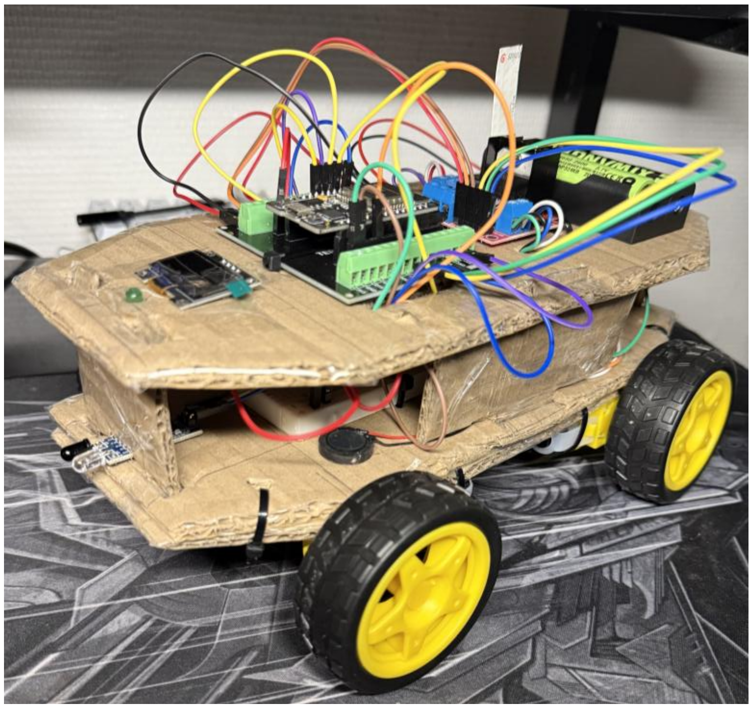
Faris Car v1.0
ESP32-Based Smart IoT Vehicle
(Portfolio Report)



Author
Muhammad Faris Bin Zulzamrin

1. Introduction

Faris Car v1.0 is an ESP32-based smart vehicle developed as a personal mechatronics portfolio project. The system integrates embedded motor control, IoT-based remote operation, user feedback mechanisms, and basic sensing into a single mobile robotic platform.

Rather than focusing solely on motion, the project emphasizes **system integration and engineering trade-offs**. Electrical power delivery, software structure, mechanical layout, and user interaction are treated as interconnected design constraints that collectively influence system behavior.

The project evolved iteratively, with features added in response to observed limitations and functional requirements. This development approach reflects real-world engineering practice, where systems mature through incremental refinement rather than fixed initial design.

2. Project Objectives

The objectives of this project are:

- Design and build a functional ESP32-controlled smart vehicle
- Implement reliable motor speed and direction control using PWM and H-bridge logic
- Develop a WiFi-based remote-control interface using the Blynk platform
- Integrate visual and audio feedback (LEDs, OLED display, speaker)
- Implement basic obstacle avoidance behavior for operational safety
- Document design decisions, limitations, and lessons learned

An additional objective is to present the project as a **portfolio-quality demonstration of embedded systems and mechatronics skills**.

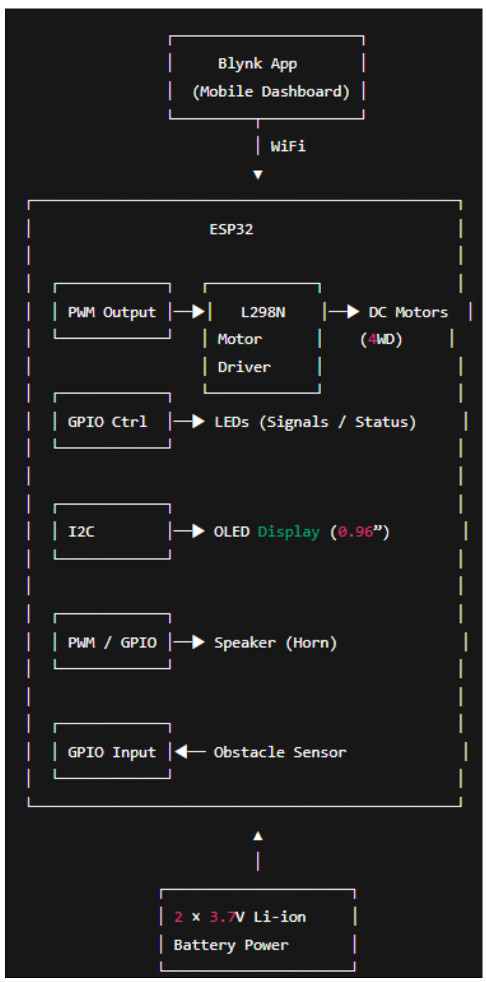
3. System Overview

Faris Car operates as a battery-powered mobile platform controlled wirelessly through a mobile application. The ESP32 microcontroller serves as the central control unit, receiving commands via WiFi and coordinating actions across multiple subsystems.

User inputs from the Blynk application are translated into:

- Motor direction and speed commands
- Lighting and signaling behavior
- OLED display updates
- Audio feedback via a speaker

The system uses an **open-loop control strategy**, meaning motor behavior is not adjusted using feedback such as speed or position. This simplifies implementation while remaining sufficient for flat, indoor operating environments.

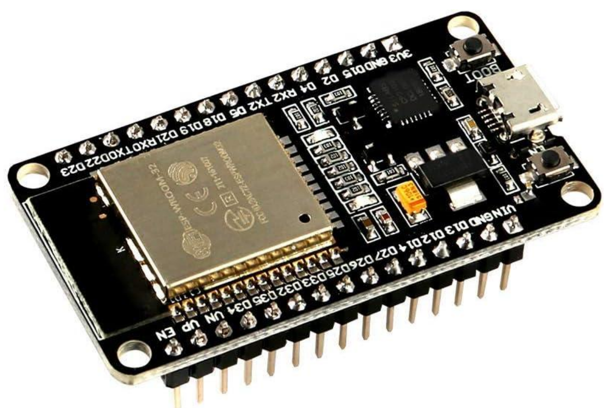


4. Hardware Design

The hardware architecture balances simplicity, availability, and educational value.

Microcontroller

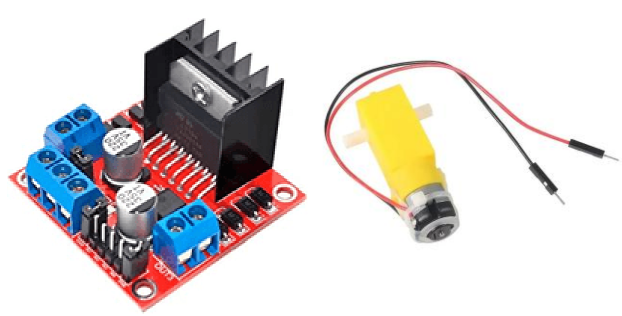
The ESP32 DevKit V1 was selected for its integrated WiFi capability, hardware PWM support, and sufficient GPIO availability. These features allow concurrent motor control, communication, and peripheral handling.



Motor Driver and Motors

Motor actuation is handled by an L298N dual H-bridge motor driver interfacing between ESP32 logic signals and four DC geared motors arranged in a four-wheel-drive configuration. This improves traction and load distribution while keeping control logic simple.

Although the L298N introduces efficiency losses due to internal voltage drops, it was selected for robustness and ease of integration at the prototyping stage.



Power System

The vehicle is powered by two 3.7 V lithium-ion batteries, providing improved energy density and current capability compared to alkaline cells.

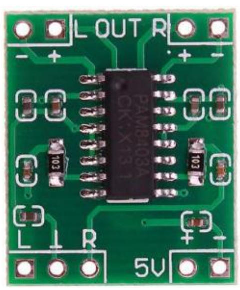
Peripherals and Sensors

- Obstacle detection module for collision prevention

- 0.96-inch OLED display for system messages

- LEDs for reverse indication, turn signals, and status

- Speaker for horn functionality



These components improve usability and system observability.

5. Software and IoT Design

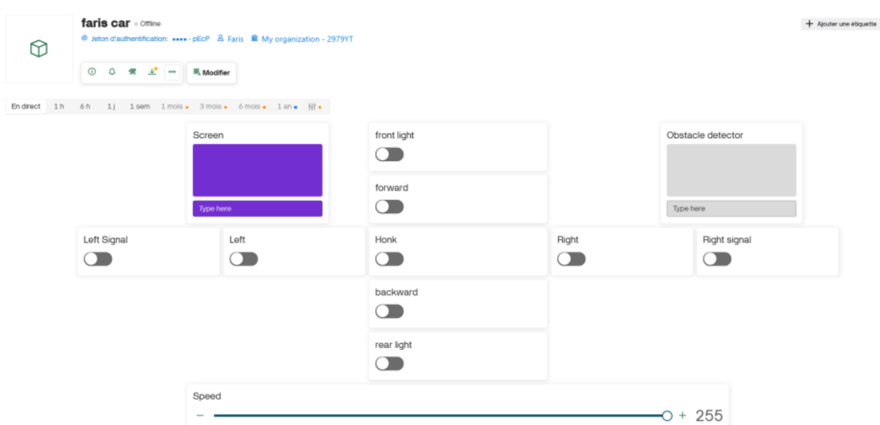
The firmware is developed using the Arduino framework for ESP32, written in C/C++. The software is structured to separate initialization, motor control, user input handling, and periodic background tasks.

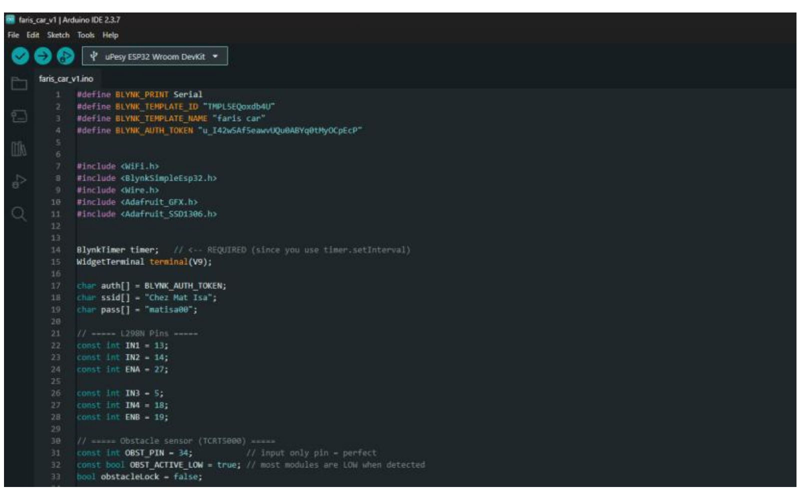
Motor speed control is achieved using PWM signals generated by the ESP32's hardware timers, while direction control is handled using digital GPIO outputs connected to the motor driver.

The Blynk platform acts as the primary human–machine interface. Through the mobile application, the user can:

- Control vehicle direction and speed
- Trigger lighting and horn functions
- Send custom text to be displayed on the OLED

This approach allows rapid modification of the user interface without requiring changes to the embedded firmware, which is a key advantage of IoT-based control systems.

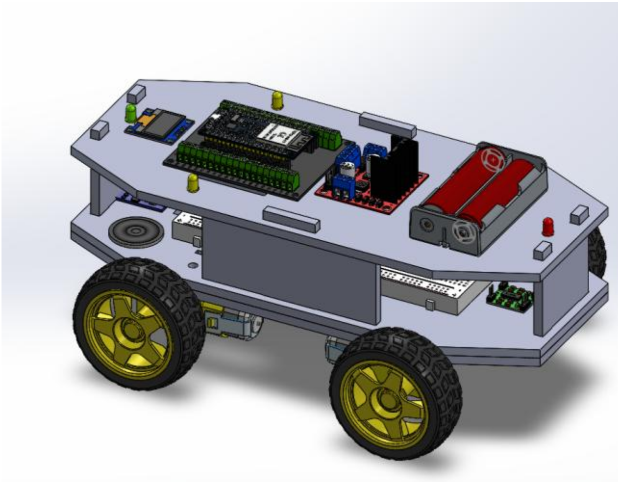


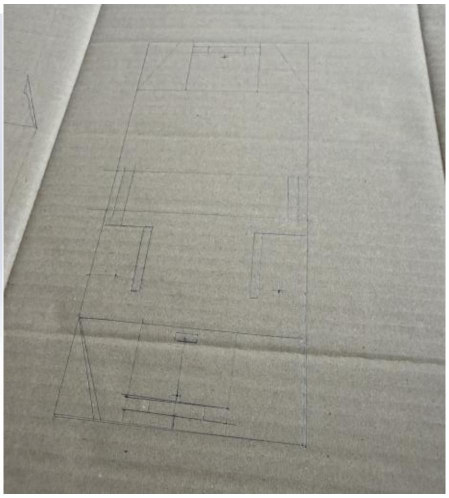


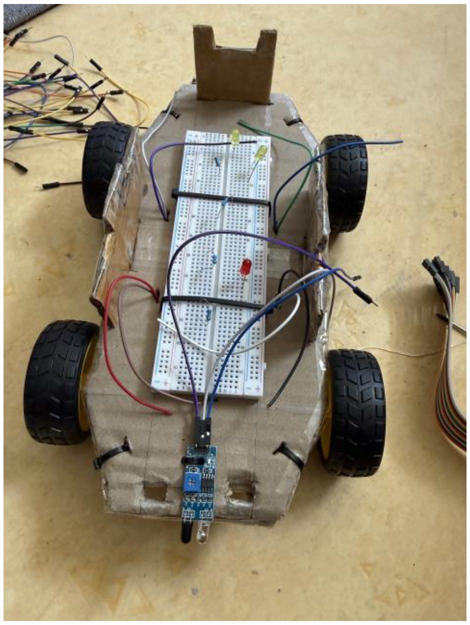
6. Mechanical Design

The chassis is constructed from cardboard materials to prioritize rapid prototyping, ease of modification, and low cost. While not mechanically optimized, attention was given to component placement, wiring accessibility, weight distribution, and sensor positioning.

SolidWorks was used for basic component placement and spatial planning prior to assembly, allowing layout constraints and clearance issues to be identified despite the non-permanent construction.







7. Testing and Validation

Testing focused on functional validation rather than formal performance benchmarking. Tests included:

- Motor direction and speed control verification
- Wireless responsiveness of the Blynk interface
- Operation of LEDs, OLED display, and speaker
- Obstacle detection and motor stop behavior

The system operated reliably under normal indoor conditions. Observed limitations were primarily related to power delivery and mechanical rigidity.

During testing, consistent vehicle behavior was observed at fixed PWM settings under identical indoor conditions. Variations in speed and responsiveness were primarily influenced by battery state and mechanical alignment rather than control logic. These observations confirm stable open-loop behavior within the system's intended operating range.

8. Results and Discussion

The completed system demonstrates successful integration of embedded control, wireless communication, sensing, and user interaction within a single mobile platform.

Performance limitations align with design choices made during development. Motor driver efficiency losses and transient current demand affect speed and torque, while the prototype chassis limits mechanical stability. These constraints are acceptable within the context of a learning-oriented prototype and provide clear direction for future improvement.

9. Conclusions

Faris Car v1.0 represents a completed smart vehicle prototype suitable for inclusion in an engineering portfolio. The project demonstrates system-level thinking, iterative development, and practical decision-making across hardware, software, and mechanical domains.

Rather than optimizing a single subsystem, the project emphasizes balanced integration and informed trade-offs. The experience gained provides a strong foundation for future projects involving improved mechanical design, higher-efficiency power systems, and more advanced control strategies.