

## Resume

Prasanth Kumar Padarthy | [prasanth.padarthy20@gmail.com](mailto:prasanth.padarthy20@gmail.com) | +91 7330813632

## Objective

Results-driven embedded software engineer with hands-on experience in developing ARM-based microcontroller firmware, RTOS-based applications, and low-level driver development. Proficient in C/C++, device drivers (UART, SPI, I2C, CAN), and embedded Linux. Excels in system-level debugging, protocol integration, and real-time software design for embedded systems.

## Experience

### Embedded Systems Trainee | Vector India, Hyderabad — Sep 2024 – Present

- Designed and implemented firmware in Embedded C/C++ for STM32 (Cortex-M4) and LPC21xx (ARM7TDMI) platforms using Keil µVision and STM32CubeIDE.
- Developed real-time applications utilizing UART, SPI, I2C, and CAN protocols for inter-device communication.
- Leveraged STM32 HAL and LL drivers for peripheral integration and abstraction, improving code modularity and portability.
- implemented interrupt-driven firmware designs with emphasis on timing and resource optimization.
- Implemented Bare - Metal and CMSIS core APIs for firmware development with precise control over timing and system resources.
- Debugged with ST-Link, UART terminals, and logic analyzers, practiced modular device driver design.

## Projects

### 1. Custom UART Protocol with 3-Way Handshaking (STM32F410RB ↔ ESP32-C3) (Ongoing)

**Skills:** Embedded C, UART, Protocol Design, Baremetal, FreeRTOS, STM32 HAL, ESP-IDF, Error Handling, Debugging Tools

- Architecting a **reliable, MCU-to-MCU serial communication protocol** with TCP-inspired 3-way handshake for connection establishment and teardown.
- Implementing **packet framing, sequence numbering, checksum-based error detection, and retransmission logic** to ensure data integrity in noisy environments.
- STM32 side: **Baremetal C firmware** with state-machine driven protocol engine, no RTOS dependency.
- ESP32 side: **FreeRTOS task-based driver** with asynchronous handling and inter-task message queues.
- Integrating **timeout handling, flow control, and dynamic packet size negotiation** for efficiency.
- Using **USB-to-TTL (FTDI) adapter** for cross-platform debugging, packet inspection, and real-time error injection testing.

### 2. Password-Authenticated Touchscreen Control System (ARM7TDMI, Embedded C)

- Designed and developed a secure device control system using ARM7TDMI and a resistive touch screen interface with password authentication.
- Utilized SPI-based EEPROM to store encrypted login credentials securely, enabling persistent and tamper-resistant authentication.
- Implemented interrupt-driven keypad interface: triggering a password change option on keypress, improving responsiveness and user interaction.
- Controlled external devices using GPIO relays, enabling or disabling outputs based on user authentication status.

### 3. Linux Minishell with Job Control (Systems Programming Project | C, Linux APIs, IPC, fork/exec/pipe)

- Built a Unix-like shell supporting commands, piping, I/O redirection, and background jobs.
- Implemented `fork()`, `execvp()`, `pipe()`, `dup2()`, and signal handlers for `SIGINT`/`SIGT`
- Supported job control commands like `jobs`, `fg`, and `bg`, handling process states and concurrency.
- Demonstrated strong understanding of Linux process management, IPC, and terminal control.

### 4. Cooperative RTOS-Style Scheduler in Pure C (No OS, No Libraries)

- Designed a cooperative multitasking system in pure C simulating an embedded RTOS scheduler with task states (`READY`, `BLOCKED`, `DELAYED`) and round-robin execution.
- Implemented custom `task_yield()`, `delay_ms()` and task registration APIs using function pointers and software timers.
- Built a simple kernel-like system for bare-metal-style task switching with minimal dependencies, targeting embedded and system-level development.
- Demonstrated scheduler behavior with simulated tick and user tasks to show time-sliced execution and cooperative yield logic.

### 5. RFID & Biometric Authentication for Exam Security (Arduino, Embedded C)

- Engineered a secure exam hall entry system on Arduino Uno by integrating an MFRC522 RFID reader and a fingerprint sensor (R305), enabling dual-layer identity verification.
- Programmed Arduino to handle real-time peripheral communication using UART (for fingerprint), SPI (for RFID), and I<sup>2</sup>C (for EEPROM), ensuring reliable multi-device interfacing and conflict-free operation.
- Implemented EEPROM-based logging of student ID, timestamp, and authentication status for audit trail; displayed authentication results on an LCD via I<sup>2</sup>C for exam staff confirmation.

## Technical Skills

- **Programming Languages:** C, C++, Embedded C.
- **Communication Protocols:** UART, SPI, I<sup>2</sup>C, GPIO, ADC.
- **Microcontrollers / Boards:** STM32, NXP LPC21xx.
- **Debugging & Tools:** ST-Link, UART terminal, logic analyzer, GDB
- **Version Control:** Git.

## Certifications

- Advanced Embedded Systems Training – Vector India
- CISCO CCNA – Networking & Fundamentals

## Education

B.Tech in Electronics & Communication Engineering  
Sir C.R. Reddy College of Engineering, AP | 2023 | CGPA: 6.55

Diploma in ECE  
Dr. B.R. Ambedkar G.M.R Polytechnic, AP | 2020 | 74.09%

SSC (10th Grade)  
Z.P.H.S Penumuli, AP | 2017 | CGPA: 8.8