Deric Dinu Daniel

dericdinu03@gmail.com | deric.dev | github.com/dericdinudaniel | linkedin.com/in/dericdinudaniel

Education

University of Michigan – Ann Arbor: *Dean's List, University Honors* Aug. 2021 – May 2025
Computer Science B.S.E.; GPA: 3.66 *Ann Arbor, MI*
Relevant Coursework: Operating Systems, Advanced Computer Architecture, Data Structures & Algorithms, Web Systems, Parallel (GPU) Programming, Computer Security, Computer Science Theory, Intro to Circuit Analysis, Music Production

Experience

Apple Starting July 2025
Incoming SoC Embedded Software Engineer *Cupertino, CA*

Microsoft June 2024 – Sept. 2024
Software Engineer Intern *Redmond, WA*

- Integrated support for new data source (Microsoft Dataverse) in security/compliance platform (Microsoft Purview) to enable data loss prevention on new asset category, furthering protection and security compliance efforts for customers.
- Utilized **C#** to fulfill business logic to support Dataverse locations, used **React/Typescript** to build intuitive asset picker component, and integrated new asset discovery API to enable end-to-end integration with new data source.
- Wrote unit tests, performed end-to-end integration tests, and ran A/B tests for flagged features in dogfood and production environments.
- Supported migration of existing API to **ASP.NET Core** to improve cost-effectiveness and scalability through containerization.

Bose – Research Division Jan. 2024 – June 2024
Embedded Systems Software Engineer Intern *Framingham, MA*

- Designed real-time streaming system for multiple mics and sensors using LC3 codec, Qualcomm audio system, and **Bluetooth LE** in a large-scale **multithreaded embedded system** in **C**, supporting nearly every high visibility wearables research project.
- Enabled ability to run larger ML models on host device (vs headphones/earbuds), multichannel audio and sensor data collection, and high-quality audio recording during music playback and voice calls, creating new (upcoming) user experiences.
- Researched and presented effects of LC3 codec on quality performance of audio (speech separation/noise reduction) deep learning models using SISDR, STOI, and PESQ metrics in an automated **Python** script.
- Prototyped **Bluetooth** LE GATT receiver system on Infineon Cypress Microcontroller to enable rapid testing of mic and sensor streaming interface to maximize bandwidth and minimize packet loss. Also resulted in accelerated development of iOS streaming receiver app.

University of Michigan EECS 482: Operating Systems Aug. 2024 – Present
Instructional Aide (TA) *Ann Arbor, MI*

- Taught Operating Systems to over 400 students covering use and implementation of multithreading, virtual memory, and filesystems.
- Assist students in designing and debugging large scale C++ projects in office hours, online question boards, and group mentorship.

Siemens May 2023 – Aug. 2023
Software Engineer Intern *Troy, MI*

- Managed database and environment instances in **AWS** using **EC2** and **S3** while reducing development server costs by up to 5% to deploy testing and demo environments.
- Resolved critical bugs in dropdown menus on **C++** server-side code by refactoring with smart pointers, fixing product inconsistency and resolving 8 memory leaks per dropdown interaction.

Projects

Out-of-Order RISC-V Processor (Synthesizable) | *SystemVerilog, RISC-V, Register-Tranfer-Level (RTL), Design Verification (DV)*

- Designed RTL and DV testbenches for a RISC-V CPU with out-of-order & N-way superscalar execution, early tag broadcast, fast branch recovery, early branch resolution, store queue (store → load fwding), non-blocking dcache, Gshare branch prediction, branch target buffer, and instruction prefetching. Achieved **11.5ns clock period** and average **1.3 CPI**.

Visual CPU Debugger | *Next.js/React, Typescript/Javascript, Python*

- Built visual debugger for above CPU core using Python to parse VCD files, and Next/React to display all CPU modules.

Thread Library | *C++, Multi-threading, Mutexes, Condition Variables, Semaphores, Unix*

- Implemented a kernel C++ thread library on Unix, handling CPU booting, thread management, management of 50+ CPUs, interrupts, atomicity, and FIFO scheduling order. Designed spin-locks, mutexes, conditional variables utilizing advanced Unix context management.

Virtual Memory Pager | *C++, Virtual Memory, Page Faults, Process Lifecycle Management*

- Designed a virtual memory pager which managed multiple processes and supported swap-backed and file-backed memory pages (similar to mmap()). Managed process creation/forking/destruction, page faults, MMU bits, and swap disk all while supporting copy-on-write.

Multithreaded Network File Server | *C++, Boost Library, Threads, Sockets*

- Built a heavily concurrent, crash consistent network file server supporting multiple users and nested file/folder structure.
- Utilized committing writes to enable crash consistency, Boost threads and upgradeable reader-writer locks to optimize for maximum concurrency, and POSIX sockets to enable network communication with clients.

Delay Audio Effect Plugin | *C++, JUCE Library, Digital Signal Processing*

- Built delay effect DAW plugin using JUCE. Features: DAW tempo sync, stereo ping-pong delay, low & high cut, dry/wet mix, and gain control.
- Implemented custom circular buffer delay line with Hermite-interpolation, one-pole filtering for analog emulation, linear parameter smoothing to reduce zipper noise, and delay time crossfading to reduce artifacts

Technical Skills

Languages : C++, C, Python, RISC-V, C#, HTML/CSS, Javascript/Typescript, SQL, SystemVerilog
Technologies : Next.js/React.js, Git, gdb, Linux, Flask, MapReduce, Sockets, Multithreading, Networks, JUCE, CUDA
Misc. : Ableton Live, FL Studio, Adobe Lightroom, Pioneer Rekordbox
Interests : Music Production/Audio, Consumer Tech, Soccer/Running/Gym