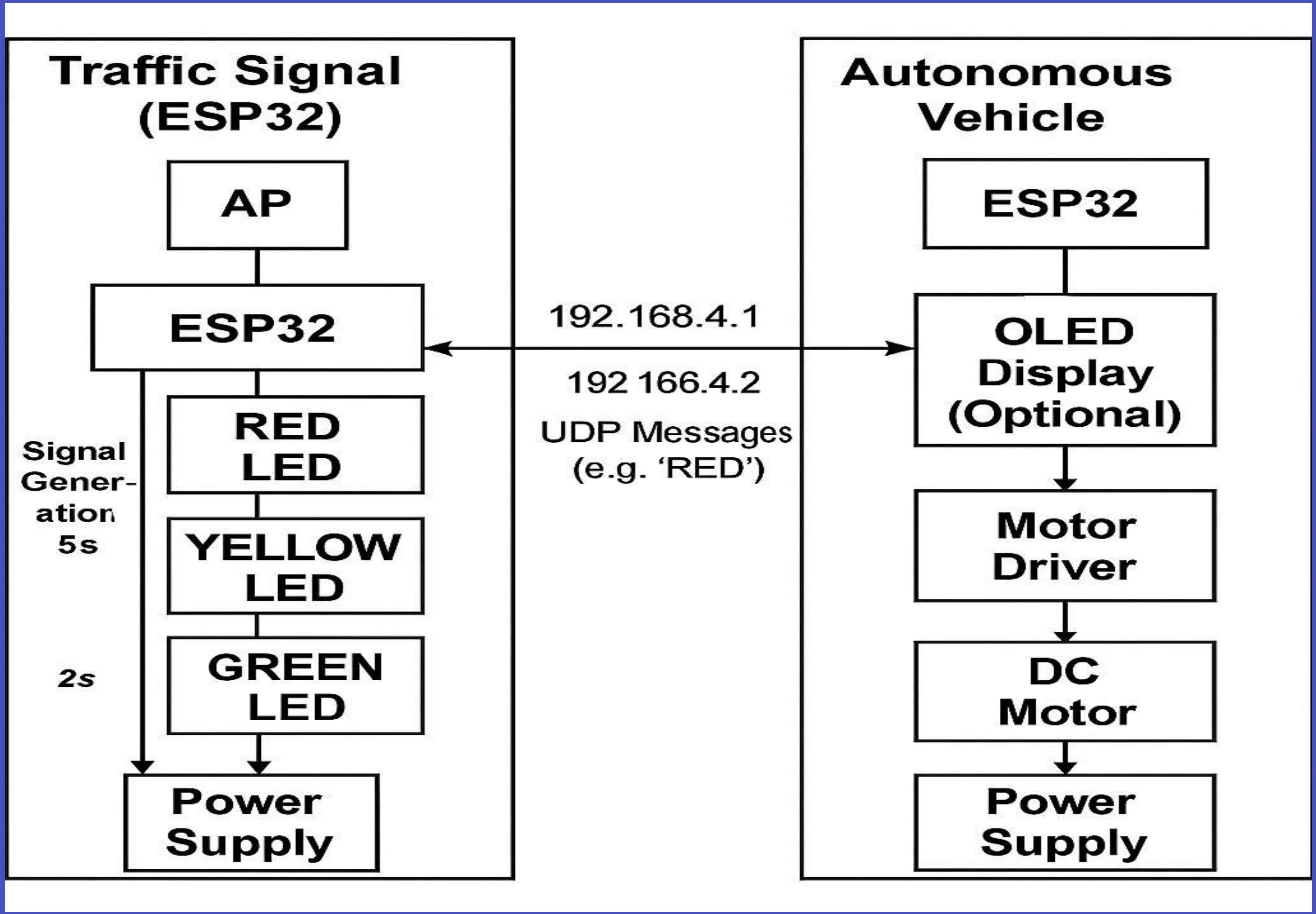
Power & Control

- L298N Motor Driver Module
- 5V/12V Power Supply
- Breadboard for prototyping
- Jumper wires and connectors

Actuation & Display

- DC Gear Motors (4×)
- RGB LEDs for signals
- Robot chassis platform
- Wheels and motor mounts

BLOCK DIAGRAM



System Architecture

01

Signal Controller ESP32

Manages traffic light LEDs and wireless transmission module for broadcasting signal status

02

Communication Link

Wi-Fi or Bluetooth protocol enables bidirectional data exchange between infrastructure and vehicle

03

Vehicle ESP32

Receives traffic data and executes control algorithms for autonomous navigation decisions

04

L298N Motor Driver

Interfaces ESP32 control signals with DC motors, providing necessary current amplification

05

Vehicle Chassis

Physical platform with DC motors that execute motion commands from control system

Software Implementation

1

Development Environment

Arduino IDE configured with ESP32 board support package, providing libraries for Wi-Fi, Bluetooth, and GPIO control with C++ programming capabilities.

2

Communication Protocol

Wi-Fi or Bluetooth Low Energy protocols enable low-latency wireless data transmission between traffic signal controller and vehicle control unit.

3

Real-Time Control Logic

Motor control algorithms process signal status and execute appropriate vehicle responses with millisecond-level precision for smooth autonomous operation.

```
// Vehicle Control Logicif (signalColor == RED) {  stopVehicle();} else if (signalColor == YELLOW) {  slowDown();} else if (signalColor == GREEN) {  moveForward();}
```

Algorithm Flow

1. System Initialization

ESP32 modules boot up, establish wireless connection, and initialize GPIO pins for sensors and actuators.

2. Signal Broadcasting

Traffic signal ESP32 reads current LED state and transmits color status via wireless communication protocol.

3. Data Reception

Vehicle ESP32 receives signal data packet and validates information integrity through error checking.

4. Decision Processing

Control algorithm evaluates signal status and determines appropriate motor commands based on traffic rules.

5. Motion Execution

Motor driver executes commands, controlling vehicle speed and direction until next signal update arrives.

Advantages & Applications

Key Benefits

Smart Coordination

Seamless synchronization between traffic infrastructure and autonomous vehicles reduces intersection conflicts and improves traffic flow efficiency significantly.

Low-Cost Prototype

Affordable components make this accessible for educational demonstrations and research, accelerating innovation in intelligent transportation systems.

Scalable Design

Modular architecture allows easy expansion to multiple vehicles and intersections, demonstrating city-wide deployment potential for smart traffic networks.

Future Scope & Conclusion

GPS Integration

Incorporate real-time navigation and route optimization based on traffic signal patterns and congestion data throughout the network.

Obstacle Detection

Add ultrasonic and camera sensors for pedestrian detection, collision avoidance, and enhanced autonomous navigation capabilities.

Cloud Management

Deploy centralized traffic control system using cloud computing for city-wide signal coordination and predictive traffic analytics.

Project Impact

This dual-ESP32 system successfully demonstrates how wireless communication between vehicles and infrastructure creates synchronized, intelligent road systems. By enabling autonomous vehicles to respond dynamically to traffic signals, we lay groundwork for safer, more efficient transportation networks that will define tomorrow's smart cities.