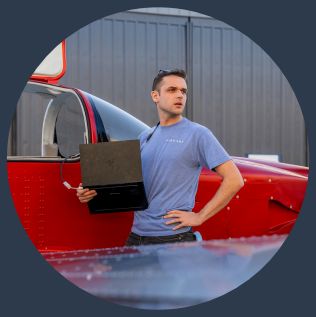

SOREN RADEMACHER

Embedded Linux SWE

sorenrade@gmail.com

+1 (919) 928-4718

Chapel Hill, NC

About

Startup-seasoned software engineer with 9+ years of experience building high-performance systems in embedded Linux, Rust, and real-time simulation. Specialized in hard tech environments, with deep experience writing safety-critical flight control software and programming large-format 3D printers. Demonstrated pedigree of versatility, with a proven ability to lead and deliver projects ranging from procedural 3D geometry tools to touchscreen GUIs and full-stack web apps.

Languages

Rust · Python · C · Lua · JavaScript · Java · C# · HTML5/CSS3 · 8086 Asm · Typst

Tools

Linux · systemd · Bash · Git · OpenGL · SQL · Blender · Rerun · Jira · Git{Hub,Lab} · GDB, Valgrind, Flamegraph, Etc.

Education

B.S. Mechanical Engineering
2013-2018 | NCSU

Minor in Computer Science
2013-2018 | NCSU

Work Experience

Airhart

2022-2025 | Lead Flight Software Engineer

- Wrote and deployed the flight control software that successfully controlled two semiautonomous manned aircraft. 100% reliability over 20+ test flights and hundreds of hours of simulated operation. `Rust` `Linux`
- A simulation environment to test the flight controller and algorithms. Multiple high-frequency, precisely timed, unsynchronized data streams sourced from a ground truth model based on telemetry from a flight simulator. `Rust` `X-Plane`
- Built tools for live debugging of our distributed system by monitoring multicast UDP traffic. `Rust`
- Logging and playback of gigabytes of live sensor data. `Rust`
- Binaries to interface with the proprietary protocols of off-the-shelf sensors in Rust: ADSB, Comm Radio Control, Rotax Engine Data, GPS, Digital Joystick. `Rust`

Carbon

2021-2022 | Senior Research Scientist
2018-2021 | Integration Engineer
2014-2018 | Integration Intern

- Developed a web-based GUI for prototype printers to improve experiment reproducibility and documentation. Backed by git, it captures and tracks all aspects of the print process: the firmware, the 3D model, tunable process parameters, and outcomes. It provides searchable print outcomes so that any experiment can be repeated. `Rust` `SQL` `HTML5/CSS3` `Lua`
- A suite of software tools to modify 3D geometry after slicing and an algorithm to procedurally add texture to arbitrary 3D models. Patent pending. `Rust` `Bevy` `WGPU`
- Developed a touchscreen GUI for our prototype printers (later replaced by the first bullet). `C` `C++` `wxWidgets` `OpenGL` .
- Contributed to algorithm design for automatic support generation software based on a novel SDF technology. `C++`
- A print preparation GUI. `Python` `wxWidgets` `OpenGL`
- A microcontroller-based 'Light Engine' Calibration tool. `Python` `C`
- Algorithms to procedurally modify geometry to improve print performance. `Python`