Khalil Burns

(343)-598-5856 | kjrburns@uwaterloo.ca | khalilburns.com | linkedin.com/in/khalil-burns | github.com/khalil-burns

Technical Skills

Languages: C++, C, Make, Assembly, VHDL, Java, Python, JavaScript, TypeScript, CSS, HTML
Developer Tools: Git, GitHub, Docker, VS Code, STM32CubeIDE, ArduinoIDE, VIM
Libraries and APIs: OpenCV, NumPy, RPi, Firebase, Tensorflow, React, Ajaxk, JQuery
Hardware: Raspberry Pi, Arduino, STM32, UART, CAN, Oscilloscope, Soldering, FPGA (Quartus Prime)

Experience

Real-Time Operating Systems Developer Jan. 2026 - April 2026
Huawei Canada, iRTOS R&D Team *Ottawa, ON*
- Implemented a scheduler benefit analysis method in the kernel using **C** to quantify main thread startup time reductions from proposed priority acceleration changes, achieving **0.5ms** average accuracy validated against reference traces
- Embedded thread wakeup chain and run queue timestamp logging into the **C** scheduler for precise thread latency analysis
- Used **OpenHarmony HDC** and **Fastboot** to perform software feature validation and debugging directly on Huawei phones
- Developed **Python** scripts to parse, repair, and structure **50+** beta trace files on a per-thread basis, measuring benefits up to **15%**

Firmware Developer May 2025 - Aug. 2025
Ford, BSP/Bootloader Team *Ottawa, ON*
- Developed and deployed firmware fixes in **C** for bootloader and NOR flash systems across multiple **Linux**-based ECUs
- Validated and debugged firmware fixes directly on hardware using **Fastboot MTD** utilities within a **Linux** environment
- Designed and implemented a reusable unit testing infrastructure in **C** with **Ceedling**, now used in the bootloader codebase
- Achieved **100%** test coverage on the block file system code, uncovering **10+** critical defects present in production code

Software Developer Sep. 2024 - Apr. 2025
University of Waterloo Biomechatronics Design Team *Waterloo, ON*
- Converted the team's FNN into a CNN using **TensorFlow**, achieving a **96%** classification accuracy on EMG signals
- Processed EMG data from 4 sensors to categorize signals into 16 distinct hand motions for prosthetic control
- Investigated advanced architectures (e.g., hybrid CNN–RNN models) to improve detection of muscle contractions

Software Developer July 2023 - Aug. 2023
Nokia, IP Networks Division *Ottawa, ON*
- Developed a third-party **Python WebSocket** server to simulate communication between Nokia's NSP and external servers to test the speed of network interactions
- Containerized the WebSocket server with **Docker** for easier deployment and scalability during testing
- Integrated **Locust** for load testing to analyze the performance of the WebSocket server

Projects

Feed-Forward Neural Network From Scratch | *Java, Spring Boot, JavaScript, HTML/CSS, WebSockets* | [GitHub]
- Created a custom feed-forward neural network in **Java**, including forward/backpropagation and weight/bias updates
- Coded the gradient descent algorithm with tunable hyperparameters (learning rate, batch size, etc.) for flexible model training
- Validated performance on the MNIST dataset, achieving **96%** accuracy without the use of any external ML libraries
- Developed a full-stack web app using a **Java Spring Boot** backend and **JavaScript** frontend to host and demo the project

3D Graphics Rendering Engine From Scratch | *Java, 3D Graphics* | [GitHub]
- Developed a **Java** rendering engine to load and display textured **.obj** files, supporting free navigation and viewing of 3D scenes
- Derived projection and rotation matrices to transform 3D vectors into 2D space, featuring custom FOV and culling and clipping
- Implemented directional lighting, texture mapping, and mipmapping to improve visual realism and rendering performance

Automatic Trumpet Playing Robot | *C, STM32, UART, Fusion 360, CAD/3D Printing* | [GitHub]
- Engineered a high-speed robotic trumpet system capable of reliably actuating **18mm** valves at frequencies up to **12 Hz**
- Developed firmware in **C** on an **STM32** MCU to control precise multi-motor timing and low-latency valve sequencing
- Implemented a **UART** interface between a host keyboard and the system to enable real-time user-controlled note sequencing
- Designed and 3D-printed a custom mechanical assembly in **Fusion** for mounting the electronics onto a real trumpet

Autonomous Tracking Car | *Raspberry Pi, Python, OpenCV, PWM* | [GitHub]
- Designed and coded a **Raspberry Pi**-based autonomous car with onboard motor driver and servo control circuitry
- Added real-time object tracking by mounting a camera on a servo, combining **OpenCV** with dynamic servo positioning
- Controlled steering and drive motors using **PWM** signals to enable responsive autonomous navigation

Education

University of Waterloo Sep. 2024 - Present
Bachelor of Applied Science in Computer Engineering – GPA: 90.18% *Waterloo, ON*
Coursework: **FPGA** & **VHDL** (Fundamentals of Digital Logic), **C++** (Data structures and algorithms)