



Embedded Systems Consulting

Service Catalogue

Firmware Division

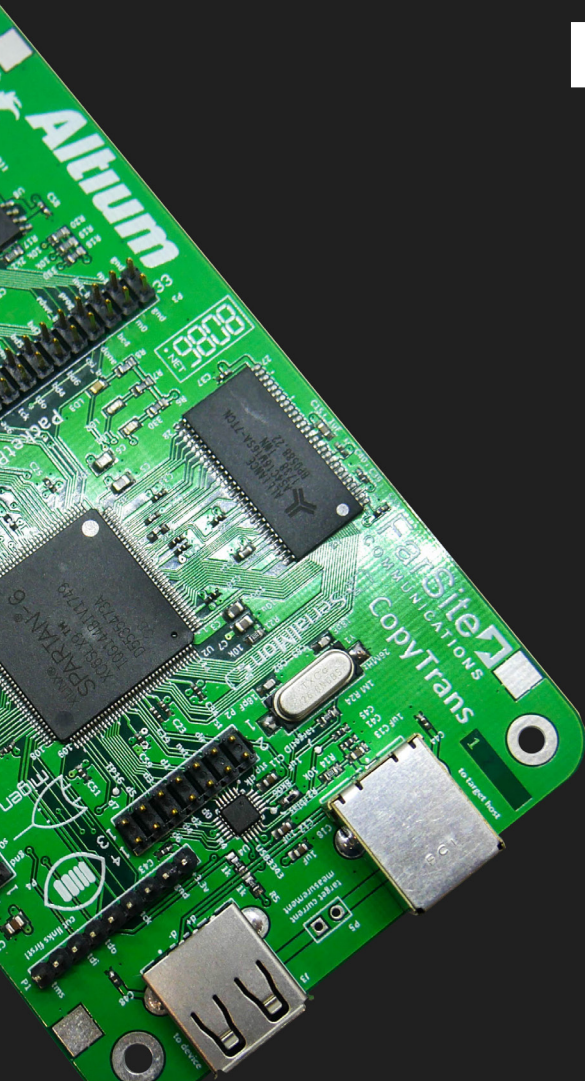
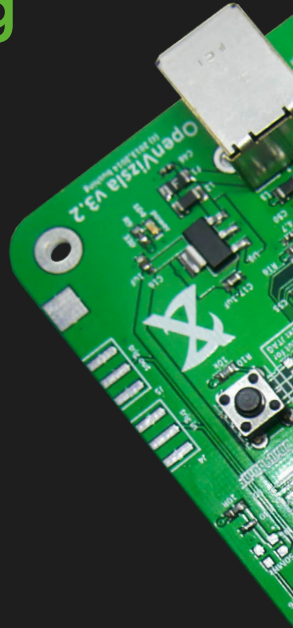
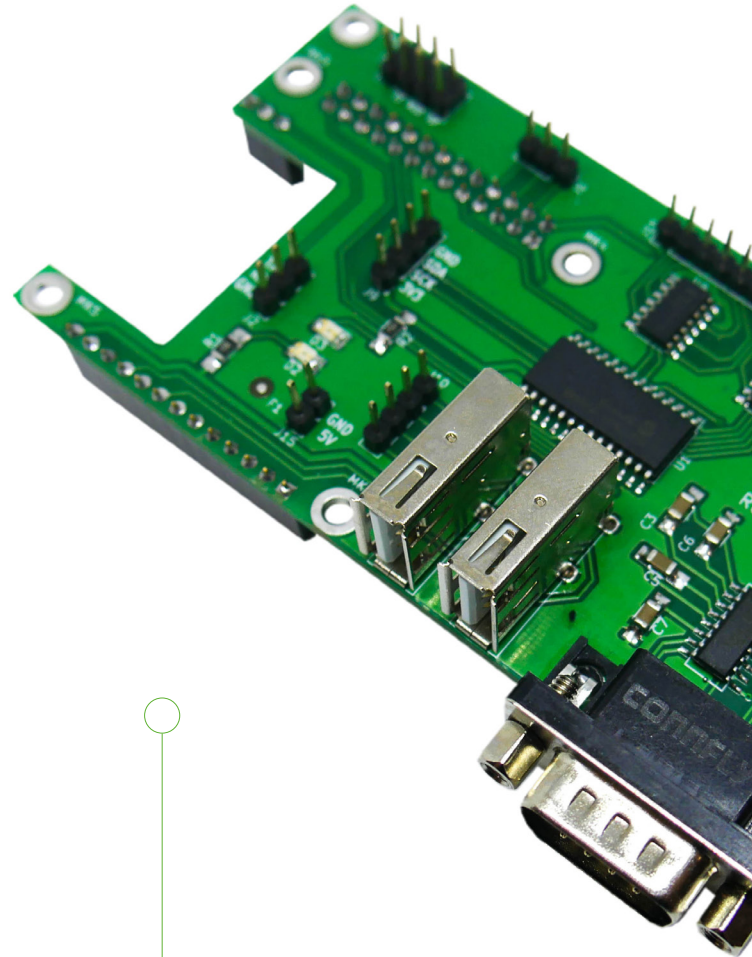
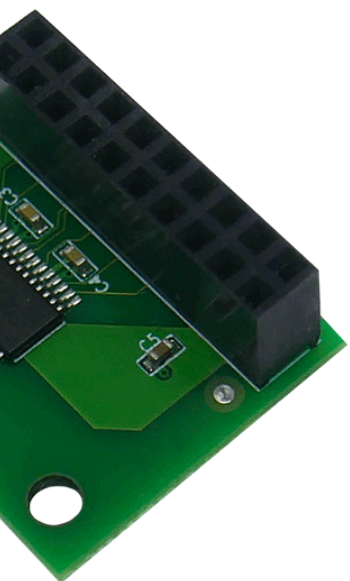


Table of contents

1. About Us
2. Our areas of expertise
3. Industries we have experience with
4. Programming languages
5. Firmware frameworks, bootloaders and hypervisors
6. Tools and libraries
7. Open-source project we contribute to
8. Hardware experience
9. Reference Projects
10. Contact



About Us

We're hands-on firmware experts with years of experience writing elegant, scalable and custom code for clients. Our team of engineers brings broad and expansive Linux knowledge to the projects we take on. We are experts at firmware development, OS development and application development, with a special focus on Embedded Linux solutions.



Our Clients Map



Our Partners

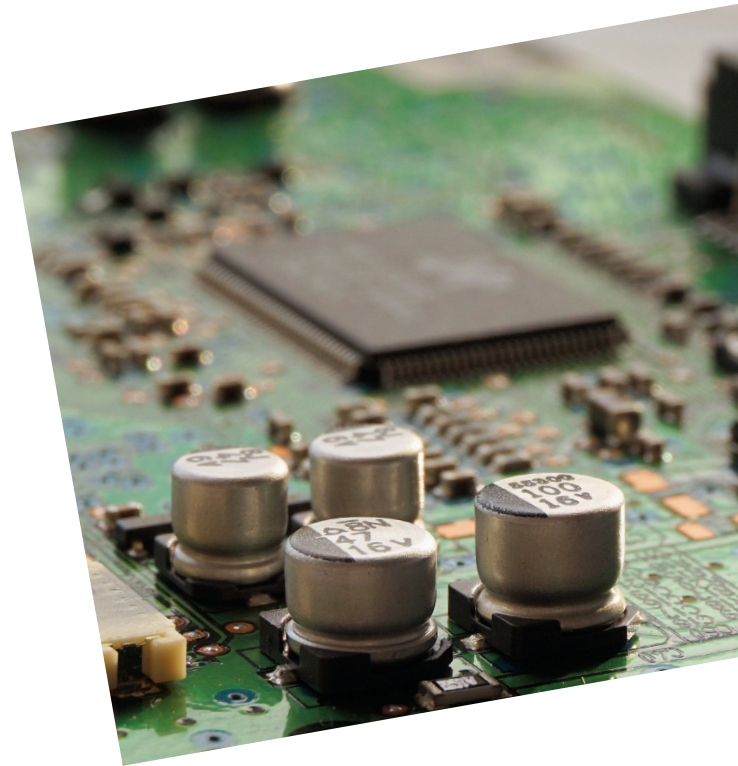


PC Engines™



Our areas of expertise

- coreboot:
 - ▷ firmware development customization
 - ▷ payload development (libpayload, SeaBIOS and more)
 - ▷ mainboard porting mainboard maintenance
- UEFI firmware development
 - ▷ UEFI applications
 - ▷ UEFI payload for coreboot
 - ▷ UEFI customization
- S-RTM/D-RTM solutions
- hypervisor development (Xen, Bareflank, ACRN)
- bootloader development (GRUB2)
- Firmware consulting
- Continuous Integration and Continuous Delivery for firmware
- Boot time optimization
- Firmware flashing
- Hardware/firmware security features enablement
- Firmware security analysis



Industries we have experience with

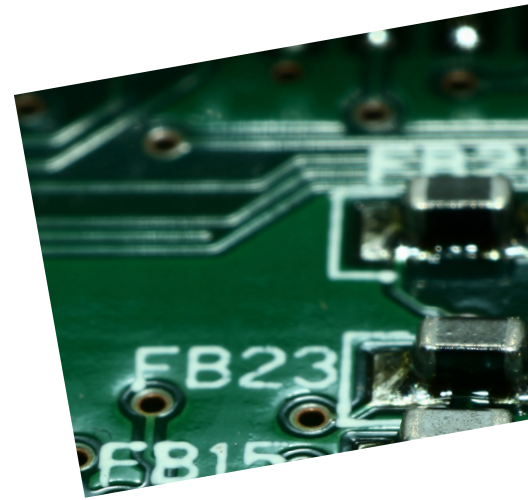
- Network appliance
- Edge Computing
- Consumer electronics
- Digital signage
- Medical
- Robotics (consumer and military applications)

Programming languages

- ▶ C
- ▶ C++
- ▶ x86 assembly
- ▶ Python
- ▶ Go
- ▶ Shell script



- ▶ Programming languages
- ▶ Bootloaders validation



Bootloaders validation

- ▶ coreboot
- ▶ tianocore/EDKII
- ▶ heads
- ▶ Xen
- ▶ Bareflank
- ▶ ACRN
- ▶ GRUB2
- ▶ SeaBIOS
- ▶ Genode
- ▶ iPXE
- ▶ LinuxBoot



Tools and libraries

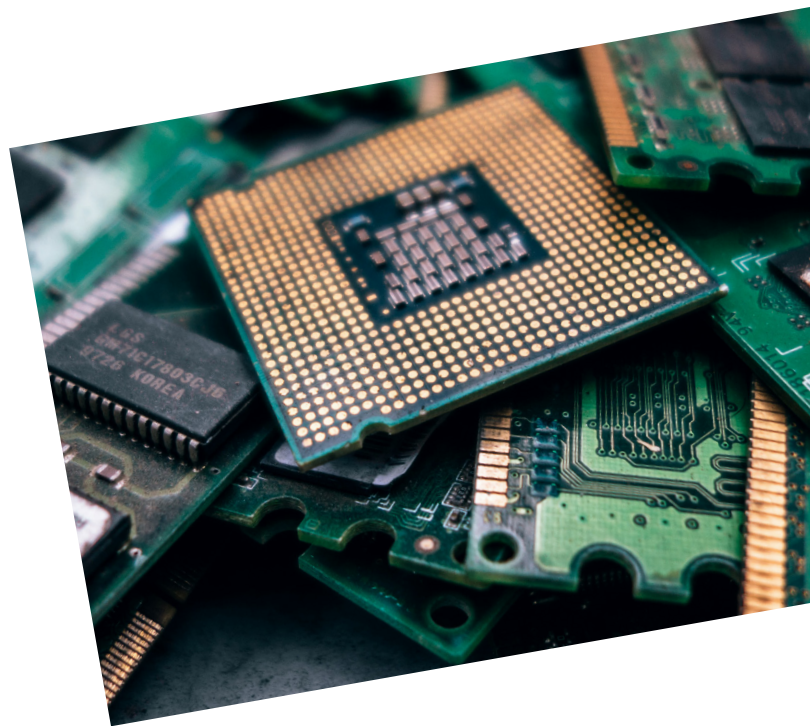
- GNU toolchains
- SCM:
 - ▷ Git
 - ▷ gerrit
- GNU/Linux build systems:
 - ▷ make
 - ▷ cmake
 - ▷ custom
- Firmware flashing:
 - ▷ flashrom
 - ▷ DediProg software
 - ▷ Linux SPI on OrangePi/RaspberryPi
- Payloads:
 - ▷ tianocore UEFI payload
 - ▷ coreboot libpayload
 - ▷ Bareflank SDK
 - ▷ BITS
 - ▷ Python
- Libraries:
 - ▷ gnu-efi

► Continuous Integration / Delivery:

- ▷ Jenkins
- ▷ Travis CI
- ▷ Gitlab CI
- ▷ Circle CI

► Security:

- ▷ TPM2 tools
- ▷ tpm tools
- ▷ LUV
- ▷ chipsec

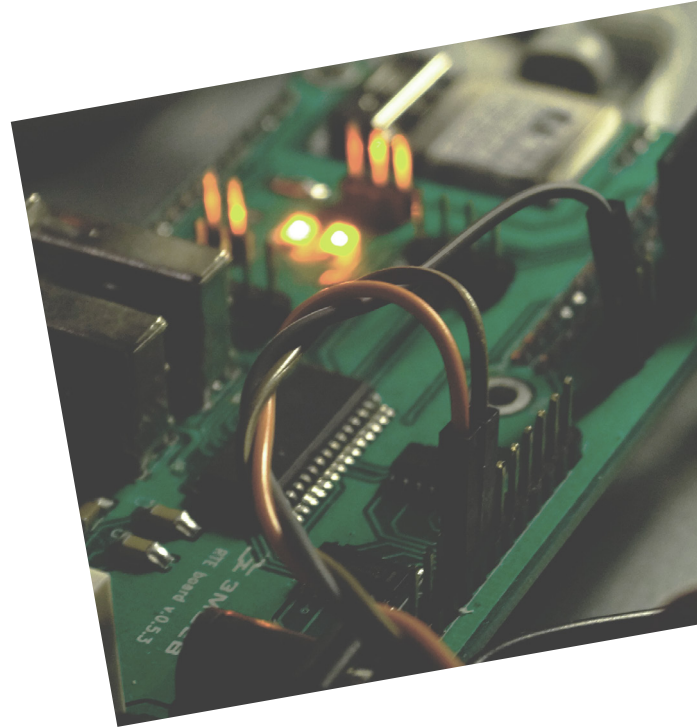


Open-source projects we contribute to

- ▶ [coreboot](#)
- ▶ [SeaBIOS](#)
- ▶ [GRUB2](#)
- ▶ [TrenchBoot Landing Zone](#)
- ▶ [memtest86+](#)



- ▶ Open-source projects we contribute to
- ▶ Hardware Experience



Hardware Experience

Architectures

- ▶ ARMv6, ARMv7, ARMv8 (Arm 64-bit)
- ▶ x86, x86_64
- ▶ FPGA/CPLD

SoC/CPU

- ▶ AMD embedded SoCs (GX412-TC, G-T40N, G-T40E)
- ▶ Intel Celeron J3060, J3160, 3865U
- ▶ Intel Core i3-7100U, i5-7200U, i7-6500U
- ▶ Intel Atom E3826, E3845, x5-E3940

Reference validation projects

coreboot firmware maintainership for PC Engines

PC Engines company produces routers based on AMD processors called APU. 3mdeb responsibilities are:

- ▶ monthly coreboot firmware updates with rebases to official repository
- ▶ new features development and bug fixing support
- ▶ upstream changes and fixes to official coreboot repository
- ▶ maintain code repositories and release page with binaries and release notes
- ▶ CI/CD for firmware building and testing (with Validation Team cooperation)

The cooperation with PC Engines lasts 4 years and still goes on.

Programming languages used:
C, x86 assembly, Shell script

Hardware used:
AMD GX412-TC, AMD G-T40E. AMD G-T40N

Technologies used: **Linux, make, GCC, Gitlab CI**

▶ Reference validation projects

Network appliance device porting to coreboot

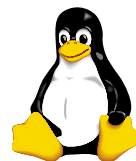
Customer had few network appliance products based on Intel processors. They wanted its firmware to be open-source and available in coreboot. 3mdeb responsibilities:

- ▶ develop the support for mainboard in coreboot
- ▶ upstream the mainboard support to official coreboot repository

Programming languages used:
C, x86 assembly, Shell script

Hardware used: **Intel Celeron J3060, J3160, 3865U; Intel Core i3-7100U, i5-7200U**

Technologies used:
coreboot, Linux, make, GCC



Chromebooks lock PIN for retail distributors

Customer had an idea of putting a PIN code verification in the firmware in order to prevent stealing the devices from the shop. 3mdeb responsibilities:

- analyze the possibility to modify original firmware code and boot path
- develop a prompt for PIN code displayed at the LCD panel of the Chromebook device
- prepare a possibility to provision and lock the device with PIN code using the verified boot mode of Chrome devices

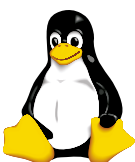
Programming languages used:

C, Shell script

Hardware used:

Intel Celeron J3060, J3160, 3865U; Intel Core i3-7100U, i5-7200U

Technologies used: **coreboot, Linux, make, GCC, vbboot**



UEFI loader for proprietary UEFI application

Customer had a proprietary UEFI application for Ethernet controller and wanted to implement a loader which would launch the application with a predefined set of parameters in EFI Shell-like manner and then proceed with booting to Shell/OS. 3mdeb responsibilities:

- develop an UEFI Loader application
- analyze the proprietary application in terms of required protocols
- fit the UEFI Loader into the platform's UEFI firmware image so that the loader would be launched automatically at startup

Programming languages used:

C, Shell script

Hardware used:

Intel Atom E3826, E3845

Technologies used:

UEFI, Linux, make, GCC, gnu-efi

 Firmware validation payload

Customer had a very old coreboot firmware used on a medical robot. This firmware was undergoing a certification. In order to assure that the firmware is compliant with necessary standards and specification to obtain the certification, customer asked us to develop a validation and verification payload. 3mdeb responsibilities:

- ▶ replace customer's special payload application with validation payload using
- ▶ specified multiboot boot protocol
- ▶ develop a payload that will verify firmware initialization correctness
- ▶ develop a test suite in payload that would check ACPI tables, SMBIOS tables and MP table conformance to their specifications

 References:

- ▶ [OSFC 2018 - BITS and CHIPSEC as coreboot payloads](#)

Programming languages used:

C, Shell script, Python, assembly x86

Hardware used:

Intel Atom E3826, E3845

Technologies used:

coreboot, make, GCC, BITS, GRUB

Custom embedded hypervisor

Customer wanted to launch multiple applications on his platform, however the restriction was that every binary must be placed in SPI flash and no external storage is allowed. 3mdeb responsibilities:

- ▶ choose an appropriate hypervisor/hypervisor SDK that would fit requirements
- ▶ develop a embedded hypervisor from scratch using hypervisor SDK
- ▶ develop software-based hardware resources isolation due to lack of IOMMU
- ▶ embed the hypervisor as coreboot payload
- ▶ port libpayload to x86_64 in order to support the hypervisor requiring 64bit mode

References:

- ▶ [OSFC 2019 - Consideration about enabling hypervisor in open source firmware](#)

Programming languages used:

C, C++, Shell script, assembly x86

Hardware used:

Intel Atom E3826, E3845

Technologies used:

coreboot, make, GCC, Bareflank, libpayload

D-RTM implementation for AMD processors

3mdeb own initiative to participate in Trenchboot project. First open-source working implementation of AMD Secure Startup feature was developed. 3mdeb responsibilities:

- cooperation with landing-zone implementation (Secure Loader for AMD D-RTM)
- cooperation with GRUB2 support for Secure Startup
- cooperation with Linux development for D-RTM launch
- implementing kexec mechanism for Xen

References:

- [OSFC 2019 - TrenchBoot - Open DRTM implementation for AMD platforms](#)
- PSEC 2019 - Less-Insecure Network Edge Virtualization with Low Size, Weight and Power

Programming languages used:

C, Go, Shell script, assembly x86

Hardware used:

AMD GX-412TC

Technologies used:

coreboot, Linux, GRUB2, Xen, u-root

Firmware security analysis for modern high-end laptop

Customer wanted to evaluate the security of the firmware of Dell XPS laptop. 3mdeb responsibilities were:

- perform firmware security analysis with commonly available tools likechipsec, LUV
- check the firmware against various attacks
- analyze the BIOS setup
- analyze 3rd party components in terms of possible persistence of attacker data
- analyze 3rd party firmware (SSD, Thunderbolt, Realtek HDA)
- SMI fuzzing

Programming languages used:

C, Python, Shell script, assembly x86

Hardware used:

Dell XPS 9560, Dediprog SF100

Technologies used:

UEFI, Linux, Dediprog software, LUV, chipsec, BITS





Embedded Systems Consulting

Feel free to ask
if you have any questions.

Languages

English, Polish

BOOK A CALL

ul. Abrahama 1A / 2.05
80-307 Gdańsk, Poland

TAX ID: PL2530164147

Piotr Król

Founder & Embedded
Systems Consultant

+48 880 673 344

contact@3mdeb.com



Website



About Us