Eryl Kenner

linkedin.com/in/eryl-kenner • Eryl.Kenner@gmail.com • (541)-631-1910 • http://ErylKenner.com

Professional Experience

System Software Engineer **2021 - Present**
Intel Corporation ***Folsom, California***

- End-to-end architect, project manager, and lead developer of a new set of multithreaded tools for validating cache coherency and concurrency on upcoming SoC's. Memory, PCIe, and DMA traffic is sent concurrently for prolonged periods across cores/sockets in order to stress LLC cache, cause cache evictions, and hit corruption/deadlock scenarios. The solution is written in Python and C, and runs both on Linux and bare-metal.
- Architected and led development of a new concurrency framework for synchronizing OS-based applications by phase (setup, execute, verify, etc.) to maximize phase overlap. The tool enabled validation engineers to explore validation space by randomizing any application parameter in a steerable way via custom config files. The tool also provided a JavaScript visualizer which validations engineers used to see predicted and achieved phase overlap.
- Lead architect of the horizontal software tools team, in which I collaborated with stakeholders across multiple validation teams and architected, led, and developed OS-based and bare-metal frameworks, tools, and libraries (primarily C and Python) for validating Intel Foundry SoC's in pre-si simulation/emulation and post-si.
- Wrote a PCIe test-card library in C which was used for bare-metal pre-si system-level emulation and post-si validation.
- Developed a Python library for easily accessing SoC components over JTAG via APB-AP/AXI-AP protocols, which was used for pre-si system-level emulation validation for multiple upcoming SoC's.

SSD Validation Intern **2020 - 2021**
Intel Corporation ***Folsom, California***

- Developed validation software to test solid-state drive features and NVMe spec compliance in Python.

Embedded Software Engineering Intern **2017, 2018, 2019**
Lime Rock, LLC ***Medford, Oregon***

- Designed, implemented, and authored a whitepaper of a novel real-time, precise dead reckoning system for a four-wheel holonomic chassis. The multiprocessing solution used a combination of inverse kinematics and a custom-implemented Newton's method algorithm (written in C, utilizing BLAS and custom linear algebra functions) in order to prevent instability when solving the overdetermined, nonlinear system. A working demo of the solution was presented to management and was chosen for use in a new DOD contract for a plane inspection vehicle.
- Created a point-to-point real-time graphical web user interface for a GPS controller
- Implemented a real-time C parser for decoding the NMEA 0183 communication standard

Artificial Intelligence Researcher **2018 - 2020**
Evolutionary Computing Systems Laboratory ***University of Nevada, Reno***

- Implemented an algorithm for path planning and navigation of an autonomous bridge inspection robot using RRT (Rapidly exploring Random Trees) which was tested in simulation.
- Developed a VR, multiplayer network-based training simulation for naval officers in C# using Unity
- Researched genetic algorithm techniques for unsupervised learning of the game Starcraft II.

Education

BS in Computer Science and Engineering
University of Nevada, Reno

Honors Student - 3.75 GPA, Minor in Mathematics, Minor in Digital and Interactive Games

Technical Skills

Python, C, C++, JavaScript, C#, bash, Make, Git, OOP, design patterns, Unity, LaTeX, SolidWorks

Personal Projects

Planet Ball - *2018* - Winner of the “Games” category in the 2018 UNR Hackathon

Competitive game with a unique movement system where players grapple onto planets to change directions. (C# in Unity)