Tyler Jones

Contact Information

Phone: (970) 261-9425 Email: me@tjones.dev Github: tjones879

Work Experience

AMERGINT Technologies

Systems Engineer, Launch/Range Systems **July 2018 – Present**

Designed, implemented, and delivered mission-critical and safety-critical systems on strict timelines written in C++ and Python.

- Lead the design and development of more than $8 MM of novel hardware-in-the-loop testing platforms for modern hypersonic and missile defense programs.
- Designed and implemented a next-generation software-based LDPC decoder architecture capable of multi-gigabit downlink rates.
- Optimized the performance of telemetry processing decommutation and commutation subsystems by up to 20x for use in range safety environments.
- Built a new line of IRIG 106 Chapter 10 data recorders for rapid development of multi-gigabit data recording and processing product lines.
- Lead development of an embedded, network-enabled IO and RF test equipment product line.

FFmpeg

Google Summer of Code Participant – Vorbis Encoder **Feb 2017 – Aug 2017**

Improved the encoding quality and speed of FFmpeg's native Vorbis encoder by redesigning major components and implementing new psychoacoustic models.

- Built a custom psychoacoustical model capable of detecting transient signals and dynamically switching encoding modes.
- Implemented noise normalization to generate a perceived gaussian noise profile and designed residue encoding to minimize quantization error.
- Rewrote the stereo coupling to dynamically switch between various lossy and lossless encoding modes.

Other Projects

Flying Drone

Built an autonomous drone from bare components and a custom real-time operating environment.

- Utilized OFDM with a custom packetized protocol for efficient command, control, and telemetry in ISM bands.
- Built Linux kernel drivers to minimize the latency of subsystem components in centralized ground stations.
- Developed full environment and sensor simulations for faster iteration of state estimation and vehicle localization algorithms.

Stream Processing Language

Designing a novel compiler and domain specific language for efficient synchronous data flow processing.

- Employed scalar evolution analysis for alias detection and identification of instruction-level parallelism.
- Designing algorithms for aggressive auto-vectorization and pipelining of numeric code.
- Implementing optimized static scheduling of stream processing applications to maintain memory locality and efficient cache utilization.

College Education

Colorado Mesa University, Grand Junction, CO

B.S. Computer Science GPA - 4.0 **Aug 2014 – May 2018**

Technical Skills

Languages: C, C++, Python, Rust
Tools: Git/SVN, Valgrind, GCC/LLVM, perf, eBPF, Docker
Other Knowledge: Linux and RTOS Kernel Development, Embedded Linux, Software-defined radio