👋 DUG#8 Shameless Plug 👋

- 🌱 **Growth & Transparency:** Showcasing our evolution and commitment to open-source.
- 📚 **Historical Record:** A resource for 3mdeb, future customers, and the privacy/security community.
- 📈 **Explore Business Model:** Learn from our open-source firmware journey, including potential pitfalls.

What we will talk about

- Hardware,
- Services,
- Pace Enterprise Training,
- Dasharo Entry Subscription (aka Dasharo Pro/Enterprise Package)
- Merchandise,
- Everything available in 3mdeb shop: https://shop.3mdeb.com

Hardware

SDWire

89,00 €
(ex. VAT)

Overview

SDWire allows to flash SD card connected to the DUT (Device Under Test), without physical contact with the device. It serves as a SD card reader and SD card mux in one device. The PCB board is designed in such a way that it fits into micro SD card slots. Thanks to this, there is no need for special cables with a micro SD adapter, like in the muxPi product.

Documentation:

- Getting started
- Specification
- Usage
- FAQ Section

28 in stock

1 ADD TO CART

Enquiry for this product

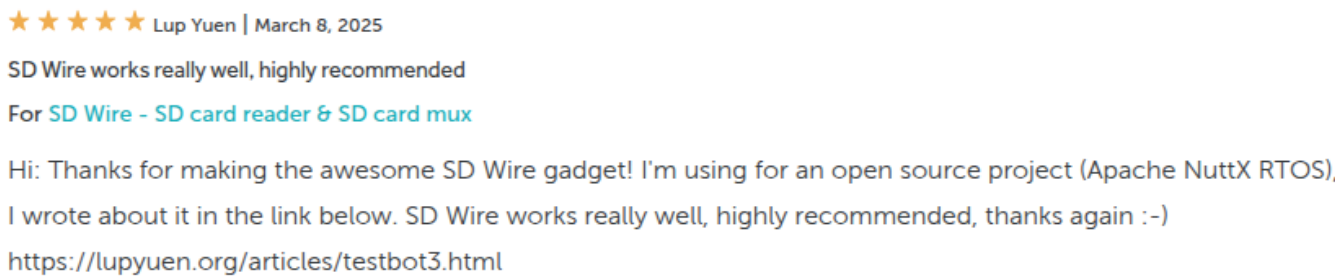
https://shop.3mdeb.com/shop/open-source-hardware/sdwire/

> Quote
>
> Developing firmware for an embedded device requires a lot of testing. (...) Some tests can be done on the host machine, but in most cases (especially related to specific peripherals), it is best to run tests on actual hardware.

> Quote
>
> Physical swap SD card / USB disk is the safest and easiest way for initial flash transfer from the development computer to the device being tested. (...) Initially, I contemplated a robot carrying a physical SD card like in the movie Hackers, but there should be an easier way... and it turns out there is! (...) They designed SD wrapper with no moving parts and called it SDWire.

https://www.kurokesu.com/main/2022/08/02/ethernet-camera-module-build-log-5-automated-flashing/

- SDWire as well as other products for DYI can be bought on Tindie.
- We appreciate very nice review and a blog post about usage of SDWire in Apache NuttX validation from Lup Yuen.


★★★★★ Lup Yuen | March 8, 2025

SD Wire works really well, highly recommended

For SD Wire - SD card reader & SD card mux

Hi: Thanks for making the awesome SD Wire gadget! I'm using for an open source project (Apache NuttX RTOS), I wrote about it in the link below. SD Wire works really well, highly recommended, thanks again :-)

https://lupyuen.org/articles/testbot3.html

https://www.tindie.com/products/3mdeb/sd-wire-sd-card-reader-sd-card-mux/

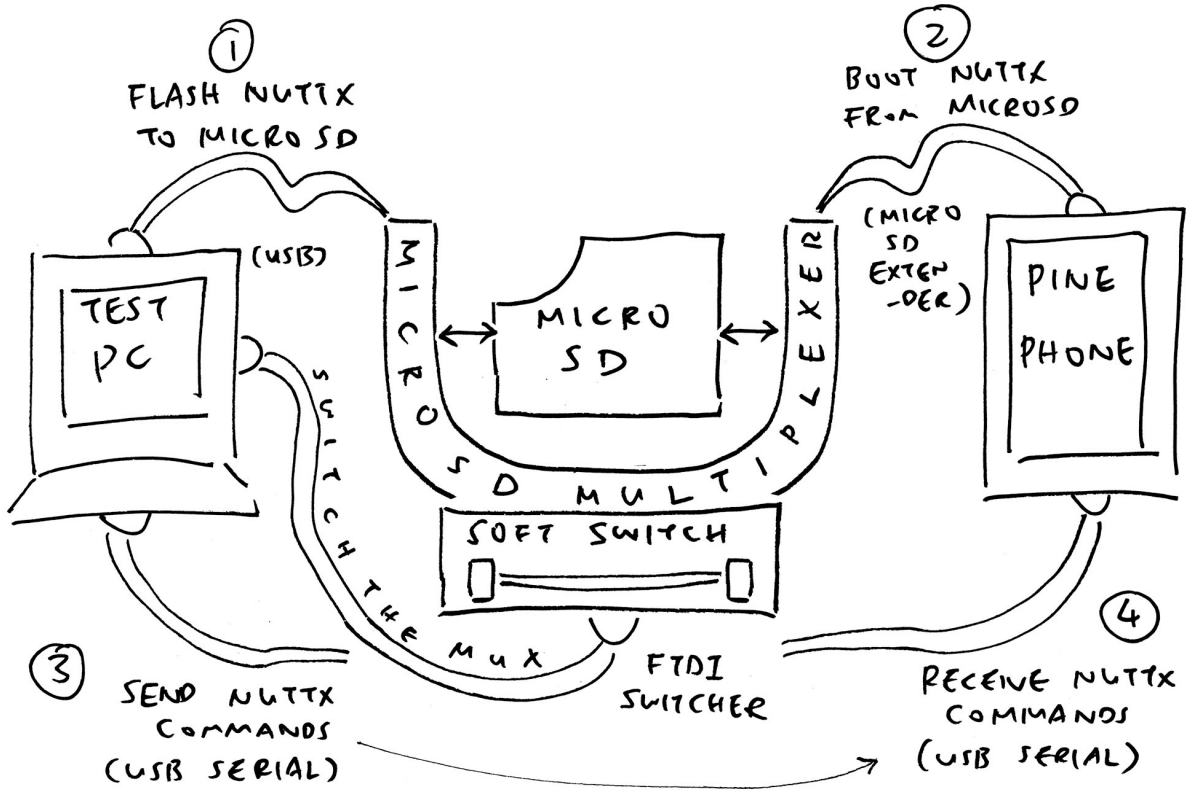
> Quote
>
> *We used Special Hardware: SDWire MicroSD Multiplexer (pic above)
>
> - Controlled by a Single-Board Computer: Yuzuki Avaota-A1 (Open Hardware) *PinePhone Test Bot kinda works!
> - Though PinePhone Battery complicates Hardware Testing *We might pivot to another Arm64 Single-Board Computer
> - Maybe we'll port NuttX to Allwinner A527 SoC

https://lupyuen.org/articles/testbot3.html

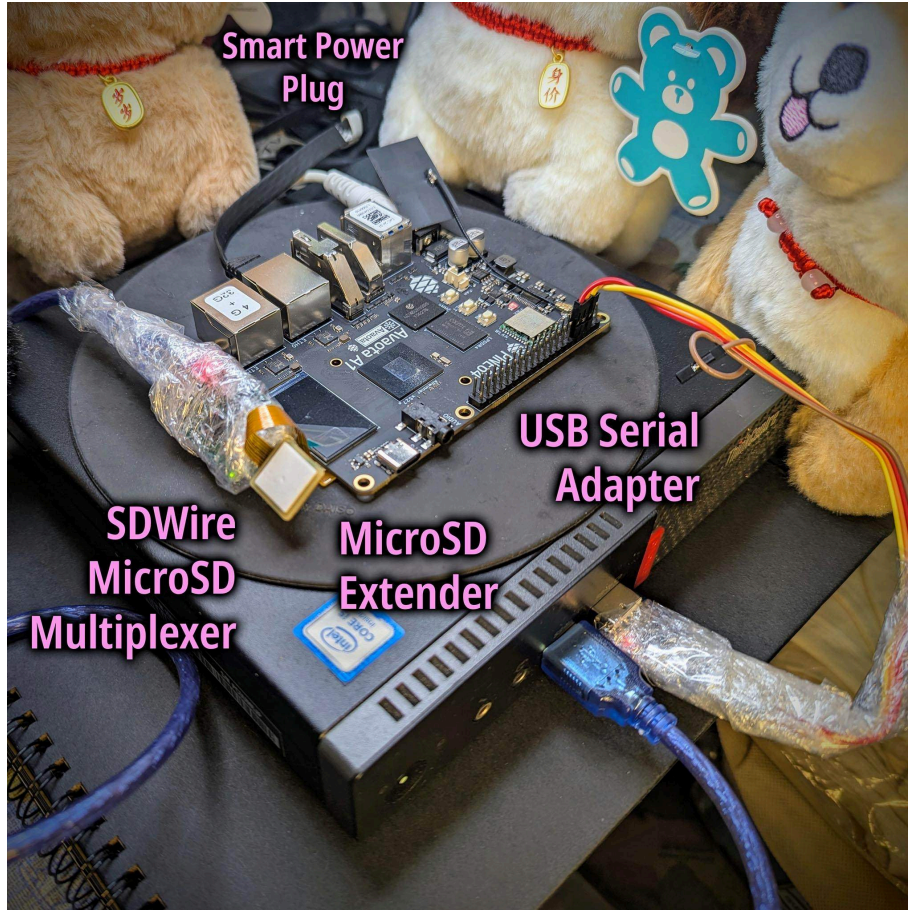
🚨 3mdeb Tindie Discount Code 🚨

5% for maximum 3pcs until end of March

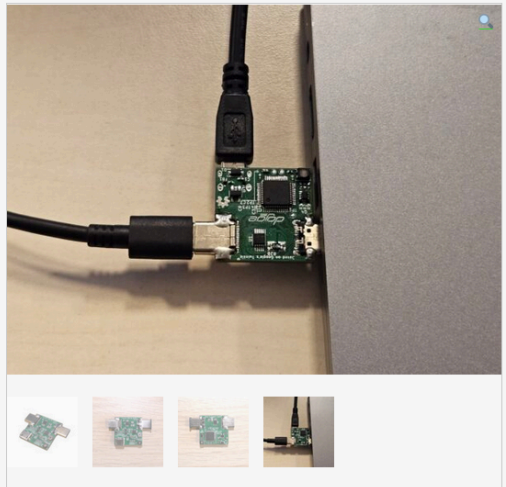
609D513D



https://lupyuen.org/articles/testbot3.html



https://lupyuen.org/articles/testbot3.html



Twonkie – USB-C sniffer

112,00 €
(ex. VAT)

Overview

An exclusive USB-PD sniffer/injector/sink inspired by the renowned Google project, Twinkie. It has been reimagined to be accessible to all enthusiasts. Twonkie is an excellent open-source solution for:

- Sniffing USB Power Delivery traffic on both Control Channel lines (CC1/CC2)
- Transparent interposer on a USB Type-C connection
- Monitoring VBUS and VCONN voltages and currents (Supports USB PD 3.1 EPR voltages up to 48V)
- Injecting PD packets on CC1 or CC2
- Putting Rd/Rp/Ra resistors on CC1 or CC2

The hardware design of Twonkie by Joachim "dojoe" Fenkes is licensed under CC BY 4.0.

Documentation:

- Twonkie Introduction at Dasharo Universe
- GitHub documentation

12 in stock

1 ADD TO CART

Enquiry for this product

https://shop.3mdeb.com/shop/open-source-hardware/twonkie-usb-c-sniffer/

Y **Hacker News** new | threads | past | comments | ask | show | jobs | submit

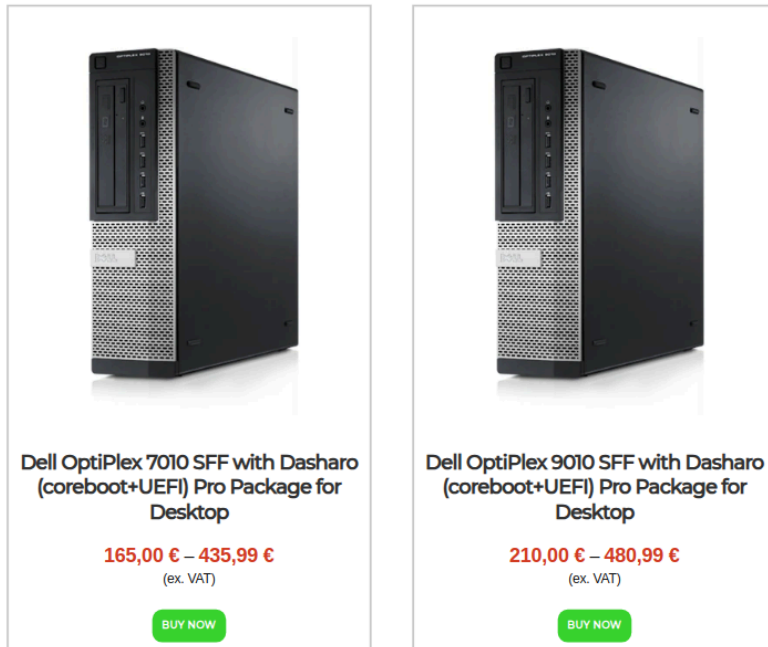
Twonkie: A USB-PD sniffer/injector/sink based on Google's Twinkie open hardware (github.com/dojoe)

168 points by transpute 8 months ago | hide | past | favorite | 26 comments

> **Quote**
>
> (...) they make sniffing the USB-PD messages considerably easier than using an amplifier and logic analyzer like I did here. If you're only interested in the protocol layer and above, these seem like excellent choices.

https://www.rbaron.net/blog/2024/06/02/usb-power-delivery-for-makers.html

Dasharo Supported Hardware



https://shop.3mdeb.com/product-category/dasharo-supported-hardware/

Dasharo Supported Hardware

https://shop.3mdeb.com/product-category/dasharo-supported-hardware/

NS70 Series 17.3" Dasharo laptop by NovaCustom

BUY NOW

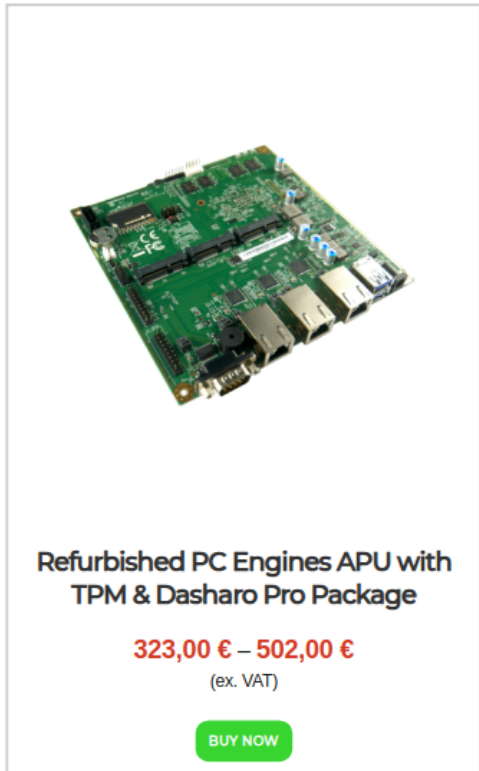
V54 Series 14" Dasharo laptop by NovaCustom

BUY NOW

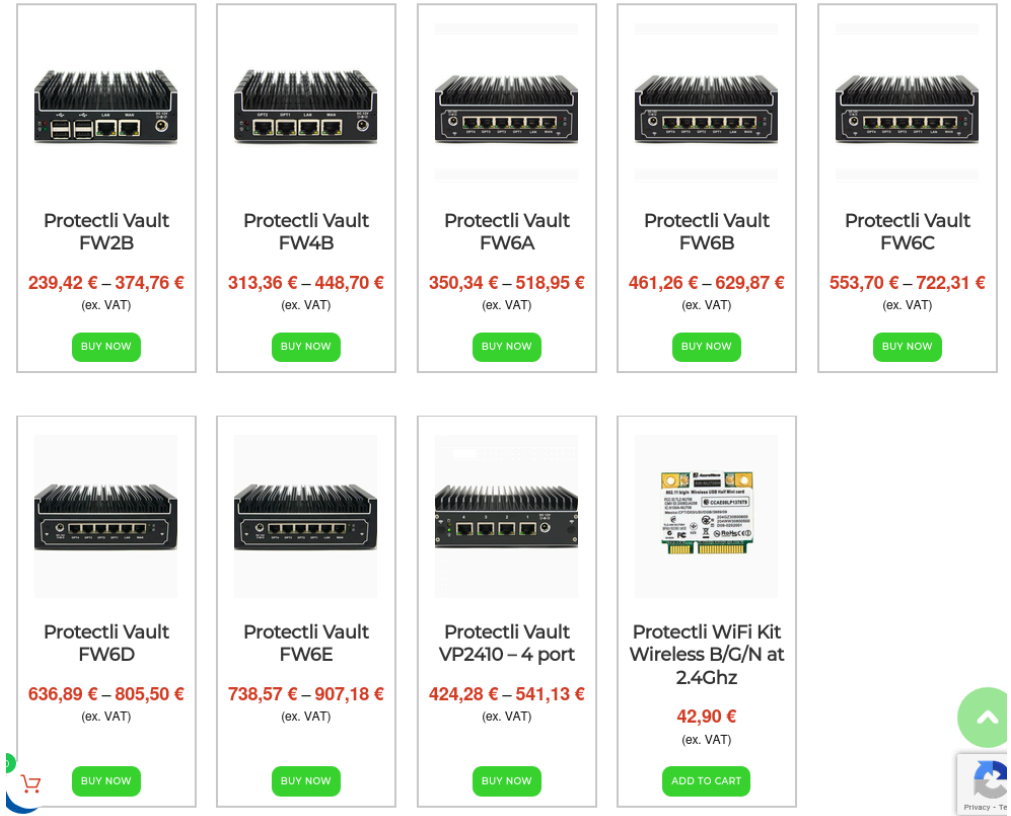
V56 Series 16" Dasharo laptop by NovaCustom

BUY NOW

https://shop.3mdeb.com/product-category/laptops/



https://shop.3mdeb.com/product-category/system-boards/



https://shop.3mdeb.com/product-category/vaults/

Pace Enterprise Training

Client Type	Individuals		Business	
Cooperation Models	OST2	Public Security Conferences	Remote PET / On-site PET	PET Corporate Program
Target Audience	Community	Individual Clients / Business Clients	Corporate, NGO and Public Sector	Corporate**
Primary Use	Self-paced Education	Individual Employees Education	Teams Education	Division Education
Access to Materials	Open	Limited to Students	Limited to Students*	Limited to Division
Support Level	Community Support and Occasional Support from Instructor	Direct Support from Instructor	Extended Direct Support from Instructor	Direct Support from Team of Instructors
Customization	-	Limited	High	High
Pricing Model	Free	Pre-Paid to Conference Organizers	Pre-Paid to 3mdeb	Annual Contract
Additional Features	-	Pace and Content Adjustment According to Audience Needs Access to Instructure during Breaks and Afterhours	Material Adjustment According to Organization Needs Analysis of Indivudal Needs of Organization	Custom Curricullum for Whole Division

* - Unless direct agreement available, including recording, speaker notes etc.

- x86, Arm and POWER9/10
- All open-source firmware topics for each available framework (coreboot, EDKII, U-Boot, TrustedFirmware, OpenBMC, Yocto etc.)
- Closed source firmware components: ME/CSME/TXE/SPS, PSP/ASP, microcode, Intel ACMs
  - based on publicly available materials
- Low-level security mechanisms with example CVE exploitation (UEFI Secure Boot, Intel Boot Guard, closed source firmware etc.)
  - vulnerability class analysis
- Trusted Computing Technologies
  - BitLocker/LUKS/Heads
  - Measured Boot
  - SRTM/DRTM and other Root of Trust for Measurement
- Firmware development life-cycle from considerations at hardware design stage to long term maintenance.
- Your topic not on the list? Feel free to contact us: `contact<at>dasharo<dot>com`

DS01CBI: coreboot for embedded linux developers

Overview

- modern x86 architecture
- firmware design principles by examples
- boot flow from power on to system take off
- Coreboot walk through
- firmware build process based on Coreboot
- coreboot developer workflow
- remote testing environment
- SPI flash theory of operation
- flashing and debugging tools
- writing payloads hands-on workshop
- FSP – from theory to integration
- firmware security basics in Coreboot ecosystem
- MinnowBoard – hands-on workshop using previously
- gained knowledge

Duration

5 days

40 hours (8h/day)

70% lectures

30% hands-on workshop

https://paceenterprisetraining.com/

DS02RTA: Intel Root of Trust training

Overview

- Based on OST2 Arch4001, Arch4021, TC3001, TC3011 and TC3211
- UEFI introduction
- Modern x86 architecture
- Where is firmware and why blobs
- Intel x86 feature set and boot process
- Intel Root of Trust Technologies
- Other Root of Trust technologies overview
- Intel Management Engines features, vPRO, me_cleaner
- Workshops using Intel Skylake-based COMe module showing the process of enabling Boot Guard and practical examples of its features

Duration

4 days

17 hours (8h/day)

100% lectures

Materials

All training materials presentations and source code will be available for the client's internal usage.

https://paceenterprisetraining.com/

DS03SSI: System Security training

Overview

- Based on OST2 Arch2001, Arch4001 and Arch4021
- x86 assembly
- x86 operating system internals
- x86 boot process
- PCI and PCI Express
- Modern Intel system architecture
- DMA and IOMMU
- ISA and Plug and Play
- Debugging with GDB and core dumps
- System emulation with QEMU
- UEFI introduction
- UEFI Secure Boot
- Introduction to Roots of Trust and Trusted Computing Technologies

Duration

9 days

37 hours (4h/day usually, except for once 5h/day)

of lectures with hands-on labs

Materials

All training materials presentations and source code will be available for the client's internal usage.

Language

English

https://paceenterprisetraining.com/

ZH01ELI: Building and Development of Embedded Linux Systems

Overview

- Open source development overview
- Brief history of Linux
- Linux kernel introduction
- Using Git for source code management
- Introduction to Embedded Linux
- Getting kernel source code
- Linux kernel configuration and compilation
- Cross development
- Linux kernel modules
- Character device drivers
- Linux kernel debugging
- Device Tree files
- Typical Embedded Linux bootloaders
- Introduction to Build Systems
- Building custom Embedded Linux system for typical hardware target
- Embedded Linux tools
- Embedded Linux application development and debugging

Duration

4 days

32 hours (8h/day)

50% lectures

50% hands-on workshop

Materials

All training materials presentations and source code will be available for customer internal usage.

https://paceenterprisetraining.com/

ZH02YPI: Yocto Project Development

Overview

- Overview of an Embedded Linux system architecture
- Overview of the Yocto Project and OpenEmbedded ecosystem
- Using Yocto Project documentation
- Building emulation image
- Building image for the development board
- Board Support Packages and Yocto Project metadata
- Customizing the build with layers
- Image customization
- Extending existing recipes
- Overview of some of the existing build systems (Autotools,
- CMake, Meson)
- Creating a custom recipe
- Creating a custom image

yocto PROJECT

Duration

4 days

28 hours (7h/day)

40% lectures

60% hands-on workshop

Materials

All training materials presentations and source code will be available for customer internal usage.

https://paceenterprisetraining.com/

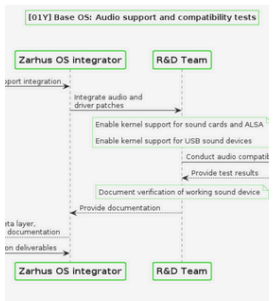
https://hardwear.io/usa-2025/training/mastering-uefi-secure-boot.php

https://shop.3mdeb.com

Backlog

Services

Zarhus Services

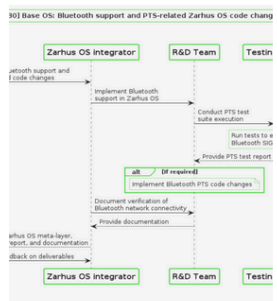


Zarhus OS: Base OS – Audio support and compatibility tests

4200,00 €

(ex. VAT)

ADD TO CART

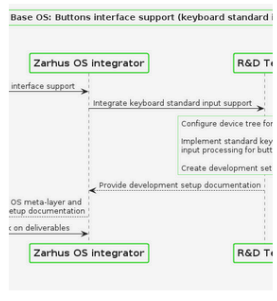


Zarhus OS: Base OS – Bluetooth support and PTS-related Zarhus OS code changes

5900,00 €

(ex. VAT)

ADD TO CART

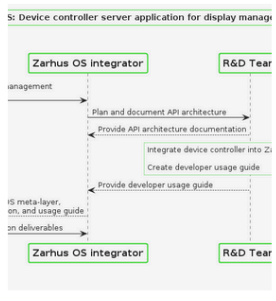


Zarhus OS: Base OS – Buttons interface support (keyboard standard input)

2450,00 €

(ex. VAT)

ADD TO CART

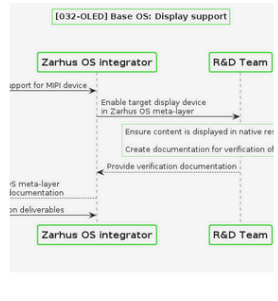


Zarhus OS: Base OS – Device controller server application for display management (HTTP)

6300,00 €

(ex. VAT)

ADD TO CART



Zarhus OS: Base OS – Display support (3x OLED mono/color)

2450,00 €

(ex. VAT)

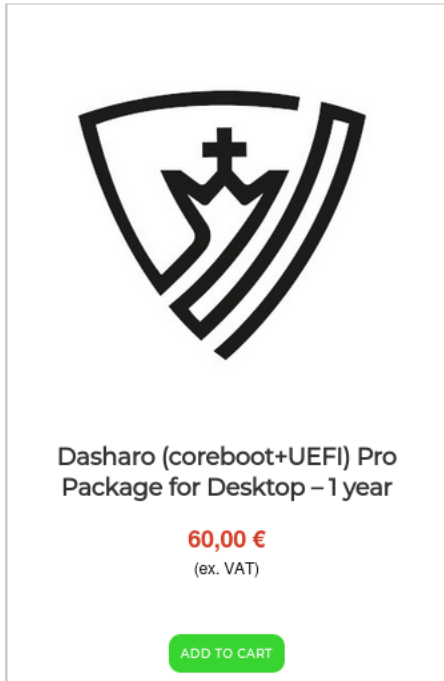
ADD TO CART

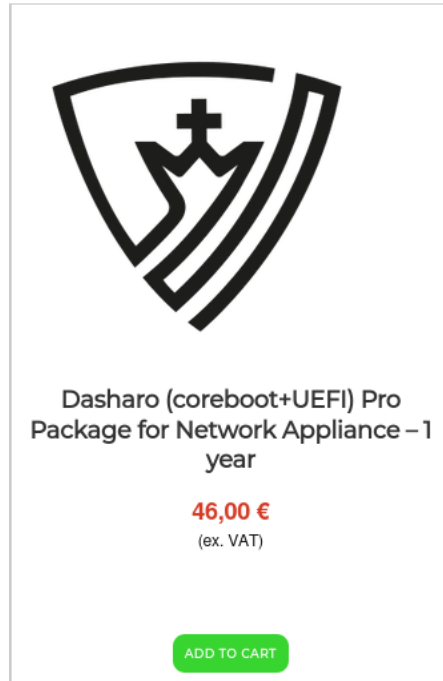
https://shop.3mdeb.com/shop/zarhus-services/

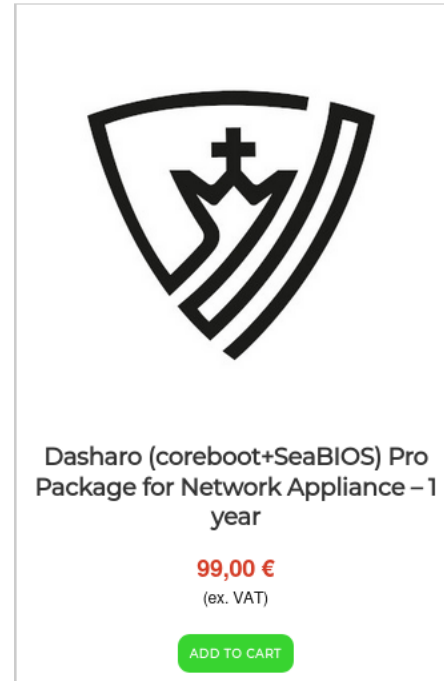
Dasharo Pro/Enterprise Package

(formerly Dasharo Entry Subscription)

Dasharo Pro/Enterprise Package

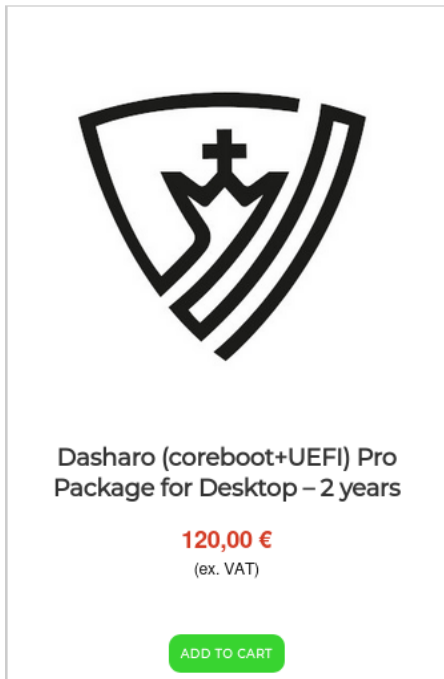


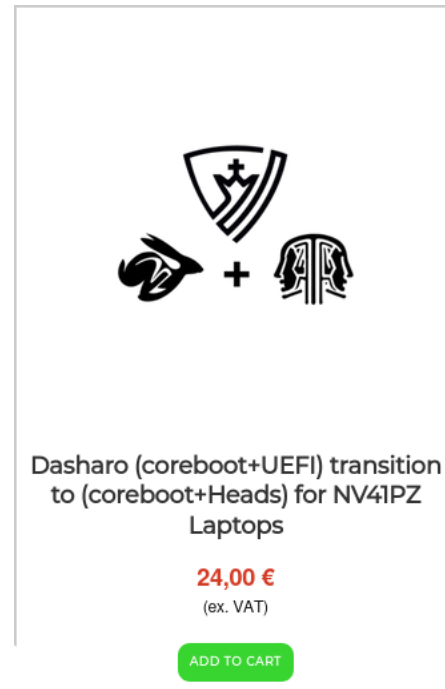




https://shop.3mdeb.com/product-category/dasharo-pro-package

Dasharo Pro/Enterprise Package





https://shop.3mdeb.com/product-category/dasharo-pro-package