

Arminster Singh

[redacted], New Jersey, [redacted]
arminders208@outlook.com |[redacted]

EDUCATION

Stevens Institute of Technology
Bachelor's of Engineering in Computer Engineering
GPA: [3.501 / 4.0]

May 2025
Hoboken, NJ

Relevant Coursework: VLSI Design, Computer Architecture, Data Structures and Algorithms, Operating Systems, Embedded Systems

PROJECTS

Project Mu port for Apple silicon platforms June 2022 – Present

- Bringing up [Project Mu \(Microsoft's UEFI firmware\) for Apple silicon platforms](#) in collaboration with other developers.
- Completed very early UEFI bringup by writing ARM64 assembly code that initializes the stack and C code that initializes the UEFI memory map.
- Created one of the only non-GIC interrupt controller UEFI modules, along with maintaining and designing 9 other C-based modules.
- Successfully booted Windows PE on Apple silicon Macs with a very small amount of emulation via hypervisor traps.
- Progressively writing drivers in order to remove the need for most emulation.
- Successfully debugged the firmware to fix memory allocation errors in some of the UEFI modules.

Bluetooth Audio Network September 2024 - May 2025

- Designed a Sonos-like Bluetooth-powered audio network using Renesas hardware.
- Brought up the internal DAC on the microprocessor dev board against a tight deadline referencing data sheets and sample code to initialize I2C properly.
- Solved issues with DAC pin assignments by tracing clock signals using schematics and PCB design files, along with a logic analyzer.
- Validated the sample sine wave was being output correctly at the right frequency using an oscilloscope.

[redacted] Glitching Controller May 2026 - Present

- Writing a Raspberry Pi Pico-based program to glitch [redacted] to reproduce publicly available glitching research on [redacted].
- Single cycle accurately testing for the glitch anchors by implementing PIO code by hand to run on the RP2040 microcontroller's PIO state machines.
- Implementing an efficient command parser over the RP2040's UART port to allow a host computer to script fault injection campaigns.

[redacted] Shellcode April 2026 - Present

- Writing shellcode to run on [redacted] to reproduce publicly available glitching research on [redacted].
- Compactly communicating shellcode status via [redacted] by using existing error reporting functions in the program.
- Split the shellcode payload into a very small assembly-only stage 1, working on building a larger C-based stage 2 loaded on-demand.

'Breakout' Recreation on an FPGA March 2024 - May 2024

- Remade the game "Breakout" on an FPGA using Vivado and VHDL.
- Solved critical issues with the implementation including the ball passing through the paddle.