Keerthi **Radhakrishnan**

5503 Research Dr Apt 3202, San Antonio TX 78240

(+1) 845-978-3307 | krad246@gmail.com | krad246.github.io | krad246 | keerthi-radhakrishnan

Education

University of Virginia — *Charlottesville, VA*

B.S. in Computer Engineering — *August 2016 - May 2020*

- **GPA:** 3.752
- **Relevant Coursework:** Hardware-Software Security, Advanced Embedded Systems, Digital Signal Processing, Digital Image Processing, Computer Architecture & Design, Embedded System Design, Operating Systems
- Got an Outstanding Teaching Assistant award for 2 years of service with the department and providing extra crucial help during COVID-19 crisis.
- Inducted into Eta Kappa Nu for exceptional department performance.

Skills

Languages	C · Assembly (MSP430, ARM, AVR, SPARC) · Python
OSs	RTEMS · seL4 . Linux
Development Tools	Docker · Cross-GCC . Scan-build (static analyzer) . Ceedling
Debugging	GDB . OpenOCD · Oscilloscope · Multimeter
Miscellaneous	Git · Bash · Tcl . Gitlab CI

Work Experience

Embedded Software Engineer — *June 2020 - Present*

Southwest Research Institute — *San Antonio, TX*

- One of the sole software developers on Polarimeter to Unify the Corona and Heliosphere (PUNCH) satellite project. Designed several critical software modules based on RTEMS, including Error Detection And Correction (EDAC) on the flight processor and highly generic table-driven, scalable SWIFT protocol stack to interface with propulsion and transceiver modules on system. Have also participated in several code and design reviews.
- Developed Gitlab CI/CD pipeline for team software development, including static analysis using `scan-build`, unit testing using `ceedling`, and on-target integration testing.
- Led a flight software EDAC anomaly investigation on unexpected flight processor resets for Cyclone Global Navigation Satellite System (CYGNSS), wrote up and presented findings to stakeholders after identifying several key bugs.
- Led and wrote a $75,000 internal research proposal to develop formally proven seL4 microkernel as a flight software hypervisor implementing time and space partitioning for mixed criticality systems - comparable to ARINC 653 operating systems.
- Developing another $75,000 internal research proposal to develop a hardware accelerator for seL4 permissions model that enforces seL4 partitioning at the hardware level.
- Have contributed to department marketing thrusts to increase revenue and department capabilities.
- Have conducted several interviews for new hires and am currently trying to take on mentorship roles for incoming students.

NASA Big Idea Challenge Engineer — *March 2020 - Present*

University of Virginia — *Charlottesville, VA*

- Part of a 5-person embedded R & D team developing a long-range power beaming energy harvesting system for lunar polar crater exploration.
- Reverse engineered a laser galvanometer using an AD2 Discovery oscilloscope and a multimeter for external control by a microcontroller.
- Designed control circuitry to interface single-ended 3.3V microcontroller signals with bipolar 10V galvanometer interface.
- Wrote embedded C firmware for wireless rover control over BLE using CC2650MA shield.
- Designing rover energy harvesting system for laser power supply to charge multi-cell LiPo battery pack.
- Co-authored paper which went on to win 'Best Technical Demonstration' AND 'Best Technical Poster' in the 2020 Big Idea Challenge.

Sensenet Embedded Systems Engineer — *August 2019 - May 2020*

University of Virginia — *Charlottesville, VA*

- Designed micropower ATSAML21 evaluation board with USB and battery charging interface.
- Mapped SAML21 current consumption to register configurations by writing embedded C firmware in Atmel Studio and measuring with both the Atmel Power Debugger and a multimeter.
- Reconfigured and tested ArduinoCore SAMD21 bootloader for SAML21 MCU.
- Integrated embedded sensor nodes into The Things Network and MQTT backend.

Projects

MS-TOS — *April 2019 - Present*

- Designed preemptive RTOS for MSP430 and MSP430X ISA supporting all code and data models in C and assembly.
- Wrote and tested 16-bit optimized data structures in C for ultra low 100 µs kernel overhead and deterministic response times including red-black trees and deques.
- Implemented and tested VTRR, a deterministic fair-share scheduling algorithm for preemption as outlined in academic literature. Combined VTRR with a fixed-priority scheme to interleave real-time and non real-time threads.
- Profiled kernel performance using hardware timers and a Tektronix oscilloscope.
- Implemented kernel synchronization primitives and other kernel services such as thread joining and killing, IPC, thread sleeping, dynamic memory allocation, tickless operation, and stack checking.