

Bhakti Rajeshkumar Ramani

[Portfolio](#) | BhaktiRajeshkumar.Ramani@colorado.edu | [LinkedIn](#) | +1 (720) 641-1557 | Boulder, USA

EDUCATION

University of Colorado Boulder, Boulder, USA
Master of Science, Embedded Systems Engineering & IoT
Honors: Merit-based Scholarship

August 2024 - May 2026
GPA: 3.70/4.00

Courses: Embedded System Design, Principles of Embedded Software, Advanced Embedded Linux, Low Power Embedded Design, Concurrent Programming, IoT Embedded Firmware
L.D. College of Engineering, Ahmedabad, India
Bachelor of Engineering, Electronics and Communication

October 2020 - May 2024
GPA: 8.59/10.0

WORK EXPERIENCE

University of Colorado Boulder
Teaching Assistant, IoT Firmware

December 2025 - Present
Boulder, USA

- Developed some curriculum slides covering BLE, WiFi, & Thread and troubleshooting support
- Conducted code reviews focusing on power optimization and memory management for embedded systems

University of Colorado Boulder
Teaching Assistant, Signal & Systems

January 2025 - December 2025
Boulder, USA

- Assisted students providing support with coursework related to MATLAB, Simulink, and Signal Systems concepts.

Amazon Web Services, Annapurna Labs
Firmware Engineering Intern

May 2025 - August 2025
Austin, USA

- Created a firmware validation framework for PCIe Gen5/Gen6 systems, cutting manual test time by 60% through automation in Shell and Python
- Integrated cross-platform automation workflows into the framework, improving scalability across various server architectures
- Built PRBS-Retimer error analysis tool generating CSVs and visual analytics per link/chip/MAC, reducing manual analysis
- Performed OSFP cable qualification using firmware diagnostics and signal integrity analysis to ensure compliance and reliability

Scanpoint Geomatics Ltd
Firmware Developer, Intern

December 2022 - July 2023
Ahmedabad, India

- Developed TFT display drivers for RISC-V-based UHF metal detectors, implementing SPI-based graphics rendering with <50ms touch response time
- Architected color management system for TFT displays, incorporating industry-standard color calibration techniques.

SKILLS

- Programming Languages:** C, C++, Python, Bash, ARM Assembly
- Microcontrollers:** STM32 (Cortex-M0/M4), ESP32, nRF52, ATmega, 6502, 8051
- Operating Systems:** FreeRTOS, Zephyr RTOS, Embedded Linux, Unix,
- Protocols:** UART, SPI, I2C, BLE, WiFi, USB-PD, HID, PCIe
- Tools:** GDB, J-Link, STM32CubeIDE, Keil, Git, Make/CMake, KiCad, Altium
- Hardware:** Oscilloscope, Logic Analyzer, Spectrum Analyzer
- Concepts:** Device Drivers, DMA, Interrupts, Power Optimization, Bootloaders, CI/CD POSIX, Unit testing, Linker Scripts

PROJECTS

BLE Capacitive Trackpad from Ground Up

- Designed a low-power end-to-end system with a custom PCB featuring Tx-Rx pads, EFR32BG13 (cortex-M4) for mutual capacitance sensing
- Wrote firmware drivers for capacitive touch, gestures, haptics and integrated BLE HID-over-GATT for wireless communication.
- Implemented features such as capacitive touch sensing, multi-tap & hold, scroll, macro switches, haptic feedback and USB-C charging.

8-bit CPU | [Link](#)

- Built a fully functional 8-bit CPU from the ground up on perf board using soldering, utilizing **TTL logic gates** and an **EEPROM**.
- Designed 16-byte RAM, x86-inspired ALU, and microprogrammed control unit for instruction decoding/execution, enabling computations.
- Integrated a PC, an Instruction Decoder, 8-bit GPR, gaining 1kHz clock speed with manual clocking capability for step-by-step debugging.

File System & NAND Flash Drivers with ARM Dev Board

- Developed drivers for raw NAND Flash memory, with Bad Block Management and Wear Leveling algorithms for memory reliability.
- Optimized storage efficiency by combining **FLogFS** with **custom drivers**, ensuring efficient block allocation and garbage collection algorithms.
- Achieved improved data integrity and device longevity through efficient Flash management and file system optimization.

Drivers for STM32

- Programmed bare-metal peripheral drivers (**DMA, DAC, ADC, I2C, PWM, SPI, UART, watchdog, Timer**) for STM32 Cortex-M0 & M4
- Implemented state machine, interrupt-driven DMA handling for efficient data processing, and optimized codebase through assembly

Bare Metal RTOS in STM32F411 using C & ASM

- Wrote a bare-metal RTOS for STM32F4 (Cortex-M4) with ARM Thumb mode Context Switching and SysTick-based Task Scheduling.
- Modeled (**Round-Robin, FCFS, Periodic**) schedulers and synchronization primitives such as **semaphores and mutexes**.

Retro Computer Recreation (6502-based Computer) | [Link](#)

- Constructed 6502-based embedded system with external RAM/ROM, using reset vector and memory access control with logic gates.
- Utilized 256K Parallel EEPROM via chip programmer, implementing core 6502 instructions (LDA, STA, BEQ, JSR, CMP, PHA, PLA).