# Roman LEONOV




## Embedded Software Entwickler

Staatsangehörigkeit: russisch
Sprache: English (C1), Deutsch (B2), Russisch (Muttersprache)
Aufenthaltserlaubnis, 30 ABS. 1 S. 1 NR. 3G VAR. 1, gültig bis 15.03.2027
Berlin, Germany · leonovro@mail.de · +49 176 26006147
LinkedIn · GitHub


## KURZPROFIL

Embedded-Software-Ingenieur mit über 13 Jahren Erfahrung in C, RTOS und Bare-Metal-Entwicklung. Fundierte Expertise in hardwarenaher Softwareentwicklung, ergänzt durch einen Hintergrund in Schaltungsentwicklung und PCB-Layout. Architekturverantwortung und erfolgreiche Markteinführung von 11 Flaggschiff-Geräten mit insgesamt mehr als 1 Mio. produzierten Einheiten.

Aufbau und Leitung eines Embedded-Teams mit 5 Ingenieuren, davon 3 direkt unterstellt, sowie Weiterentwicklung des Bereichs von einer Einzelverantwortung zu einer strukturierten Entwicklungsorganisation. Umfassende Expertise in USB Host/Device, Kryptografie sowie Hardware-Software-Debugging mit JTAG/SWD, Protokollanalysatoren und Oszilloskopen.

In den Jahren 2024–2025 führender Contributor zur ESP-USB-Komponente im ESP32 IoT Development Framework von Espressif; zusätzlich Open-Source-Beiträge zum USB-Host-Stack von Zephyr RTOS.

## KENNTNISSE

Embedded Software & RTOS: Embedded C, Zephyr RTOS, FreeRTOS
CPU-Architekturen & Mikrocontroller-Plattformen: ARM Cortex-M-Serie, Xtensa, RISC-V
Schnittstellen & Protokolle: USB, UART, SPI, I²C, SDMMC, 1-Wire, CAN, NFC
Security & Kryptografie: PKI, ECC, RSA, SHA, AES, DES
Hardware & Debugging: Schaltungsentwicklung (KiCad), Hardware-Debugging und Board-Bring-up, USB-Protokollanalyse
Tools & Automatisierung: GCC, Git, GitHub Workflows/Actions, Python, CI/CD-Pipelines

## BERUFSERFAHRUNG

**Senior Software Engineer · Espressif Incorporated · Berlin, Deutschland (Remote) · 10.2022 – 11.2025**
Espressif entwickelt ESP32-Mikrocontroller und Systems-on-Chip (SoCs) mit Wi-Fi-/Bluetooth-Konnektivität sowie Software für vernetzte Geräte und IoT-Anwendungen.

Stack: Embedded C, ESP-IDF, FreeRTOS, TinyUSB, DWC2, USB-Protokollanalysatoren, Python, GitHub Actions/CI

Entwickelte das USB-Subsystem von ESP-IDF vom Silicon Bring-up bis zur Integration produktionsreifer Software und war für Stack-Architektur, öffentliche APIs sowie USB-Host-

und USB-Device-Funktionalität verantwortlich. Übernahm zudem die Verantwortung für Espressifs TinyUSB-Integration, einschließlich plattformspezifischer Anpassungen, Upstream-Synchronisierung, CI und Release-Pflege.

- Entwarf und implementierte die erste Unterstützung externer USB-Hubs im ESP-IDF USB Host Stack unter Beibehaltung der bestehenden Client-API; entwickelte den Hub-Treiber und die Port-State-Machine und ergänzte 10 dedizierte Testfälle. Die Funktionalität wurde ohne stack-spezifische Anpassungen in einer öffentlichen Demo auf der Embedded World 2025 eingesetzt, in der zwei UVC-Kameras über einen USB-Hub betrieben wurden.
- Refaktorierte die bestehende USB-Geräte-Enumeration in einen dedizierten, Queue-basierten State-Machine-Treiber und schuf damit die architektonische Grundlage für Hub-Topologien und die Verarbeitung mehrerer gleichzeitig erkannter Geräte. Ergänzte 37 Testfälle, darunter Failure-Path-Szenarien mit einem rekonfigurierbaren TinyUSB-Mock-Device zur gezielten Fehlerinjektion in einzelnen Phasen der Enumeration.
- Implementierte Teardown und Reinitialisierung für den ESP-IDF TinyUSB Device Stack und ermöglichte damit die automatisierte Rekonfiguration des DUT ohne Power-Cycle. Ergänzte Espressif-spezifische Hardware-in-the-Loop-CI für ESP-IDF v5.1 bis v6.0 sowie ESP32-S2- und ESP32-P4-Targets mit 23 automatisierten Hardware-Testläufen für Device-Beispiele und Integrationstest-Anwendungen.
- Etablierte und betreute 6 milestone-basierte esp_tinyusb-Release-Zyklen und pflegte Espressifs TinyUSB-Fork über mehrere Upstream-Upgrades hinweg. Migrierte plattformspezifische Patches zwischen Release-Branches, löste Konflikte in DWC2, APIs und CI und nutzte eine stufenweise Synchronisierung, um validierte Upstream-Fixes zu integrieren, ohne unfertige Änderungen zu übernehmen.
- Implementierte Cache-Unterstützung für ESP32-P4 in TinyUSB und brachte die Änderung upstream in das Hauptprojekt von TinyUSB ein, wodurch die Plattformunterstützung direkt in den Upstream-USB-Stack aufgenommen wurde.

**Karrierepause · 05.2022 – 08.2022**
Umzug nach Deutschland.

**Aladdin R.D. · Moskau, Russland · 12.2014 – 04.2022**
Aladdin R.D. ist ein russisches Cybersecurity-Unternehmen, das Lösungen für Authentifizierung und elektronische Signaturen entwickelt, darunter die Produktlinie JaCarta. Gemessen am Umsatz 2024 belegte das Unternehmen Platz 5 im TAdviser Security 100.

Karriereentwicklung: Embedded Software Engineer → Embedded Software Team Lead

Leitete die technische Weiterentwicklung der Embedded-Plattform JaCarta 3 von einer ersten R&D-Implementierung hin zu einer wiederverwendbaren Produktionsarchitektur, die in einer Produktfamilie aus USB-Sicherheitstokens und Smartcard-Lesegeräten eingesetzt wurde. Verantwortlich für Firmware-Architektur, Plattformentwicklung, Provisionierungs- und Konfigurationsinfrastruktur sowie Produktionsprogrammierungsprozesse für Fertigungsserien von bis zu 500.000 Geräten.

**Embedded Software Team Lead · 07.2017 – 04.2022**
Stack: ARM Cortex-M, USB (CCID/HID/MSC), PKI (ECC/RSA/SHA/AES), JavaCard, Secure Element, NFC, JTAG/SWD, GRUB2

- Behielt die technische Gesamtverantwortung für die JaCarta-3-Architektur und leitete die kontinuierliche Weiterentwicklung der gemeinsamen Embedded-Plattform,

einschließlich Architektur- und Code-Reviews, des Boot-/Update-Subsystems sowie der Migration bestehender Legacy-Funktionalität in die gemeinsame Codebasis.
- Skalierte die Embedded-Entwicklung von einer Einzelverantwortung zu einem Team: stellte 5 Ingenieure ein, betreute und entwickelte sie fachlich weiter, verteilte Verantwortungsbereiche auf verschiedene technische Themen und führte weiterhin 3 Entwickler direkt.
- Entwarf und implementierte einen x86-Emulator für JaCarta 3, mit dem derselbe C-Applikationscode unverändert sowohl auf dem Emulator als auch auf ARM-Cortex-Targets ausgeführt werden konnte. Dieselbe Testsuite wurde vom Embedded-Team, von QA und angrenzenden Teams ohne physische Hardware genutzt. Ermöglichte die Entwicklung und Validierung von Plattformanwendungen auf der Emulator-/Tooling-Infrastruktur, darunter die 3 wichtigsten produktiven PKI-/RSA-Anwendungen, und entkoppelte damit die Applikationsentwicklung von MCU- und Hardware-Integration.
- Entwarf die Detailarchitektur einer Pre-Boot-Authentifizierungslösung für ein USB-Sicherheitsgerät, einschließlich MSC-Speicher- und LUN-Layout, Authentifizierungsprotokoll vor dem Laden des Betriebssystems, dynamischer Freigabe von Volumes abhängig von Autorisierungsrollen sowie eines plattformübergreifenden GRUB2-Boot-Flows. Die Lösung wurde in einer Produktionsserie von 500.000 Geräten ausgeliefert.
- Leitete die Entwicklung von Smartcard-Lesegeräten über Firmware und Hardware hinweg: entwarf einen biometrischen Authentifizierungsablauf auf Basis der FingerCell-Bibliothek mit JavaCard/Secure Element als sicherem Speicher und führte Bring-up sowie RF-Tuning der 13.56 MHz NFC-Antenne für ein kontaktloses Lesegerät durch, einschließlich Smith-Chart-Messungen und Optimierung des Matching-Netzwerks.

**Embedded Software Engineer · 12.2014 – 06.2017**
Stack: C, ARM Cortex-M, Nordic nRF51/nRF52, USB (CCID/HID/MSC), Flash/EEPROM, FatFS, RexOS, JavaCard, ISO7816, ISO14443A/B, NFC

- Konzipierte die Embedded-Plattform JaCarta 3 und ersetzte produktspezifische monolithische Firmware durch eine modulare Architektur mit einer wiederverwendbaren Plattformschicht und produktspezifischen Anwendungen. Die Plattform wurde später in 11 serienmäßig produzierten Sicherheitsgeräten eingesetzt.
- Entwarf eine fehlertolerante Dual-Slot-Architektur für Firmware-Updates: Neue Images wurden in den inaktiven Flash-Slot geschrieben, kryptografisch validiert und erst nach einem finalen Commit bootfähig gemacht, sodass die vorherige Firmware bei Stromausfall oder unterbrochenen Updates funktionsfähig blieb.
- Erhöhte die Sicherheit des Firmware-Update-Prozesses durch den Ersatz CRC-basierter Integritätsprüfungen durch einen schlüsselbasierten kryptografischen 32-Bit-MAC sowie durch verschlüsselte Firmware-Pakete, Versions- und Downgrade-Prüfungen und einen sicheren Update-Flow über CCID/APDU.
- Überarbeitete den persistenten Speicher für Embedded-Anwendungsfälle, indem FAT-orientierte Semantik durch eine EEPROM-ähnliche Abstraktion mit 32-Bit-Objekt-IDs und 128-Byte-Sektoren ersetzt wurde. Dadurch wurden unnötige Flash-Erase-/Rewrite-Vorgänge reduziert und eine transparente Verschlüsselung gespeicherter Daten ermöglicht.
- Entwickelte 6 produktspezifische Anwendungen auf Basis der JaCarta-3-Plattform, die alle in Serienprodukten eingesetzt wurden, darunter Funktionen für OTP, U2F und sicheren Datenspeicher.
- Implementierte Composite-USB-Sicherheitsgeräte mit CCID, HID und Mass Storage, einschließlich Unterstützung für bis zu 4 LUNs sowie dynamischer Umschaltung von

Wechseldatenträgern nach erfolgreicher Authentifizierung durch Remapping von SD-Kartenbereichen.
- Portierte das Open-Source-Betriebssystem RexOS auf Nordic nRF51/nRF52 und implementierte MCU-/Plattformunterstützung für UART, RTC, WDT, RNG, ADC, Flash, Timer, Power Management und RF; mehrere Pull Requests wurden upstream übernommen.

**Firmware Engineer · WWPass Corporation · Moskau, Russland · 02.2012 – 11.2014 (Teilzeit 02.2012 – 02.2013, anschließend Vollzeit)**
WWPass ist ein Cybersecurity-Unternehmen, das Hardware- und Softwarelösungen für Authentifizierung entwickelt, darunter USB-/NFC-Sicherheitstokens und Smartcards. Das Unternehmen beschäftigt 151–200 Mitarbeiter und hat seinen Hauptsitz in Nashua, New Hampshire, USA.

Stack: C, ATmega32U2, USB, NFC (13.56 MHz), KiCad, SolidWorks

Entwickelte Firmware und Hardware für PassKey und SecurityKey. Dabei handelte es sich um USB-/NFC-Hardware-Sicherheitstokens für kryptografische Authentifizierung, die Embedded-Firmware, sichere Speicherung von Zugangsdaten sowie NFC-/USB-Kommunikation kombinierten. Entwickelte außerdem Firmware und Hardware für NFC-/USB-Smartcard-Lesegeräte, konstruierte das Gehäuse in SolidWorks und fertigte Prototypen mittels CNC-Bearbeitung.

- Entwickelte die Hardware für ATmega32U2-basierte USB-/NFC-Authentifizierungstokens, einschließlich Schaltplan, PCB-Layout und Anpassung der 13.56 MHz NFC-Antenne mithilfe eines Antennenanalysators und Smith-Diagramms. Die Designs wurden in Produktionsserien von ca. 10.000 PassKey- und 15.000 SecurityKey-Geräten eingesetzt.
- Verbesserte die Zuverlässigkeit der USB-Enumeration auf dem ATmega32U2 durch eine Watchdog-basierte Recovery-Logik für einen USB-Stack eines Drittanbieters. Dadurch wurden sporadische Hänger während eines Compliance-Tests mit 150 Zyklen beseitigt und der vollständige Testablauf konnte erfolgreich abgeschlossen werden.

**Firmware Engineer · Bauman Moscow State Technical University (NIIRET BMSTU) · Moskau, Russland · 09.2011 – 02.2013**
Forschungseinheit innerhalb des Fachbereichs RL1 der Bauman Moscow State Technical University mit Schwerpunkt auf ingenieurwissenschaftlichen und angewandten Forschungsprojekten. Studierende werden dabei in Forschungsarbeiten, Studienprojekte und Abschlussarbeiten eingebunden. Die Einheit umfasst etwa 10–15 wissenschaftliche Mitarbeiter.

Stack: C (Bare-Metal-Entwicklung), STM32F100, STM32F200, CC1101, UART

- Entwickelte einen verteilten Zeitsynchronisationsmechanismus für ein Netzwerk aus 200 CC1101-basierten Wearable-Geräten mit STM32F200-Mikrocontrollern. Das System nutzte Zeitschlitzzuweisung und die Weitergabe einer Referenzzeit von einem Basis-Knoten; erreicht wurden eine netzwerkweite Synchronisation innerhalb von ca. 5–10 Sekunden sowie ein zuverlässiger Betrieb auf einem etwa 2 Hektar großen Waldgelände während einer dreitägigen Veranstaltung.

- Entwickelte und realisierte eine STM32F100-basierte Antennenpositionierungssteuerung für den Remote-Laborbetrieb, einschließlich Schaltungs- und PCB-Design, Bare-Metal-Firmware und eines eigenen UART-Protokolls zur PC-gesteuerten Einstellung von Azimut und Elevation. Integrierte die Steuerung in die Host-Software und führte das System bis zu einem funktionsfähigen Prototypen.

## AUSBILDUNG

**Ingenieurabschluss · Bauman Moscow State Technical University · 09.2007 – 07.2013**

anabin-Status: H+

WES-Äquivalenzbewertung: Masterabschluss im Bereich Entwicklung von Funk- und elektronischen Systemen.

## EHRENAMTLICHES ENGAGEMENT

**The Zephyr Project · USB Area Collaborator · 06.2026 – heute**

Mitarbeit im USB-Bereich des Zephyr Project, einem kleinen, skalierbaren Open-Source-RTOS für eingebettete IoT-Geräte.

## SONSTIGES

- Deutscher Führerschein, Klasse B