# Nick Armstrong

(437) 344-3107 | n2armstr@uwaterloo.ca | linkedin.com/in/nick-armstrong-eng | github.com/n-arms

## Technical Skills

**Languages**: x86 Assembly, C, C++, Rust, Java, Python, TypeScript, Go, Haskell, WGSL
**Frameworks**: STM32 HAL, Web GPU, WPILib (ROS2 like robotics framework), React, Preact, Node.js
**Developer Tools**: Oscilloscopes and Logic Analyzers, STM32CubeIDE, GDB, GCC, Make, Gradle, Git, GitHub, Altium Designer, Fusion 360, Onshape, Microsoft Office, Agile & Scrum
**Relevant Skills**: Microcontrollers, Embedded Debugging, I2C, SPI, UART, CAN, SWD, Interrupts, DMA, Systems Programming, Linux

## Projects

**Grocery usage pattern monitoring tool** | *C, STM32, GDB, Git, GitHub* Sept. 2024 – Dec. 2024
- Implemented and debugged a sensor driver, using hardware timers and GPIO interrupts to minimize overhead while communicating over a synchronous protocol, identified bugs using an oscilloscope
- Designed a custom embedded protocol using Hamming error correction and a reduced baud rate in order to reliably communicate between two microcontrollers over a distance of 2 meters
- Developed a linear regression model to accurately track and predict grocery usage

**Functional programming language compiler** | *Rust, Git, GitHub* Dec. 2022 – Present
- Develop a compiler for a functional programming language, using state-of-the-art transformations to minimize heap allocations and eliminate indirect function calls
- Analyze generated Assembly to identify performance issues and improve optimizations
- Used the Rust type system to guarantee correctness of program transformations

**Robotic path planning library** | *Java, Gradle, Bash, Git, GitHub* Jan. 2024 – Apr. 2024
- Developed dead-reckoning based odometry to adaptively correct robot trajcetory and react to disturbances
- Implemented cutting-edge optimization techniques such as resilient backpropagation to generate trajectories
- Bundled the library as a reusable package that interfaces with Gradle using a Bash install script
- Deployed to a Linux device at robotics comptetitions, enabling complex autonomous routes to score more points

## Experience

**Embedded Systems Development Intern** | *Ford Motors Research & Development Centre* Jan. 2025 – Apr. 2025
- Improved a Python library to generate embedded C++ code to support manufacturing of all Ford cars
- Identify required permissions in large code base in order to write SELinux policy files
- Debug and flash embedded devices using SSH and CAN to facilitate software development

**Drone Firmware Project Manager** | *WARG, 90+ member student design team* Sept. 2024 – Present
- Implement CAN communication between nodes using the STM32 HAL, enabling powerful inter-robot comms
- Research and develop a decentralized leader election bully algorithm to identify the controller node
- Debug drivers using oscilloscopes, logic analyzers, and GDB, fixing faulty code and identifying wiring issues
- Lead a sub team of 3 developers to develop drone firmware using project boards and GitHub pull requests

**Robotics Programming Lead** | *FIRST Robotics, 50-member team Arctos 6135* June 2020 – June 2024
- Deployed code in an SoC Linux environment using FTP, SSH, Ethernet (TCP/IP Networking)
- Managed a soft real-time 2k LoC codebase with frequently changing requirements and minimal testing time
- Established the use of issues, pull requests, and code reviews to enable a dozen programmers to contribute to a competitive robot in order to promote agile development in a strong safety culture

**Summer Camp Counsellor** | *High Park Nature Centre* June 2021 – Sept. 2024
- Streamlined internal documentation processes in collaboration with supervisors, improving team legacy
- Independently resolved conflicts, managed schedules, responded to emergencies, and adapted plans

**Computer Science Speaker & Club Executive** | *Bloor CI* Oct. 2022 – May 2024
- Practiced and demonstrated data structures, algorithms, and time complexity analysis for a LeetCode-style contest

## Education

**University of Waterloo** | *Waterloo, ON* 2024 – Present

Bachelor of Applied Science in Computer Engineering — 3.98 GPA