

Artur Andrzejczak

Principal Software Engineer: Embedded Linux, Kernel Drivers & Camera Systems

Łódź, Poland | andrzejczak.artur@gmail.com

linkedin.com/in/artur-andrzejczak-aa1a2630 | github.com/akandr | [Google Scholar](https://scholar.google.com/citations?user=akandr) | akandr.cc

Professional Summary

Principal Software Engineer with 15+ years in embedded software and Linux kernel development, working on automotive camera and image-sensor drivers for in-cabin ADAS: driver monitoring (DMS) and occupant monitoring (OMS). Develops the V4L2 kernel camera drivers used in HARMAN's driver- and occupant-monitoring programs: sensor bring-up, GMSL2 SerDes and MIPI CSI-2 integration, board support (BSP) on NVIDIA Tegra/Jetson and Samsung Exynos Auto, diagnostics and power management, up to start of production. Also worked on embedded RTOS and bare-metal firmware, PCI kernel and userspace drivers, hardware design and debugging, plus lab and production test tooling. Uses coding agents (Claude Code, GitHub Copilot) in daily work.

Technical Skills

Languages: C, C++, Python, Bash, Assembly

Kernel & Embedded: Linux kernel & device drivers, board support packages (BSP), platform bring-up, device tree, kernel interrupts, DMA, I2C/SPI, sysfs, power management (suspend/resume, S2R), VFIO/UIO, PCI/PCIe (AER, SR-IOV PF2VF), Buildroot, JetPack / L4T, embedded RTOS (Contiki OS), bare-metal firmware, MCAL / AUTOSAR (automotive)

Camera & Imaging: Image sensor drivers (OmniVision, STMicroelectronics), IR / RGB / RGB-NIR modes, RAW pixel formats, MIPI CSI-2 (C-PHY / D-PHY), V4L2 subdev / media-controller, GStreamer (v4l2src, nvarguscamerasrc) and FFmpeg for capture validation and frame conversion, GMSL2 SerDes (Maxim MAX9295/9296, MAX96717/96724, MAX96716; pixel and tunnel modes), sensor bring-up, sensor boot and firmware upload, ISP / auto-exposure (AE), driver diagnostics and fault injection (functional safety, ASIL-B sensor), multi-camera synchronization, Android camera framework (Qualcomm)

Platforms: NVIDIA Tegra / Jetson (AGX Xavier, Xavier NX, Nano), Samsung Exynos Auto, Renesas RH850, Qualcomm Snapdragon (MSM89xx), ARM, Atmel AVR, MSP430, x86-64

AI-assisted development: Claude Code, GitHub Copilot, ChatGPT, Microsoft 365 Copilot

Tools & methods: git (Gerrit, GitLab, GitHub, Bitbucket/Stash), Perforce, GCC, GDB, Make, Autotools, Google Test / gMock, Coverity, Klocwork, CodeSonar, Bullseye, Jenkins CI, Black Duck (FOSS / SCA scans), Saleae, Introspect, Rohde & Schwarz, CANoe, DaVinci Configurator/Developer, Green Hills MULTI, Altium Designer, Sparx Enterprise Architect (architecture review), ALM (requirements and tickets); static analysis, unit testing, Agile / Scrum

Professional Experience

Principal Software Engineer

HARMAN International (Samsung) – Camera Team

Jun 2022 – Present

Łódź, Poland

Automotive Engineering Services, ADAS / Intelligent Cockpit. Ready Care, Ferrari and Bentley programs: driver monitoring (DMS) and occupant monitoring (OMS) systems. Camera-driver development, debugging and platform integration on NVIDIA Tegra and Samsung Exynos Auto, and technical support for the teams that build on those drivers.

Camera drivers & features

- Developed the vb56g4a (DMS) and vb1940 (OMS/Bentley) V4L2 kernel camera drivers, together with GMSL2 link, NvM, LED and under-display-camera (UDC) drivers. Both run on NVIDIA Tegra and Samsung Exynos Auto and cover IR, RGB and RGB-NIR modes, runtime-detected camera variants, and interchangeable Maxim GMSL2 SerDes combinations (MAX9295/9296, MAX96717/96724, MAX96716) with MIPI CSI-2 (C-PHY/D-PHY) capture, pixel and tunnel modes, and sensor boot and firmware upload.
- Designed the driver's power model as a multi-level sleep and wake state machine exposed to userspace, and delivered multi-camera cold boot, reset recovery, adaptive 3/6 Gbps GMSL2 link, DMS/OMS frame synchronization and dual-camera parallel operation.
- Built the diagnostics subsystem (temperature, PoC voltage, link status, eye-opening monitor, imager faults) with fault injection, feeding the functional-safety (FuSa) diagnostics of an ASIL-B image sensor; added NvM readout of camera identity and calibration data; integrated SAM-IS ISP auto-exposure into the Samsung camera framework.

- Debugged link and timing problems across the whole capture chain with Saleae, Introspect and Rohde & Schwarz: C-PHY/D-PHY behavior, pixel and tunnel mode, packet and pipe mapping, frame timing. Caught a wrong deserializer variant on prototype hardware early and kept development running on a 3 Gbps workaround until corrected boards arrived.

Platform integration, tooling & lab

- Brought up camera drivers on NVIDIA Tegra (Jetson AGX Xavier, Connect Tech carrier boards) and Samsung Exynos Auto, including device tree and kernel media-pipeline (Tegra VI / V4L2) integration. Built Solectrix frame-grabber init sequences, a frame-grabber NvM/OTP programming tool (also used to support production programming), a GMSL SerDes link monitoring tool, Image Quality lab tooling and Exynos/NVIDIA build and deployment scripts; contributed to RFQ handling.
- Set up a remote camera-driver development lab (remote power and lighting, IR-LED presence and exposure checks, remote oscilloscope, logic analyzer and flashing), which shortened validation and let the team work remotely on shared hardware. Wrote the automated load/unload stress harness and log analyzers used to qualify driver releases, and a self-check script that spots camera wiring, wrong-port and LED-power faults so other teams can get a camera running on their own.

Ownership & cross-team work

- Camera-driver point of contact for the program: support hardware (incl. EMC testing and HW2SW interface reviews), image quality, validation, homologation, feature and platform teams across several sites, and work directly with external vendors (Samsung, Analog Devices/ST, Solectrix).
- Turn incoming field and integration reports into reproduced, diagnosed and documented issues; review the team's driver code; represent camera software in program status meetings. Team lead of the three-person camera software team for a period.
- Brought GitHub Copilot into the camera team and help colleagues get going with coding agents; HARMAN "AI buddy".

Consultant / Expert Software Engineer

Jun 2019 – May 2022

Mobica (contracted to a leading microprocessor manufacturer)

Łódź, Poland

- Developed the Linux kernel and userspace driver (x86-64) for a PCI hardware cryptography and compression accelerator, within a 7-engineer team, and was its team lead for part of the engagement; handled PCI device probing, hardware detection and firmware loading from configuration.
- Migrated the driver from kernelspace UIO to userspace VFIO, and implemented PF-to-VF (SR-IOV) communication, sysfs configuration/status interfaces, kernel interrupt handling and PCI Advanced Error Reporting (AER). Diagnosed and fixed timing-dependent race conditions on new hardware revisions, segfaults across the driver and its Software Abstraction Layer middleware, and nightly-test failures.
- Profiled and optimized middleware startup (memory-pool initialization); built kernel-driver unit tests with Google Test/gMock, ran Bullseye coverage and Coverity/Klocwork static analysis, fixed Autotools build scripts and prepared a Fedora/RPM release. Earlier, delivered a benchmarked DPDK-based Vector Packet Processing (VPP) plugin in Linux userspace.

Senior Software Engineer

Jun 2016 – May 2019

Software Engineer

Aug 2014 – May 2016

HARMAN International (incl. Symphony Teleca) – Vehicle CPU Team

Łódź, Poland

NTG6 / NTG6@223 / NTG7, Daimler infotainment (Mercedes-Benz MBUX).

- Brought up custom automotive head-unit boards on Renesas RH850 and NVIDIA Tegra: low-level driver development, hardware and software integration, and MCAL configuration.
- Developed embedded C system software and complex device drivers on a real-time automotive stack (AUTOSAR): SPI, CAN / CAN-FD, watchdog, NvM and peripheral drivers (ADC, GPT, ICU, DIO, PORT, ETH).
- Implemented SPI inter-processor communication (shared-memory protocol), time-windowed watchdog task monitoring, and CAN-FD integration for a Continental rear-view camera.
- Before the Symphony Teleca acquisition (2014–2015): developed Android/Linux camera manufacturing tests on the Qualcomm camera framework (MSM8974/MSM8992, Snapdragon) in C/C++ for a phone manufacturer.

Embedded Software & Hardware Engineer

Alitec

Jul 2010 – Jun 2014

Łódź, Poland

- Developed embedded software in C (GCC) for ARM, Atmel AVR and Linux, and designed industrial control and measurement hardware (circuits and PCBs) in Altium Designer.
- Built a medical-transport tracking device (TI OMAP/BeagleBoard, Buildroot Linux, AVR watchdog), an RS-485/Modbus strain, temperature and humidity monitoring module (built for an industrial-hall roof, later reused with a Modbus display unit for crane-load measurement), and firmware for a 16-channel 24-bit measurement card (ARM7, Ethernet).

Embedded Systems Engineer / Researcher

Lodz University of Technology – Dept. of Microelectronics & Computer Science

Dec 2011 – Apr 2014

Łódź, Poland

- Designed and built the wireless sensor nodes of a distributed structural-health-monitoring and vehicle-weight-measurement system (TULCOEMPA): communication architecture, node hardware (strain-gauge front end, 24-bit ADC, low-power design) and node firmware (Contiki OS on MSP430 for IEEE 802.15.4 mesh routing and power management, ARM Cortex-M4 measurement firmware), plus the OMAP/Linux data sink (Buildroot).
- Presented at international conferences (DGM Best Paper Award), completed a research internship at Empa (ETH Domain), Zürich, and taught undergraduate courses (ARM microcontroller programming, PCB design, digital electronics, Java/C#).

Education

Doctoral (PhD) studies, Computer Science

2011 – 2017

Lodz University of Technology (Politechnika Łódzka) – research on wireless sensor networks and embedded Linux for measurement systems, published as first-author journal and conference papers; dissertation not submitted. University scholarships for top doctoral students in three consecutive years (2013–2015) and the Łódź Voivodeship doctoral scholarship “Nauka drogą do biznesu”.

M.Sc. Eng., Electronics & Telecommunication

2006 – 2011

Lodz University of Technology – specialization: Microprocessor Systems & Programmable Devices

Projects, Publications & Training

Open source: *ov5647-jetson* (github.com/akandr/ov5647-jetson): tegracam V4L2 driver and device tree overlays bringing the OmniVision OV5647 (Raspberry Pi Camera v1) to NVIDIA Jetson boards across L4T and JetPack releases, with raw V4L2 and hardware-ISP capture paths. *BC-250* (github.com/akandr/bc250): GPU compute on the AMD BC-250 (RDNA2 APU, GFX1013, 16 GB unified GDDR6) for local LLM inference (llama.cpp, Ollama, Vulkan) and stable-diffusion.cpp, with a companion *bc250-rocm* repository on ROCm/HIP and amdgpu kernel behavior.

Publications & talks: Eight peer-reviewed papers (2012–2015, five as first author) on wireless sensor networks and structural health monitoring, in IEEE MIXDES, the *International Journal of Distributed Sensor Networks*, the *International Journal of Electronics and Telecommunications* and *Microelectronics Reliability*. Two articles in *Paged Out!* Issue #9 (Jul 2026, ISSN 3043-0534): “Your Car Is Watching You: How Camera Pixels Cross Meters of Cable Into Linux” (p. 21) and “Edge AI on old mining silicon: LLMs and Diffusion on 16 GB AMD APU” (p. 35). Co-founder (2017) and organizer of CEHUG Łódź, the local C/C++, embedded and hardware user group; talk there, June 2026: “Your Car Is Watching You: In-Cabin Cameras from Sensor to V4L2 Driver” ([recording](#)).

Training: GMSL University, Analog Devices (2026); HARMAN AX AI Acceleration Program; Linux Kernel Debugging (Sages); AUTOSAR 4 in Practice (Vector Academy); Altium Designer.

Languages: Polish (native), English (fluent), German (basic).