Nahjay Battieste

954-494-7138 nahjaybattieste@gmail.com www.linkedin.com/in/nahjay-battieste-a84655224 https://github.com/Nahjay
Portfolio | https://nahjay.github.io/

Education

University of Florida **Aug. 2024 – Expected May 2027**
Bachelor of Science in Computer Engineering | GPA: 3.92 *Gainesville, Florida*

Broward College **Aug. 2022 – May 2024**
Associates of Arts | GPA: 3.95 *Coconut Creek, Florida*

Technical Skills

Languages: C, C++, Python, VHDL, System Verilog, Rust, Assembly, Bash, PowerShell, MatLab
Developer Tools and Technologies: VS-Code, Git, Jira, Perforce, Quartus, Vivado, Postman, Linux, Docker, Makefiles
Frameworks: PyTorch, CUDA, TensorFlow

Experience

SmartSystems Lab — University of Florida **October 2024 – Current**
Undergraduate Researcher *Gainesville, Florida*

- Researching hardware acceleration for machine learning models by identifying computational bottlenecks and mapping performance-critical kernels to FPGA-based custom logic.
- Implementing hardware–software co-design strategies to optimize dataflow and execution efficiency for ML inference workloads.
- Collaborating with graduate researchers to explore novel SoC architectures and hardware design methodologies that improve throughput, latency, and energy efficiency in ML systems.

Undergraduate Teaching Assistant — University of Florida **January 2026 – Current**
Digital Design (EEL4712C) *Gainesville, Florida*

- Assisted in instruction of digital logic and hardware design concepts including FSMs, datapath/control separation, pipelining, and timing analysis using VHDL and System Verilog.
- Led lab support sessions and debugged student designs involving RTL simulation, synthesis, and FPGA deployment.
- Reviewed and graded assignments focused on register-transfer level design and hardware verification.

NVIDIA **May 2025 – August 2025**
GPU Firmware Engineer Intern *Santa Clara, California*

- Developed a remote testing pipeline to validate firmware on target hardware, automating GPU ROM data extraction to configure test environments and execute Pytest suites.
- Created an automated regression hunting tool that performs a binary search across Perforce changelists to pinpoint commits responsible for build failures or performance degradation in the firmware.
- Enhanced a containerized development environment for firmware builds and compilation, building upon a previous project to further streamline the build process to increase productivity and reduce errors for engineers.

NVIDIA **May 2024 – August 2024**
GPU Firmware Engineer Intern *Santa Clara, California*

- Developed a wizard and series of scripts using Python, Batch, Bash, PowerShell, and Docker to create custom containerized environments used to build and develop our firmware packages and their microcode components, significantly reducing the ramp-up time for new team members and drastically reducing development time for existing team members.
- Optimized boot instantiation sequence in C for next-gen GPU architecture, removing obsolete code sections and integrating new architectural features to ensure compatibility and performance.
- Identified and resolved MASM errors in the build process for core architectures using assembly language. Updated the assembler version, applied correctly sized directives to struct instantiations for ROM data tables, and optimized assembly files for improved performance.

Cell Antenna Corporation **May 2023 – Apr 2024**
Software Engineer Intern *Coral Springs, Florida*

- Engineered a full-stack direction-finding system using FastAPI and Rust for the backend, and a JavaScript web interface, to visualize real-time signal data from an antenna array.
- Developed a GUI (Python/DearPyGui) and a Rust-based asynchronous API (Actix-Web) to streamline control of a Yocto Linux-based SDR and a GPS device, improving user efficiency.
- Wrote and optimized time-critical firmware in Assembly for an Atmel microcontroller synchronized to a GPS PPS signal and developed a Rust program to process raw IQ data from an FPGA-based SDR.

Projects

Jetson Nano Custom Kernel Modules | *C, CUDA*

- Integrated a Linux kernel module with GPU-accelerated CUDA kernels into an application for image processing on the NVIDIA Jetson Nano.
- Designed and optimized CUDA kernels to exploit GPU parallelism, achieving significant performance gains over CPU-only execution.
- Managed kernel–user space coordination and data movement to support efficient CPU–GPU execution.

3×3 CNN Convolution Accelerator (FPGA) | *VHDL, Python*

- Designed a fully pipelined FPGA-based accelerator for 3×3 RGB convolution, achieving one output per cycle after window fill pipeline latency.
- Architected a packed RGB memory format to reduce memory accesses and simplify address generation and controller complexity.
- Verified cycle-accurate behavior using a Python reference model and VHDL testbench with valid-bit tracking across pipeline stages.

32-bit MIPS-based CPU Design | *VHDL*

- Designed and implemented a 32-bit, 5-stage pipelined MIPS-like processor in VHDL with a custom ISA and hazard-aware pipeline control.
- Built instruction and data memory subsystems with word-aligned addressing and peripheral interfacing for program execution.
- Validated functional correctness and timing through simulation and successful deployment on an FPGA board.

Critical Analysis of Cache Replacement Algorithms | *C++*

- Developed a C++ cache simulator to compare LRU and LFU replacement policies across large-scale ChampSim memory traces.
- Implemented efficient cache data structures using hash maps and linked lists to support low-latency lookup and eviction.
- Collected and analyzed cache performance metrics including hit/miss rates and average memory access time.