

RAHUL S

Bangalore, India | +91-9400996955 | rahul.santhoshg@gmail.com

[GitHub: github.com/ultron-re](https://github.com/ultron-re) | [LinkedIn: linkedin.com/in/1rahuls](https://www.linkedin.com/in/1rahuls)

Professional Summary

Embedded Systems Developer with hands-on experience in **C**, **FreeRTOS**, and real-time microcontroller-based systems. Skilled in interfacing with **UART**, **SPI**, **I2C**, and working with **Bluetooth (SPP)**, **Wi-Fi protocols**, and automotive-grade hardware like **ELM327**. Transitioned from embedded QA to development, with a focus on **low-level driver development**, **RTOS-based task design**, and **ECU interfacing**. Proficient in debugging, modular C design, and building scalable embedded applications.

Skills

Programming Languages: C, C++, Python, Bash

Communication Protocols: UART, CAN, Bluetooth SPP

Microcontrollers: ESP32, Raspberry Pi Pico W, STM32 (Nucleo)

Embedded Systems: FreeRTOS, GPIO, Timers, Interrupts, Task Scheduling

Tools & Platforms: Git, CMake, Jira

Projects

Car ECU Data Logger | Raspberry Pi Pico W, FreeRTOS, ELM327

May 2025 – Present

- Developed a FreeRTOS-based system that parses real-time **OBD-II data** from a car's ECU via a **Bluetooth ELM327 dongle**.
- Built **UART driver modules**, handled **interrupt-driven communication**, and designed **modular FreeRTOS tasks** for data acquisition.
- Logged vehicle data such as **RPM**, **coolant temperature**, and displayed trends using a UART monitor.
- **Key Skills:** C, UART, Bluetooth SPP, FreeRTOS, GPIO, Interrupts, CMake, Embedded Debugging

Attendance Monitoring Using Facial Recognition

B.Tech Final Year Project

- Developed a real-time attendance system using facial recognition with the Viola-Jones detection algorithm.
- Applied preprocessing techniques like histogram normalization and skin classification.
- Captured student faces via CCTV and matched against a registered database.
- **Tools:** OpenCV, Python, Haar Cascade, NumPy

Image Steganography | C, BMP File Processing

Personal Project

- Developed a steganography system in **C** to hide and extract secret messages in **.bmp images** using the **Least Significant Bit (LSB)** method.
- Manually parsed and manipulated **BMP headers and pixel data** using bitwise operations.
- Designed custom logic to calculate image capacity, manage message embedding, and reconstruct hidden text from modified LSBs.
- Used only **standard C libraries**—no external dependencies—for file I/O and memory operations.
- **Key Skills:** C, Bitwise Operations, File I/O, BMP Format, Steganography Algorithms

Car Black Box | Embedded C, I2C, EEPROM

Personal Project

- Designed an embedded C system for continuous vehicle event logging on a custom dashboard module.
 - Implemented I2C communication with EEPROM for persistent data storage and wear-level management.
 - Handled real-time data capture (speed, sensor inputs) and retrieval via UART interface for diagnostic analysis.
 - **Key Skills:** C, I2C, EEPROM, UART, Embedded Debugging.
-

Work Experience

Automation Test Engineer (Embedded QA)

LG Soft India, Bangalore | *Jun 2023 – Feb 2025*

- Collaborated with embedded teams to validate Renault's automotive infotainment firmware.
 - Automated system-level tests over UART, Bluetooth, and USB protocols using Python and Bash.
 - Simulated real-time I/O conditions and built custom UART-based test rigs.
 - Gained deep insights into firmware architecture, driving personal embedded projects.
-

Education

B.Tech – Electrical and Electronics Engineering

Federal Institute of Science and Technology (FISAT), 2021

CGPA: 7.46

Embedded Software Engineering Certificate (ELE/Q1501 Level 5)

Emertxe (NSDC/ESSCI Certified), 2023

Certifications

- Embedded Systems – Coursera (University of Colorado)
 - Introduction to Embedded C Programming – Udemy
 - FreeRTOS Hands-On – FastBit Embedded
-

GitHub Projects

- **Car Black Box:** Embedded C system for vehicle dashboard and event logger (I2C, EEPROM).
https://github.com/ultron-re/car_black_box
- **learning_c:** Repository of C programs demonstrating pointers, data structures, and file I/O.
https://github.com/ultron-re/learning_c
- **Image-Steganography:** Secure image communication using the LSB algorithm.
<https://github.com/ultron-re/Image-Steganography>