ENTROPY TECHNOLOGY DESIGN, INC.

Senior FPGA / Firmware Engineer

Job Description | Full-Time, Part-Time, Employee, or Contractor Options

Field	Description
Primary Focus	FPGA implementation, firmware interfaces, real-time processing pipelines, AMD/Xilinx workflows, ADC/data acquisition integration, and transition planning to future SoC-based hardware.
Work Style	Hands-on technical contributor in a small, fast-moving defense-adjacent technology company.

Role Summary

Entropy is seeking a Senior FPGA / Firmware Engineer to implement and validate signal-processing capabilities on FPGA-based hardware. The current platform includes AMD ZCU-104 hardware, with a planned roadmap toward future SoC-based platforms. This role will work closely with DSP engineers to translate detection and classification algorithms into real-time, testable, field-ready implementations.

Why This Role Exists

This role is intended to provide focused FPGA and firmware depth in SDR, embedded software, tooling, field support, and DSP. The ideal candidate can make sound hardware/software partitioning decisions, implement reliable firmware, and support integration into Entropy's operational sensing systems.

What You Will Do

- Partner with DSP Engineers to translate detection/classification algorithms into FPGA-accelerated and embedded implementations.
- Implement, test, and optimize real-time processing pipelines with attention to latency, throughput, buffering, memory constraints, and repeatability.
- Build and maintain firmware/software interfaces between ADC/data acquisition, FPGA fabric, and host-side applications and APIs.
- Support the roadmap transition from AMD ZCU-104 to future SoC platforms, including system partitioning, performance tradeoffs, and integration planning.
- Use AMD/Xilinx workflows such as Vivado and IP Integrator, Vitis HLS, Vitis, Zynq, MPSoC, or similar tools to support development, bring-up, debugging, and validation.
- Support bench testing, regression testing, configuration baselines, and troubleshooting of FPGA/firmware behavior.
- Produce clear engineering artifacts including build notes, configuration baselines, test procedures, troubleshooting notes, and implementation records.
- Collaborate with hardware, DSP, embedded software, and field integration teammates to move prototypes toward customer-ready capability.

Required Qualifications

- Strong hands-on experience building firmware or embedded systems that interact with FPGA fabric, hardware accelerators, ADCs, and data acquisition systems.
- Experience implementing and debugging real-time data pipelines where timing, buffering, throughput, and memory use matter.
- Proficiency in C/C++ and Python in a Linux-based engineering environment.
- Working knowledge of FPGA development workflows, preferably AMD/Xilinx or comparable platforms.
- Ability to collaborate with DSP engineers to implement algorithms, validate signal integrity, and debug performance regressions.
- Comfortable working in a small company environment with evolving priorities, short deadlines, and limited layers of support.
- Strong documentation discipline and ability to communicate technical decisions clearly.

Preferred Qualifications

- Experience with AMD/Xilinx Zynq, MPSoC, Vivado, Vitis, ZCU-104, and related platforms.
- Experience with DAQ systems, target acquisition systems, sensing platforms, SDR-adjacent pipelines, and fielded sensor systems.
- Exposure to weak-signal detection, sub-MHz sensing, noise mitigation, and real-time signal-processing environments.
- Python experience for test harnesses, validation scripts, automation, and engineering tools.
- Git-based workflows, CI/release discipline, regression testing, and configuration control.
- Exposure to defense-adjacent development constraints such as access control, documentation rigor, customer delivery readiness, and technical data discipline.
- Ability to translate DSP algorithms written in C# into Vitis HLS using C or C++.
- Experience in producing IEEE IP-Xact IP cores from HDL or HLS
- Experience writing hardware abstraction layer of software for custom designed IP cores
- Experience using Vitis to create Platform project from exported Vivado design and write test applications.

Success Measures

- Algorithms are implemented in a testable, repeatable, and performant FPGA/firmware environment.
- System interfaces between acquisition, FPGA fabric, and host-side applications are stable and documented.
- Latency, throughput, buffering, and memory constraints are measured, understood, and improved where needed.
- Builds, baselines, test procedures, and troubleshooting notes are clear enough for the team to reuse.
- Prototype capability advances toward field-ready/customer-ready operation without unnecessary technical debt.