Simon Marcotte

scmarcot@ualberta.ca – github.com/simonMarcotte
Edmonton, AB

Engineering Co-op Program

EDUCATION

Computer Engineering, BSc Co-op, University of Alberta — **Class of 2026**
Cumulative Grade Point Average — 3.2/4.0
Completed Academic Terms — 7/8
Completed Co-op Work Terms — 3/5

WORK EXPERIENCE

Software Engineer Co-op — **January 2024 – August 2024**
General Fusion — Vancouver, BC

- Designed and deployed a Python based data acquisition system for 20+ fusion reactor diagnostics using Docker.
- Enabled SPI communication between Arduino and FPGA using SystemVerilog, simulated and tested in Intel Quartus Prime.
- Validated diagnostic server connections by testing data transmission from oscilloscopes over TCP/IP via fiber optics.
- Developed Python device drivers to control oscilloscopes, FPGAs, and microcontrollers, improving hardware integration.
- Created Bash and Python validation scripts to debug hardware fault patterns, reducing manual debugging time by 50%.
- Achieved a 9x speedup in a custom digital signal processing analysis algorithm when refactoring from MATLAB to Python.
- Deployed a Flask-based microservice bypassing legacy code to serve raw metadata files to an ETL pipeline.
- Introduced CI/CD with GitHub Actions to automate server setup, reducing server failures from manual configurations.
- Managed project status and documentation in Confluence and Jira, using Agile to accelerate project timelines.
- Collaborated with software, electronics, and controls engineers to gather requirements and execute project objectives.

Program Instructor — **May 2023 – August 2023**
Faculty of Engineering - University of Alberta — Edmonton, AB

- Deployed a GCP hosted Python Discord Bot to automate onboarding processes for 500+ members across 4 months.
- Automated Google Cloud setup with Terraform and integrated a secure VPC with custom subnets and firewall rules.
- Led virtual engineering camp redesign, coordinating team tasks and deadlines, advancing schedule by one month.
- Coordinated communication between staff and students, led weekly meetings to adapt to student's needs.

Embedded Software Developer — **September 2024 – Present**
Alberta Heart — Edmonton, AB

- Developing microprocessor software for an artificial heart, focusing on a fault tolerant design and real-time control.
- Leveraging the software development life cycle in an 8+ person team, in preparation of the ISMCS Heart Hackathon 2025.

PROJECT EXPERIENCE

GPU Mandelbrot Speedup | GitHub | *C, C++, CUDA, ROCm, HIP, GPU Acceleration* — **November 2024**

- Achieved a 79x speedup in Mandelbrot set rendering by parallelizing GPU threads on NVIDIA GPUs with CUDA and AMD Radeon GPUs with ROCm HIP, accelerating the CPU-based C++ implementation by 156 seconds.
- Dynamically scaled blocks with image dimension, capable of generating an 25k x 25k resolution gray scale image in 2s.
- Optimized memory transfer using unified memory and reduced redundant operations by caching results in the kernel.

Kernel Resource Monitor | *C, Linux Drivers, Threading, Raspberry Pi* — **December 2024**

- Developed a Linux kernel driver in C to monitor CPU and memory usage of a Raspberry Pi for user level applications.
- Implemented thread-safety using kernel mutexes and background threads to periodically update cached metrics.

16-Bit CPU Design | *VHDL, Assembly, Xilinx Vivado, Digital Logic Design* — **November 2024**

- Designed and implemented a 16-bit CPU with a Finite State Machine (FSM) based Controller and dual-ALU Datapath architecture, supporting SIMD and custom assembly instructions like LDI, STR, ADD, SUB, JMP, and NOT.
- Ran a custom key-based encryption program using an assembly-like instruction set to validate CPU functionality.
- Validated CPU with testbenches covering edge cases, reset conditions, arithmetic operations, and parallel SIMD workflows.

Vehicle Telemetry Server | GitHub | *Python, Go, gRPC, Kafka, Docker, React, Postgres* — **October 2024**

- Designed a real-time vehicle data streaming service using a gRPC service in Python, Go, Apache Kafka, and Docker to simulate data transmission from IoT diagnostics on 1000+ vehicles to a PostgreSQL database, visualized on a React webapp.
- Developed Kafka Pub/Sub services in Go to process vehicle data and insert it into the database for fleet analytics.

SKILLS

Languages: Python, C, C++, Go, SQL, MATLAB, CUDA, HIP
Tools & Technologies: Linux, Git, GitHub Actions (CI/CD), Docker, Flask, Apache Kafka, REST, gRPC
Cloud & Database: Google Cloud Platform (GCP), Terraform, PostgreSQL, MongoDB
Embedded & Hardware: FPGAs, Raspberry Pi, Arduino, Oscilloscopes (Keysight, Teledyne), AWG, Fiber Optics, SPI, TCP/IP