Natanel Roizenman

natanel.roizenman@gmail.com | +1 (647) 542-2174 | linkedin.com/in/nroize | natanel.ca | Ontario, Canada

Education

University of Waterloo – BASc in Computer Engineering Expected 2027

- Communications and Signal Processing Specialization

Skills

Languages: C, C++, Python

Tools: WinDBG, GDB, JTAG, Logic Analyzers, ETW, Git, Perforce, CMake

Technologies: embedded systems, display protocols (DisplayPort, HDMI, DP Alt Mode), low-level drivers, firmware development, power optimization, Unix/Linux kernel, Windows kernel-mode drivers

Experience

Software Engineering Intern, Apple – Cupertino, CA May 2025 – Aug 2025

- Architected a scalable firmware-level serialization/deserialization framework in C++ to support data extraction for regression testing and pre-submission validation
- Reengineered display pipeline analysis tooling in C++ firmware, improving reliability and adding support for external display pipes, new display features, and extending tools for legacy and upcoming platforms
- Integrated display pipeline data capture and analysis into automated kernel panic diagnostics in C++ and Swift, improving triage and debugging workflows for display teams

Associate Engineer (Part-Time), AMD – Toronto, ON Jan 2025 – Apr 2025

- Led enablement and optimization of new features for an upcoming semi-custom product in C and C++
- Contributed regularly to the upstream Linux kernel in C
- Part-time contract during academic studies

Software Engineering Intern, AMD – Toronto, ON Sep 2024 – Dec 2024

- Resolved critical HDR stability issues affecting approximately 7% of configurations, eliminating display corruption and visual artifacts in OLED panels
- Optimized Panel Self-Refresh (PSR) implementation across display panels, reducing display power consumption by up to 15% in typical usage scenarios
- Reduced HDCP-related customer escalations by 25% through targeted DisplayPort and DP Alt Mode fixes

Software Engineering Intern, AMD – Toronto, ON Jan 2024 – May 2024

- Optimized DisplayPort compliance testing by automating manual intervention with Python, reducing validation time by more than 50% and saving ~20 engineering hours per product launch
- Collaborated with senior developers to quickly debug and resolve critical display driver issues using WinDBG
- Resolved 2x more tickets per week than expected, enabling on-schedule launch of major laptop products

Projects

3D Printed E-Reader

- Developed C++ firmware for an e-paper display, implementing custom I2C communication protocols based on component datasheets without vendor drivers
- Implemented efficient display refresh algorithms and power management systems, optimizing battery life

Custom Casio Module (Ongoing)

- Designing a custom low-power PCB replacement for the Casio DBC611 calculator watch, including component selection, schematic capture, and layout
- Developing an efficient event-based operating system in C for the PIC24 microcontroller, implementing LCD driver interfaces and precise timing control with multi-year battery life