PowerDebug X51 - Getting Started

2026-04-03 - Comments (0) - Setup / update



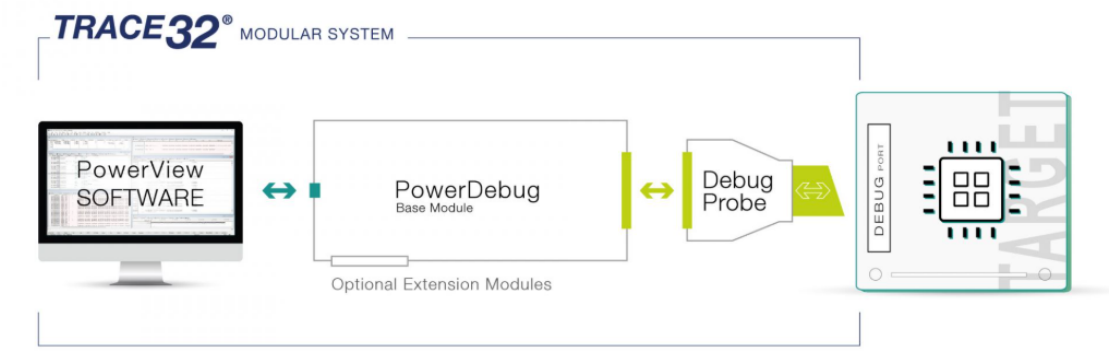
The PowerDebug X51 is a high-performance, modular, and future-proof debug controller. It can be expanded with PowerTrace, a leading embedded off-chip trace solution, as well as logic-analyzers. You can connect it to your PC via USB 3 or 2.5 Gigabit Ethernet, making it the perfect solution for both on-site and remote debugging, whereby you can even supply it with power via USB-C without a power supply unit.

No matter what your application looks like today or in the future, the PowerDebug X51 meets all your challenges, maximizes your productivity, and ensures a valuable return on investment.

Required Hard- and Software

Required Debug Probe

To use your PowerDebug X51, you need a **suitable debug probe** (aka. Debug Cable). The debug probe is plugged to the 25-pin SUB-D connector of your PowerDebug X51 labeled with “DEBUG CABLE”



To find the right debug probe for your target SoC please visit https://www.lauterbach.com/supported-platforms/chips

You can also get an overview of all kinds of available debug probes at https://www.lauterbach.com/products/debugger/powerdebug-system/debug-probes

If you did not order a new debug probe with your PowerDebug X51, please make sure that the software maintenance of your existing debug probe covers TRACE32 software **2024/09** or later.

Required PowerView Software

To use your PowerDebug X51, you'll also need the **PowerView** software from Lauterbach. It runs on Windows, Linux, and macOS machines.

Your PowerDebug X51 **requires PowerView release R.2024.09** or later (or nightly build 172015. or later).

You can download the latest PowerView release from https://www.lauterbach.com/3232

Details on the supported operating systems can be found under https://www.lauterbach.com/supported-platforms/toolchain/host-os

Setting-Up The Tool

PowerDebug Hardware

To set up the hardware, please follow these steps:

- Connect your **debug probe** (aka. Debug Cable) to the 25-pin SUB-D connector to the connector of the PowerDebug x51 labeled with "DEBUG CABLE".

- Connect the PowerDebug X51 with your PC via the supplied **USB cable**. Although the device supports all types of USB modes, we recommend using a USB 3 port on your PC. For best results, use a USB 3 port that is mounted directly on your PC's motherboard.

Alternatively, you can connect your PowerDebug X51 to your TCP/IP network using an Ethernet cable. However, we recommend using USB for the first time.

- Connect the supplied **Desktop Power Supply** to a suitable power outlet and plug the barrel connector into the PowerDebug x51 power jack labeled “POWER”. Turn on the desktop power supply (it has its own power switch).

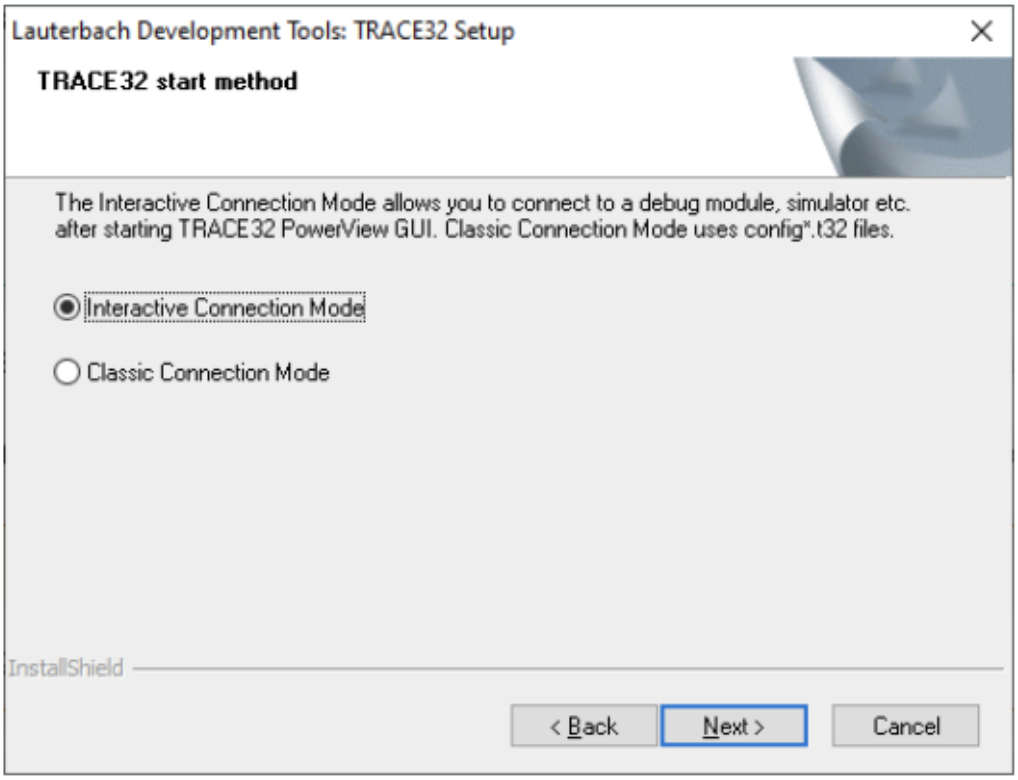
Your PowerDebug X51 should now power up. The LED labeled with POWER lights red, while the LED labeled with SELECT blinks slowly to indicate that the debugger is ready to get controlled by the PowerView software.

PowerView Software

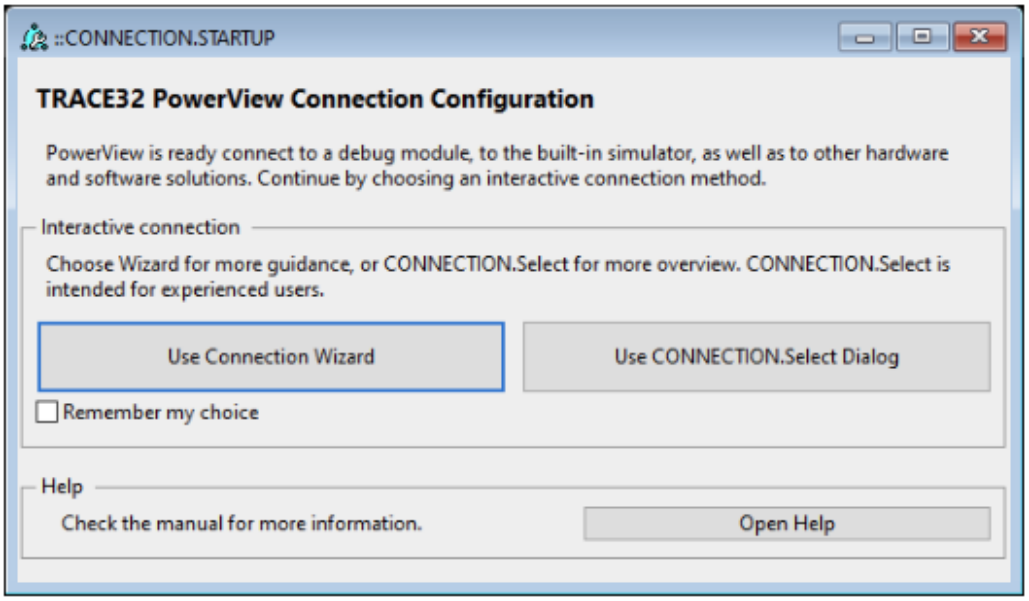
If you have not already done so, please install the Lauterbach PowerView software, as it is an integral part of your TRACE32 debug system. (If PowerView is already installed, please ensure that you’re using release R.2024.09 or later - otherwise PowerDebug X51 won’t work).

- Download the PowerView Installer from https://www.lauterbach.com/3232

- During the installation of TRACE32 PowerView, you are asked to use choose the new *Interactive Connection Mode* or the *Classic Connection Mode*. Lauterbach recommends using the **Interactive Connection Mode**



- After installation, **start the TRACE32 PowerView software**. Choose the TRACE32 program for the main architecture of your target SoC. E.g.:
    - To debug a chip of type Arm/Cortex please select “TRACE32 Arm”
    - To debug a chip of type TriCore/AURIX please select “TRACE32 TriCore”
- The PowerView GUI starts and opens a dialog to guide you to the first connection to your new debugger. Lauterbach recommends choosing **Use Configuration Wizard**.



For more information please see the **TRACE32 Installation Guide**.
This guide is especially helpful if you've chosen the *Classic Connection Mode*.

How to Use the Tool in General

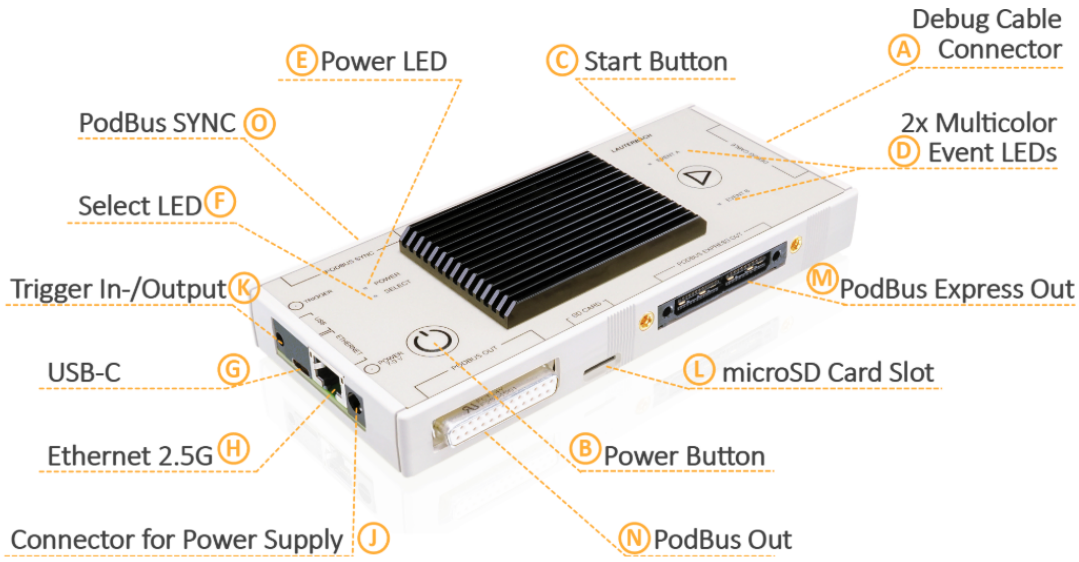
When you have never used TRACE32 PowerView before, we recommend to start reading with the **Training Basic SMP Debugging**. Continue with the **Training Source Level Debugging**.
To automate processes, learn about the PRACTICE scripting language in the **Training Script Language PRACTICE**

You can find all TRACE32 manuals online or in the subfolder **pdf** in your TRACE32 installation on your hard drive. If PowerView is already running on your desktop, access all manuals by selecting Menu → Help → Contents.

To learn more, check out the Lauterbach Knowledgebase which also includes many helpful **training videos**.

Lauterbach also offers **training classes** in multiple locations both online and face-to-face.

PowerDebug X51 Interfaces



A. Debug Cable Connector to connect a required architectures specific debug probe from Lauterbach.

B. Power Button to temporarily turn off the unit. By default, it will turn on again when power is reconnected. (Can be changed in PowerView via Menu → Misc → Interface Config...)

C. Start Button to launch scripts at your fingertip (via `GLOBALON` command).

D. Two Multicolor Event LEDs controllable from script (via `DebugModule.SIGnal` command).

E. Power LED is red when power is on, and blinks purple when power is insufficient for tool extensions (like a PowerTrace when powered via USB-C).

F. Select LED blinks red in standby mode, lights red when a PC is connected, and lights blue when running from SD card.

G. USB-C to connect the debugger to your PC, or to remotely control a serial terminal on your target device. See the "Serial Terminal Remote Control" section below for more details on controlling serial terminals. Optionally you can power the unit via USB by using a USB cable of type C-to-C (not included).

H. Ethernet to connect to your PC via network - now with up to 2.5 Gbit/s.

J. Connector for Power Supply for the supplied Desktop Power Supply

K. Trigger In-/Output is a multipurpose I/O, to react on events. e.g. force your target application to break on the rising edge of a connected signal.

L. microSD Card Slot to use the device as a standalone flasher or tester. Insert a microSD card face down. See more below at SD-Card Slot.

M. PodBus Express Out for powerful extensions like PowerTrace off-chip trace extension or a PowerIntegrator logic analyzer.

N. PodBus OUT for legacy extensions like a PowerProbe.

O. PodBus SYNC to synchronize two PowerDebugger by connecting the *PodBus OUT* of the first unit with the *PodBus SYNC of the second unit*.

How to Use the New Features

Even if you are already familiar with TRACE32, you may still be wondering how to use the new features of PowerDebug X51.

Serial Terminal Remote Control

Especially if your target is running a rich OS like Linux, you might need to control a serial terminal of the target to properly run the application, you'd like to debug. If your target board is located in a remote lab, you can use PowerDebug X51 to control the target's serial terminal remotely.

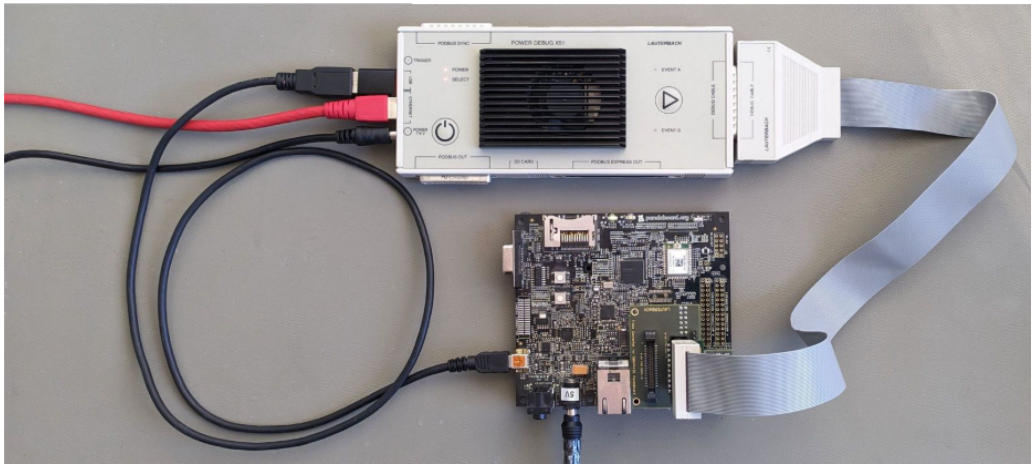
This feature **requires R.2025.02 or later**.

To remotely access your target's serial terminal, follow these steps:

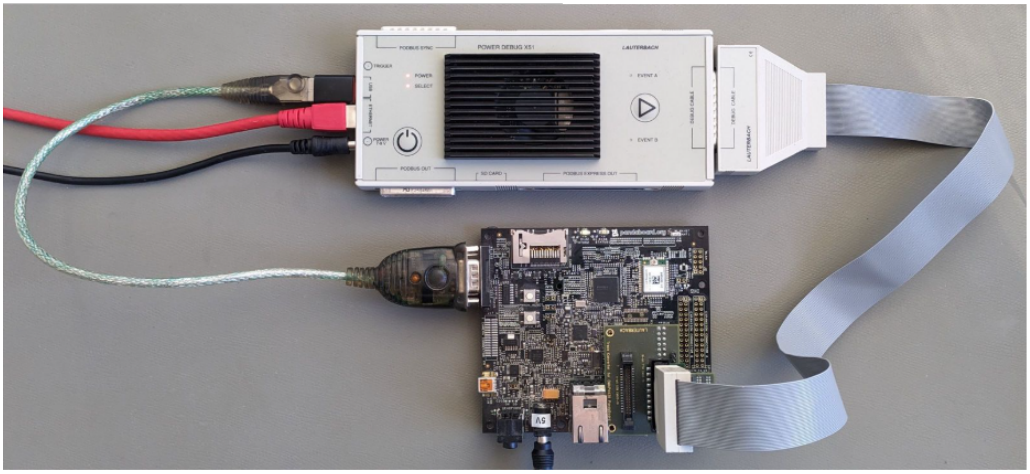
1) Connect your PowerDebug X51 to your PC via Network (Ethernet)

2) Connect your target's serial port to the USB-C port of your PowerDebug X51.

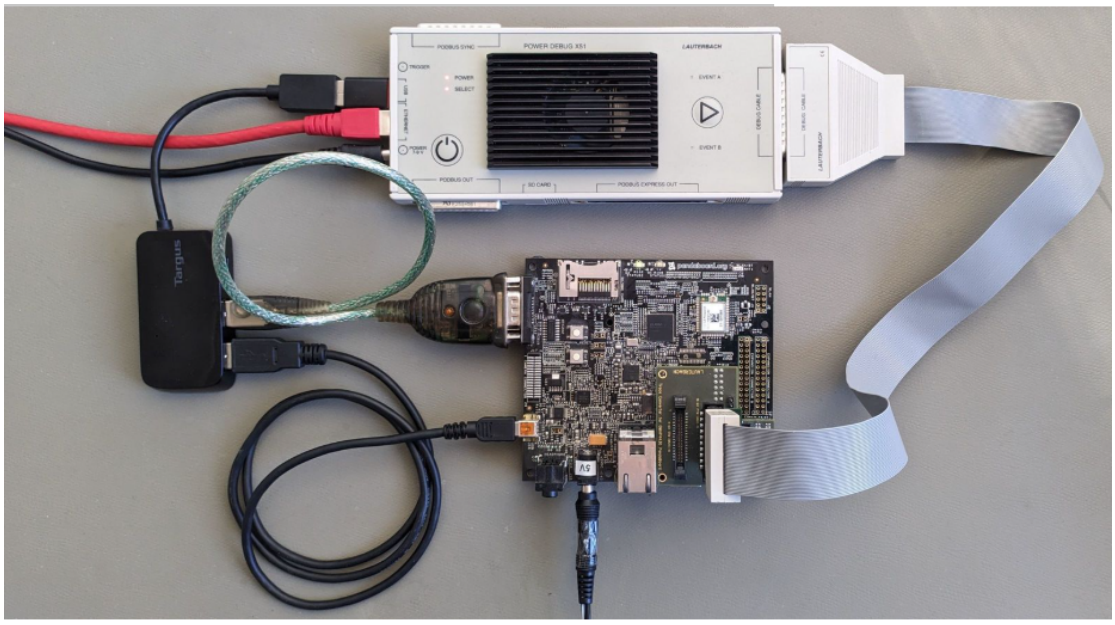
- Most target boards today already have an RS-232 to USB converter built in. In this case, all you need is a suitable USB cable:



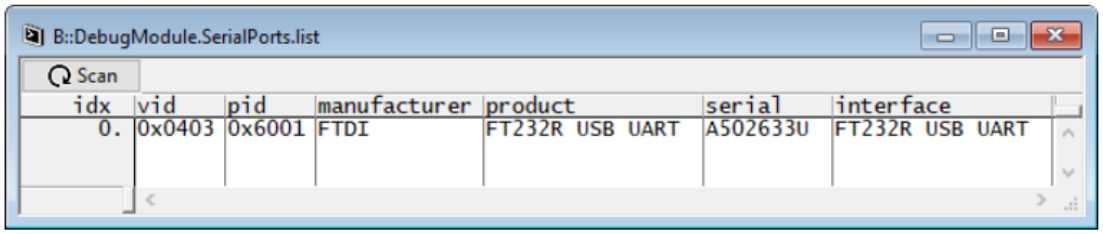
- If your target board has a classic RS-232 port with a 9-pin SUB-D connector, you'll need an additional off-the-shelf USB-to-Serial adapter:



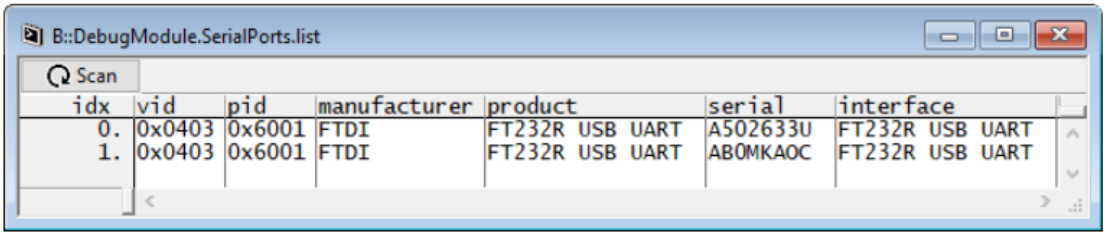
- You can also use a USB hub to connect to multiple serial ports:



- Start PowerView and execute the command `DebugModule.SerialPorts.list`. It will show you all the detected devices:



If you have more than one serial port connected to your PowerDebug X51 (via a USB hub), then you will see more than one device:



Remember the index of the terminal you want to control from the first column labeled “idx”.

Troubleshooting: If your serial terminal adapter does not appear in the `DebugModule.SerialPorts.list` window, please try the following:

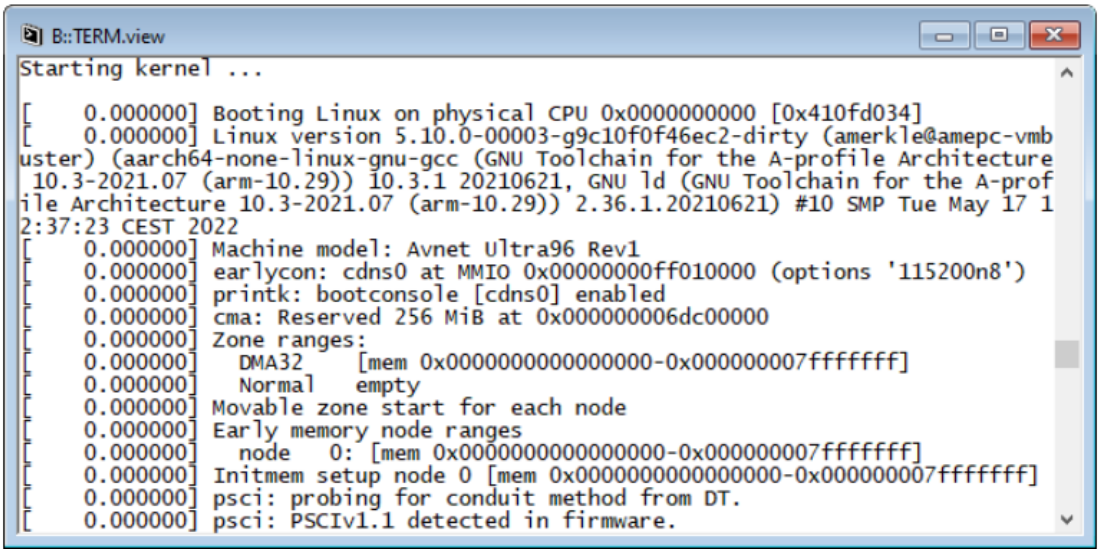
- Close PowerView and then disconnect the power supply. Then re-connect the power-supply, restart PowerView and execute `DebugModule.SerialPorts.list` again.
- Verify that your serial port is usable when connected to a PC instead of the PowerDebug X51.
- Try a different serial terminal adapter or contact technical support.

4) Now open a serial terminal as usual in PowerView, but use **DebugModule** <*idx*> as the parameter <*method*> for the command `TERM.METHOD`
Assuming that you have only one serial port connected, transceiving at 115200 baud per second, 8 bits, one stop bit, and no parity, the required command to access the terminal is as follows:

```
TERM.METHOD #1 DebugModule 0 115200 8 NONE 1STOP NONE
TERM.Mode #1 VT100
TERM.SIZE #1 80. 100. 2000.
```

For more details on `TERM.METHOD` see the General Commands Reference Guide T.

5) Finally open the terminal window with the command`TERM.view`. The output from a target running Linux might look like this:



For advanced users:
If you have more than one serial port connected, you can find the right serial port in a script with the PRACTICE functions `DebugModule.SerialPorts.COUNT()` and `DebugModule.SerialPorts.INFO(<idx>,<item>)` e.g. like that:

```
DebugModule.SerialPorts.SCAN     // Detect all serial ports
PRIVATE &idx
WHILE   &idx<DebugModule.SerialPorts.COUNT()
(
    IF "DebugModule.SerialPorts.INFO(&i,"serial")"=="A502633U"
    (
        // Found port with serial number "A502633U" => show that erminal
        TERM.METHOD #1 DebugModule &idx 115200 8 NONE 1STOP NONE
        TERM.Mode #1 VT100
        TERM.SIZE #1 80. 100. 2000.
        TERM.view
    )
    &i=&i+1.
)
```

SD-Card Slot

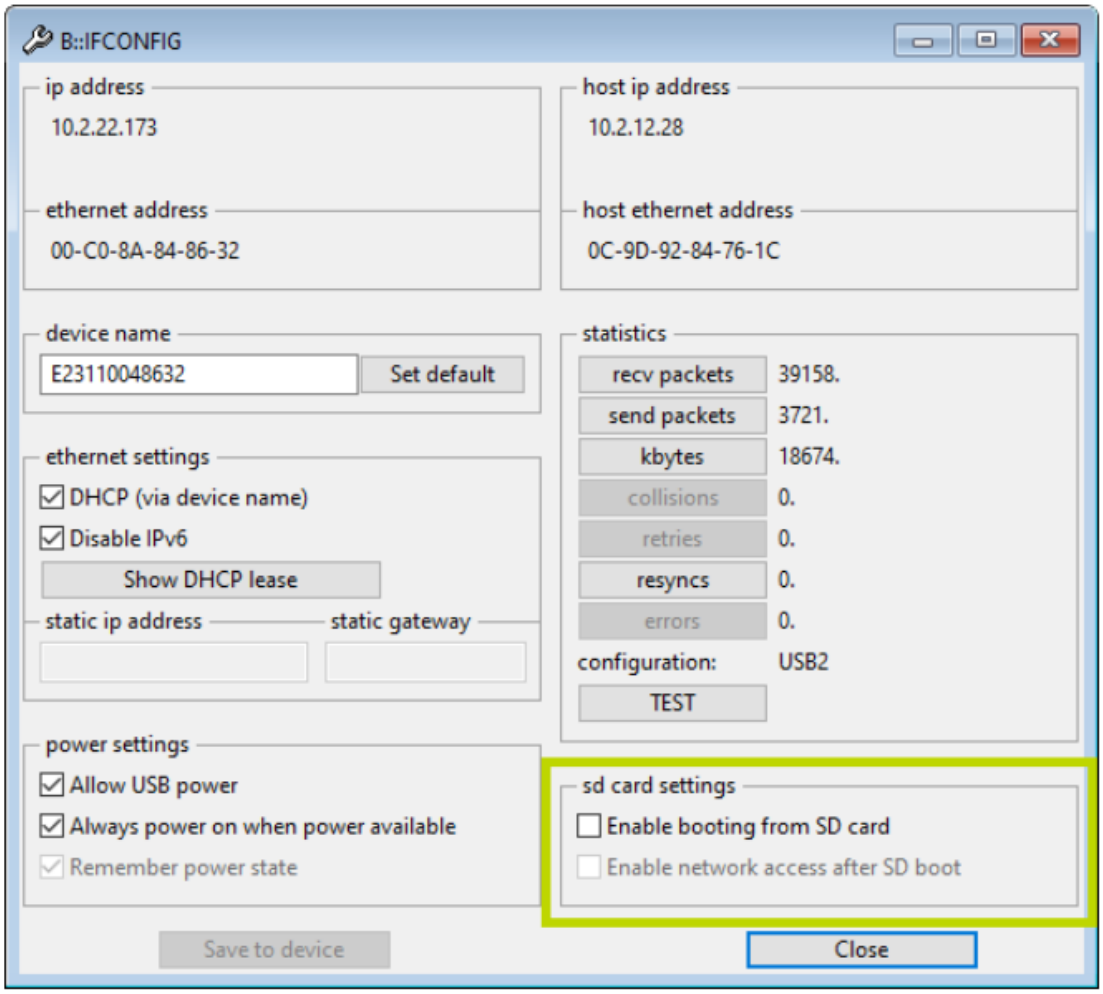
PowerDebug X51 has a microSD card slot for autonomous operation: You can run PowerView for Linux on ARM64 on the SD card together with a collection of Linux tools.
This allows to use PowerDebug X51 as a standalone tool for flash-programming or in-field testing. You can also remotely access the PowerView software running on your PowerDebug via your web-browser.

Note: microSD cards need to be inserted upside down.

This features will become available during 2026.
During 2026 Lauterbach will roll out information on how to prepare an SD card to run PowerView on the PowerDebug X51.

For **security** reasons, the microSD card slot is disabled by default. To enable the slot, connect your PC via USB and open the `IFCONFIG` dialog

via Menu → Misc → Interface Config... In the dialog select the *Enable booting from SD card* checkbox.



To indicate that a PowerDebug X51 is running from an SD card, the SELECT LED (below the Power LED) will light blue (instead of the usual red).

Although not recommended, there is also an option to permanently disable the microSD card slot by ordering the LA-3577X license. To re-enable such a permanently disabled microSD card slot, you would need to send the device to Lauterbach.

Factory Reset

For technical reasons, PowerDebug X51 stores some configuration settings (e.g. the device name) in a way that does not guarantee that old data is erased immediately, even if changed by the user.

To ensure that all user-specific data is removed reliably, you can perform a factory reset.

For details on performing a factory reset, please see the `DebugModule.FactoryRESet` command in the PowerView Command Reference.

Technical Support

If you need support, please visit the Lauterbach help center.