Nolan McCleary

250-944-0668 | nolancmccleary@gmail.com | | |

Stack

Languages: C, C++, Python, Verilog/SystemVerilog, Matlab

Development Tools: Git, BASH, Make, CMake, Verdi, Perforce, LTspice, GitLab CI/CD

Lab: JTAG, Logic Analyzer, Oscilloscope, Signal Generator

Experience

SerDes Firmware Design and Validation Intern
Sep 2024 – Present

Synopsys — *Toronto, Ontario*

- Architected and led the development of a JTAG-based chip validation SDK for use on a TSMC 3nm 224G Ethernet PHY and associated test chip at the specific request of a major customer. Derived a new bringup sequence for the chip during the development of this tool for use by the firmware team, lab teams, and customers during testing.
- Developed various Ethernet PHY calibration and channel adaptation routines in a baremetal C environment.
- Performed waveform-level inspection to quantify the runtime improvements of various compiler optimizations down to the individual CPU clock cycle, ultimately shortening the critical path of the firmware's initial calibration sequence by 8 percent.
- Wrote a custom Python script to automatically parse generated assembly files and compare used ICCM region sizes to those allocated by the firmware codebase's linker script, allowing the detection of over-buffered regions in memory which led to a 15 percent reduction in overall firmware binary size.
- Refactored the e224g firmware codebase and migrated the build system from Make to CMake.
- Performed post-silicon validation of various firmware calibrations and channel adaptations on a TSMC 3nm test chip. Analyzed eye diagrams, histogram/scatter plots, and bathtub curves in order to debug chip-level issues and quantify firmware improvements.

Embedded Software Intern
May 2023 – Aug 2023

General Dynamics — *Ottawa, Ontario*

- Developed three separate Linux drivers in C++ to acquire, decode, and stream critical Zynq Ultrascale+ MPSoC (Xilinx) information over TCP through a series of custom JSON-RPC API endpoints. This included real-time monitoring of processor temperatures, Quad SPI (QSPI) partition boot status, and boot binary checksums.
- Engineered an automated build system via Buildroot to allow a consolidated Ultrascale+ firmware core to be built without any external board support package. Key components to be built included an Embedded Linux kernel (PetaLinux), FPGA image, bootloader, the root filesystem, and all relevant baremetal applications. Created a GitLab CI/CD pipeline using this build system to perform static code analysis on the firmware core across all used Xilinx toolchains concurrently, greatly accelerating the new core feature integration process as a result.

Embedded Systems Intern
May 2022 – Dec 2022

Teck Resources Limited — *Sparwood, BC*

- Designed, built, and tested a remote voltage spike suppression and detection system with functionality for high-speed transient capture, Ethernet networking, and dynamic system network parameter reconfiguration over Ethernet. This voltage monitor was deployed across four different sites and helped to successfully suppress and identify the source of previously unidentified transients occurring at operationally critical locations inside heavy equipment, saving up to $100,000 in previously lost throughput on each mitigated transient occurrence.

Wildland Firefighter
May 2020 – Aug 2020; May 2021 – Aug 2021

Government of British Columbia — *Fort Nelson, BC; Vanderhoof, BC*

- Two summers of initial attack firefighting with the BC Wildfire Service, the second being the second most destructive season in The Province's history.
- Directed helicopter-based remote water transport and deployment operations.
- Supervised contract personnel during multi-day incidents in a fast-paced, high-pressure environment.

Projects

Multicore RC4 Decryption Engine

Link to source code and documentation *June 2024*

- Used SystemVerilog to develop a custom decryption engine for implementing the RC4 stream cipher encryption/decryption algorithms on an Altera Cyclone V FPGA.
- Parallelized decryption process across 90 individual decryption cores in order to greatly accelerate system performance and successfully break encrypted messages with key lengths of up to 28 bits.
- Implemented a pipelined architecture to allow a faster reference clock frequency to be used during many-core builds.

Laser Projector

Link to source code and documentation *February - April 2024*

- Designed, programmed and tested an ARM MCU based laser drawing robot.
- Utilized hardware timers and interrupts along with digital filtering to facilitate hard real-time PID control of the laser beam via the simultaneous operation of two DC motors.
- Employed direct memory access (DMA) along with hardware buffering to ensure reliable and high-speed acquisition of laser coordinate data over a USART connection. This approach allowed for the real-time streaming of arbitrarily large laser image files to the controller, ensuring scalability by removing any constraints that would otherwise be imposed by storing data directly on the system's flash memory.
- Developed a USART-based laser image preprocessing and streaming application in Python to reduce computational load on the controller and ensure compatibility with any file following the ILDA laser image standard.

Education

University of British Columbia Vancouver, BC

Bachelor of Applied Science in Electrical Engineering *Graduating 2026*