Sean Link

seanybaggins1@gmail.com | Linkedin | Github | Website

Education

- B.S. Computer Engineering at Embry-Riddle   August 2016 – May 2020

Work Experience

Sr. Software Engineer at Antech (formerly Lightdeck Diagnostics)   May 2023 – Present

- Developing and maintaining CI/CD infrastructure using CMake, Nix, Terraform, and AWS for embedded Linux deployments
- Implemented A/B partition update system using Mender with custom U-Boot modifications to enable atomic firmware updates, automatic rollback on boot failure, and robust recovery mechanisms for field-deployed medical devices
- Owned Yocto build system as primary maintainer, managing bootloader (U-Boot), kernel, and rootfs layers for production embedded Linux distribution
- Led documentation transformation by championing adoption of the Diataxis framework across engineering teams, retrofitting existing documentation to improve discoverability and usability for developers
- Architected reproducible development environments using Nix and direnv, enabling consistent builds across developer machines and CI/CD pipelines
- Established Nix-based packaging pipeline to generate multiple distribution formats (RPM, arx bundles, tar) from single source, reducing packaging maintenance overhead
- Migrated AWS infrastructure from manual provisioning to Infrastructure as Code using Terraform, enabling version-controlled, reproducible deployments and reducing configuration drift

Software Engineer II at NetAlly   November 2021 – May 2023

- Created a bare metal product using a STM32 Cortex-M Microprocessor and the Rust Programming Language
- Brought up new hardware to run Android
    - Created Android App to communicate with Rust APIs via the Java Native Interface (JNI)
    - Modified the Linux Kernel via Menuconfig and C source code
    - Modified Android source code to allow for remote control of devices via VNC
    - Standardized build environments using Docker and the Nix Package Manager

Software Engineer I at Syncroness   April 2020 – August 2021

- Automated workflows and deliverables using AWS EC2, Jenkins, GKE and GitHub Workflows
- Developed medical device to send encrypted telemetry using C on an ARM Cortex-M4
- Created teaching materials for the Rust Programming Language
- Developed monitoring software for spacecraft using C++ for an FPGA target

Software Engineering Intern at Syncroness   May 2019 – August 2019

Developed medical device that determined the probability of a blood sample being septic (infected)

- Used test-driven development to program peripherals using Python
- Verified and configured proper serial communications between subsystems
- Automated unit test coverage reports across multiple programming languages
- Modified automated unit tests in Travis CI
- Created a custom Kernel using Yocto

iOS Development Intern at Jeppesen a Boeing Company   May 2018 – August 2018

- Created feature for peer to peer file sharing using Bluetooth
- Created documentation using Git Source Control and Markdown

Software Readiness Intern at Jeppesen a Boeing Company   May 2017 – August 2017

- Developed software to display geographical data in Google Maps
- Tested and troubleshooted flight optimization algorithms by modifying Lua scripts

Teaching / Mentorship

Tutor at Red Rocks Community College   June 2025 – Present

- Physics, Math, Computer Science

Volunteer Tutor at CU Boulder/I have a Dream Foundation   September 2024 – December 2024

- Assisted 4-6th graders to complete interactive physics labs

Tutor at Embry-Riddle   September 2017 – May 2020

- Data Structures and Algorithms, Microprocessors, Introduction to Computer Programming, Linear Circuit Analysis, Calculus 1 – 2, Physics 1

Relevant Qualifications

- Active Open Source Contributor
    - Packaging:
        - https://github.com/NixOS/nixpkgs/issues/274274

        - https://github.com/nix-community/nix-bundle/pull/122

    - Pull Requests: PR1, PR2, PR3, PR4, PR5, PR6, PR7
    - Review: https://github.com/stm32-rs/stm32f3xx-hal/pull/189
    - Author: Rust at Syncroness, Inverted Pendulum
- Embedded Systems Programming both Bare Metal and Embedded Linux
- Real-Time Programming using Linux RT Extensions
- Control Systems Design using State Space and Simulink

- o   Demo: https://youtu.be/w2OGMYozBlk
- Team Robotics competition
    - o   Demo: https://youtu.be/BFhPgVvaZl0

Programming Languages, Tools & Protocols

ARM, AWS (EC2, S3, IAM), BASH, Bitbucket, Bluetooth, C/C++, CAD, Calculus, Chacha Cypher, Cmake, Controls, Diataxis, direnv, Docker, Fish, Git, GitHub Workflows, Google GKE, I2C, Infrastructure as Code, IPv4, IPv6, Jenkins, Jira, Linear Algebra, Linux, Logic Analyzer, Mathematica, MATLAB, Mender, MQTT, NAT, Nix, Numpy, OpenSSL, OpenCV, Oscilloscope, Python, RPM, Rust, RS-232 (a.k.a. UART), Simulink, SPI, SSH, Statespace, TCP/IP, Terraform, TLS, U-Boot, VHDL, VNC, WPA2, and Yocto