DAVID EASON SMITH

Collegeville, PA 19426 | 917.548.1006 | davideasonsmith@gmail.com
GitHub: @EasonNYC | LinkedIn: @in/easonsmith | Portfolio: EasonRobotics.com

Senior Embedded Software Engineer with 10 years of experience shipping commercial products across IoT, automotive/EV, health tech, defense, and robotics. **Patent holder** in AI health monitoring. Deep expertise in ARM firmware, embedded Linux, real-time systems, sensor integration, and full product lifecycle from prototype through production deployment.

TECHNICAL COMPETENCIES

Languages:	C/C++ (16 yr, primary), Python, Bash, ARM Assembly
Platforms:	Embedded Linux (Yocto, Ubuntu), FreeRTOS, ROS2, Bare-Metal ARM (STM32, NXP, TI)
Protocols:	I2C, SPI, UART, CAN, Modbus, BLE, WiFi, TCP/IP, MQTT, UDP, USB, Ethernet
Systems:	Board Bring-Up, Device Drivers, Sensor Fusion, LiDAR, GPS/IMU, Radar, OpenCV, GStreamer
Tools:	Git, JIRA, GitLab CI/CD, Grafana, Make/CMake, Altium, JTAG/SWD, Oscilloscopes, Logic Analyzers
Cloud/Build:	AWS IoT, Cross-Compilation (ARM64), U-Boot, Linux Kernel Config, Docker

PROFESSIONAL EXPERIENCE

Senior Embedded Software Engineer | Treeswift *(Robotics & Environmental Tech)* Philadelphia, PA | Aug 2025 – Present

Serving as functional technical lead for the embedded software team (one senior and one junior engineer), guiding workflow, architecture, and code review at a robotics company building backpack-mounted mobile sensor platforms for environmental field data collection. Own the full embedded software stack: sensor drivers, ROS2 middleware, embedded Linux, and build systems for ARM-based platforms integrating Ouster LiDAR, SBG GPS/IMU, and Insta360 cameras.

- **Platform Migration:** Led complete platform migration from ROS1 to ROS2 Jazzy, porting 20+ driver nodes, PyQt application layer, state machine, and diagnostic infrastructure
- **Sensor Drivers (C++/Python):** Engineered multithreaded Ouster LiDAR driver with dedicated UDP receiver, packet processor, and async writer threads—eliminating packet drops during high-rate 3D point cloud capture
- **Board Bring-Up:** Drove next-gen ARM64 platform bring-up: kernel configuration, thermal profiling, GPIO subsystem migration, networking, and peripheral integration
- **Build Systems:** Architected automated build pipeline for custom Ubuntu 24.04 + ROS2 Jazzy images with ARM64 cross-compilation
- **OTA Update System:** Designed a Docker-based A/B field update system: converted the OS image from MBR to GPT, built dual independently-flashable image slots with manifest-driven seeding, enabling atomic rollback-capable updates delivered via SD card without network connectivity
- **Data Integrity:** Implemented real-time Built-In Test (BIT) monitoring on the SBG INS that detects accelerometer/magnetometer hardware faults and spoils affected recordings before they reach the cloud pipeline, after diagnosing a silent sensor failure that had produced unusable field data
- **Time Synchronization:** Implemented end-to-end PPS time synchronization across GPS/INS, LiDAR, and the system clock using Chrony and hardware PPS signals, resolving systematic timestamp misalignment in georeferenced point-cloud output; built a custom single-node GPS/IMU lifecycle driver from scratch
- **Release Management:** Shipped 6 production releases across 2 hardware platforms for enterprise fleet customers; cut image build time ~5x (5+ hr to ~1 hr); 90+ tickets and 390+ commits across two repositories

Embedded Software Engineer | InductEV *(Automotive & EV Charging)* King of Prussia, PA | Jul 2023 – Jul 2025

Part of the embedded team developing a wireless charging system for electric buses, cars, and yard tractors. Worked across interconnected ARM-based TI microcontrollers and embedded Linux platforms on ground-side high-voltage inductive charging and vehicle-side FLiR camera foreign object detection.

- Implemented vehicle alignment algorithm providing real-time driver feedback for safer positioning over the ground charging coil
- Discovered and resolved grounding issue in FLiR camera-based Foreign Object Detection system (OpenCV), preventing potential field failures across 39 deployed cabinets
- Debugged cellular modem connectivity issues using low-level AT commands; fixed device enumeration via custom udev rules
- Evaluated and deployed a more reliable hardware watchdog timer across the production fleet, improving cabinet uptime monitoring
- Created 5 PMM (Power Management Module) firmware releases; improved OCPP and network/modem logging for field diagnostics

Embedded Software Engineer | Miku Inc. *(Health Tech & IoT)* Woodbridge, NJ | Oct 2017 – May 2023

Core engineering team member at a health-tech startup that shipped multiple generations of an AI-powered IoT baby monitor with contactless radar-based breathing detection and AWS IoT cloud integration. C/C++ in a multithreaded Yocto Linux environment.

Miku Pro (Version 2)

- Co-invented AI algorithm for contactless illness detection (US20240206745A1; allowed, issuance pending) combining radar, computer vision, and ML for early fever detection during sleep

- Designed and built the complete WiFi/BLE onboarding and pairing system from scratch: 31,600+ lines of C++ across GATT server, WPA supplicant integration, AES-256 encryption, and state machine architecture
- Developed medical-grade 24/7 video recording system for hospital-based infant seizure research, contributing to clinical findings published at the American Epilepsy Society conference
- Implemented room ambiance audio level capture and processing pipeline (GStreamer)
- Created Python GUI tools enabling non-technical warehouse staff to perform mass camera firmware updates

Miku Original (Version 1)

- Architected complete GStreamer-based audio pipeline from scratch: real-time capture, music/white noise playback, two-way talk, and software audio mixing
- Performed hardware and software bring-up for cameras, ambient light sensors, IR LEDs, speakers, and microphones
- Led extensive memory leak remediation across Algorithm Audio, Video, and Radar modules; converted raw pointers to smart pointers and added proper cleanup paths

Embedded Software Engineering Intern | Persistent Systems, LLC *(Defense)* New York, NY | May 2016 – Sep 2016

- Performed board bring-up, HW/SW debugging, and schematic review for an NXP ARM Cortex-M based HDMI-to-MyDP peripheral on the MPU5 military/civilian radio platform
- Ported 40,000+ lines of 8-bit 8051 C code to a pin-compatible 32-bit NXP ARM Cortex-M microprocessor using Kinetis SDK
- Cross-compiled NXP x86 bootloader utilities for ARM-Linux kernel; created auto-upgrade firmware feature for cable-detect events

PATENT

Co-Inventor, US Patent Application US20240206745A1 (Published 2024)

"System and Method for Monitoring a Person for Signs of Sickness"

- Contactless health monitoring system using low-power radar, IR imaging, and audio to detect illness onset during sleep through biometric pattern recognition
- Multi-modal sensor fusion with ML-driven analysis to distinguish normal sleep patterns from illness-related biometric deviations
- Real-time alerting system with cloud connectivity and mobile app integration for caregivers

EDUCATION

New York University | Tandon School of Engineering May 2017

Dual Degree B.S. in Electrical and Computer Engineering

Relevant Coursework: Real-Time Embedded Systems (Grad.), Artificial Intelligence (Grad.), Sensor-Based Robotics (Grad.), Feedback Control, Data Structures & Algorithms

LEADERSHIP & NOTABLE PROJECTS

NASA Robotic Mining Competition – Team Captain & Project Manager 2014 – 2015

- Led a team of 13 BS and MS students to design, build, and compete with an autonomous lunar mining robot at NASA Kennedy Space Center
- Designed the robot WiFi communications link with watchdog timer safety feature and TCP/IP auto-reconnect; featured in NYU Engineering press

NYU CubeSAT – Embedded Systems Engineer 2016 – 2017

- Built the mission science payload for a 1U mini-satellite: FreeRTOS on STM32, custom PCB design in Altium, real-time multi-sensor data acquisition (GPS, Geiger counter, IMU, humidity, pressure)

Camelot AI: 5-DOF Robotic Manipulator & Game-Playing AI Engine | github.com/EasonNYC/Camelot

- Python-based AI agent controlling a robotic arm using iterative deepening, alpha-beta pruning, and move ordering in VREP simulation