

Ron Rounsifer

Senior Embedded & Systems Software Engineer · C/C++ · Embedded Linux (Yocto) · Drivers & Bring-up · Safety-Critical Systems

Minneapolis, MN (fully remote) · ronronsifer@gmail.com · 248.924.7411 · [linkedin.com/in/ronaldronsifer](https://www.linkedin.com/in/ronaldronsifer) · ronsifer.github.io

SUMMARY

Seven years building safety- and mission-critical software: a next-generation medical RF generator at Medtronic, DoD/DARPA autonomy and networking programs at Raytheon BBN, and a GPS-denied drone swarm built independently. Works from silicon to UI — embedded Linux and self-healing boot, RTOS/bare-metal firmware, drivers and bring-up, mesh networking, ROS 2 autonomy, and full-stack tooling — under formal V&V and regulated design controls.

SKILLS

- **Languages:** C, C++, Rust, Python, TypeScript, Bash
- **Embedded / Systems:** Embedded Linux (Yocto, PetaLinux), A/B dual-boot and recovery, RTOS and bare-metal, ESP-IDF, SPI/I2C/UART drivers, HAL design, board bring-up, hardware-in-the-loop debugging
- **Networking:** Custom mesh routing (ESP-NOW), self-healing topologies, TCP/UDP, QUIC/Noise (design), p2p/DHT
- **Robotics:** ROS 2 Jazzy, PX4, mavros, OpenVINS (VIO), Gazebo Harmonic, PX4 SITL, Crazyflie
- **Hardware:** KiCad schematic/PCB layout, JLCPCB fabrication, power/LDO design
- **Process:** IEC 62304, FDA design controls, requirements traceability, formal V&V, DoD/DARPA program delivery

EXPERIENCE

Medtronic — Coronary & Renal Denervation R&D · Aug 2024 – Present

Senior Software Engineer · Software Lead, OS & Manufacturing · Minneapolis, MN

- Software lead for the operating-system and manufacturing/production software workstreams on Medtronic's next-generation RF generator platform for the Symplicity renal denervation system: architecture, cross-functional coordination with electrical and manufacturing engineering, and verification planning.
- Built the embedded Linux platform (Yocto/PetaLinux) with A/B dual-boot for both OS and application partitions, enabling self-healing recovery from a failed update or software fault without service intervention.
- Designed and implemented the temperature-sensing subsystem firmware: sensor acquisition through fault detection and safety limits.
- Implemented SPI peripheral drivers and hardware abstraction layer, including bring-up and verification on target hardware.
- Author requirements, design documentation, and verification protocols with full traceability (IEC 62304, FDA design controls).

Raytheon BBN — Advanced Concepts & Technology Group · Oct 2019 – Apr 2024

Engineer II, Research (Sep 2022 – Apr 2024) · Associate Scientist (Oct 2019 – Sep 2022) · Remote

- AORTA/RIPL: designed a generic C++ message-filter interface letting engineers retain, drop, or propagate control and data messages across the network; unit-tested with CTest and run on a Kubernetes cluster for distributed experiments.
- DARPA COHO: built the container architecture for an RF engine and cyclostationary signal processor (Python, Docker, Redis, ZMQ) and the experiment UI (TypeScript/React) on AWS with Jenkins CI/CD.
- DARPA STOIC: built real-time and post-processing visualization for positioning, navigation, and timing (PNT) algorithms.
- DARPA ASTARTE: implemented the interface bridging new sensor hardware to subscribed nodes on the network (Java, JUnit).
- MOA / DARPA Ground Truth: extended field-unit software (AngularJS, Flask) and built simulation scenarios with automated Python reporting.

INDEPENDENT PROJECTS

- **Covey (GPS-denied SAR drone swarm)** — Crazyflie scouts plus a PX4 "shepherd" running OpenVINS and ROS 2 Jazzy on a Raspberry Pi 5, built simulation-first in Gazebo Harmonic / PX4 SITL. Wrote a custom ESP-NOW mesh routing protocol (ESP-IDF C) with sub-second self-healing on node loss; asyncio ground-control backend and TypeScript/D3 UI.
- **Custom carrier boards** — Designed and fabricated Raspberry Pi 5 + multi-ESP32 + GPS carrier boards in KiCad (JLCPCB), including power budgeting, LDO selection, and bring-up.
- **Vera (SaaS)** — Full-stack product for e-commerce resellers: Expo/React Native, Next.js, Supabase.
- **Agentic tooling** — Multi-agent Claude Code workflow with strict directory ownership; MCP servers wrapping serial/ESP tooling for hardware-in-the-loop debugging.

EDUCATION & HONORS

B.S. Computer Science, minor in Data Science (AI/ML), Grand Valley State University, 2019 · 2020 BDI Winner, Raytheon BBN (team HASHawk) · Publication: "Parallelizing FluidC, a Community Finding Algorithm," IEEE EIT 2019 (~4× faster convergence).