

William Park

236-996-9116 | www.williampark.org | linkedin.com/in/parkwilliamy | github.com/parkwilliamy | w5park@uwaterloo.ca

EDUCATION

University of Waterloo

Bachelor of Applied Science in Honours Electrical Engineering
Cumulative GPA: 91.44%

Waterloo, ON

Sept 2024 – May 2029

TECHNICAL SKILLS

Languages: SystemVerilog, Verilog, VHDL, Tcl, Bash, Assembly, Python, C/C++, Rust

Developer Tools: Git, Vivado, Quartus Prime, IcarusVerilog, Verilator, Xschem, SPICE, Ngspice, Magic, STM32CubeIDE, OpenOCD, GDB, Cargo, Foxglove, ROS2, Docker, WSL, Ubuntu, Linux, NVIDIA JetPack

Libraries: cocotb, CUDA, MediaPipe, OpenCV, pyserial

Protocols: SPI, I2C, CAN, UART

WORK EXPERIENCE

UWASIC

Oct 2025 – Present

Digital Team Lead

Waterloo, ON

- Leading a team of ten students to design and fabricate the first ray-tracing ASIC on the **Skywater 130nm process** through Tiny Tapeout
- Designing architecture to efficiently implement essential operations in ray-tracing like division and square roots
- Conducted parameter sweeps for samples-per-pixel and color depth in a C++ ray tracer to establish performance constraints and design requirements
- Analyzed renders to measure arithmetic operation frequency, determining critical targets for hardware optimization

Kiwi Charge

Sept 2025 – Dec 2025

Electrical Engineering Intern

Toronto, ON

- Developed a ROS2-based teleoperation pipeline for a four-wheel differential-drive prototype, integrating joystick input and real-time motor command streaming with the **CAN** protocol
- Designed an electrical schematic for the prototype, including power distribution, motor driver interfaces, CAN communication, and sensor integration, enabling reliable bring-up and debugging
- Implemented Python scripts to configure and tune multiple motor drivers in parallel over CAN, **reducing setup time by 98%** and improving system scalability

PROJECTS

R500

June 2025 - Dec 2025

- Designed and implemented a **RISC-V** processor in **Verilog** on the Xilinx Artix-7 FPGA
- Reached **100% compliance** with the RV32I ISA based on the official RISC-V compliance test suite
- Achieved a throughput of **46.3 million instructions per second** using a **57 MHz** clock, along with a **CPI of 1.23** and a **branch prediction accuracy of 96.5%**
- Optimized performance through pipelining, forwarding, stalling, flushing, and gshare global branch prediction
- Implemented a **UART** memory interface with a custom frame format for memory inspection and program loading
- Utilized **SystemVerilog** to create unit testbenches, employing directed tests and constrained random verification
- Traced simulation waveforms in **Vivado Design Suite** to resolve incorrect pipeline responses to hazards
- Employed **Verilator** for high-performance simulation of memory-intensive Assembly programs
- Automated design workflow using Python, Bash, and Tcl scripts

Robotic Hand

Feb 2025 – Apr 2025

- Built a robotic hand that accurately mimics user hand movements in real-time with computer vision using an **STM32 microcontroller** and PCA9685 motor driver
- Transformed camera data into precise servo movements using Rust, Python, MediaPipe, and OpenCV
- Programmed STM32 and PCA9685 registers to configure and manage **I2C** and **UART** data transmission
- Utilized OpenOCD and GDB to debug Rust code, validating register values to ensure seamless data transactions