LAUTERBACH
DEVELOPMENT TOOLS

PowerTrace Serial User's Guide

Release 02.2025

MANUAL

PowerTrace Serial User's Guide

TRACE32 Online Help

TRACE32 Directory

TRACE32 Index

TRACE32 Documents ..... 🗁

PowerTrace Serial User's Guide **1**

Introduction **5**

Intended Audience 5

Prerequisites 5

Contacting Support 6

Installation **8**

Hardware Installation 8

PowerTrace Serial 8

Adapter for AGBT (LA-3556) 11

Adapter for RH850 (LA-3561) 12

Adapter for Mini-PCIe (LA-3526) 14

Adapter for PCIe Slot Card (LA-3525) 15

Configuration Switches 20

Configuration Guide 23

PCIe-Gen4 Preprocessor for PowerTrace Serial (LA-3529) 24

Hardware Installation 24

Target Connector Pinout 27

Startup sequence 29

Hardware Status and Link Information 30

Configuration Jumpers and Switches 32

Replacement parts 33

Preprocessor Aurora 2 for PowerTrace Serial (LA-3945) 34

Software Installation 35

Recommendation for the Software Start 35

Recommendation for Power Down 35

Trace Port Utilization **36**

HSSTP-based Trace Port 36

Start-up Script 37

Example 37

AGBT-based Trace Port 41

Serial NEXUS-based Trace Port 41

PCIe-based Trace Port 41

Start-up Script 42

Example 42
FAQ **43**
Diagnosis **44**
Device LED Codes 44
POWER 44
SELECT 44
RECORD 44
RUNNING 44
Displaying Error Messages 45
Searching for Errors 46
Types of Trace Decoder Errors 48
Harderror 48
Flowerror 48
Fifofull 49
Trace Test Failure Messages 49
Diagnosis Check List 50
Basic Checks 50
Advanced Checks 54
Recommendations for Target Board Design 55
Technical Data **56**
PowerTrace Serial Accessory Sets 56
LA-3521 - ACC-PTSERIAL-ETM 1-6 56
LA-1235 - 40pin Flex Cable 56
LA-2770 - Mipi34-ARM20 Adapter 57
LA-2114 - 34pin Half Size Cable 57
LA-3522 - ACC-PTSERIAL-ETM 7-8 58
LA-1239 - 80pin Flex Cable 58
Electrical Specification for PowerTrace Serial 59
Electrical Characteristics 59
Absolute Maximum Ratings 59
Switching Characteristics 59
Timing Characteristics 60
Maximum Lane Skew 60
Additional Data 60
PCB 60
Device Connector 60
Simulation Model 61
Simulation Data Request 61
PowerTrace Serial 61
Dimensions 61
Connector Layout 62
SerialPort 0 62

SerialPort 1 62
Logic Analyzer Probe (only PowerTrace Serial V1) 62
Aux Port V1 (only PowerTrace Serial V2) 62
DEBUG 63

PowerTrace Serial User's Guide

Version 13-Feb-2025

Introduction

PowerTrace Serial is state-of-the-art trace extension that covers all high-speed serial trace ports from low lane count, low bit rate up to the highest lane count and fastest bit rate available in the embedded market. it support all kinds of chips supporting a serial trace port. Not only supporting the Aurora/HSSTP protocol, but also excellent support for tracing over PCIe.

PowerTrace Serial can capture up to 8 serial lanes with up to 12.5 Gbit/s per lane.
Even higher trace rates are possible via optional preprocessors plugged to the device.

Higher speeds are always combined with higher demand on system design. New interfaces require digging into new papers of peripheral components. There is more work to do and higher potential of malfunction. This User's Guide intends to flatten the path to a well working Gigabit-trace setup.

Intended Audience

This manual categorizes users into the following groups:

- *Serial trace average users*: In addition to reading chapter **Installation**, they should focus on the sections **Trace Port Utilization** and **Basic Checks**.
- *Serial trace advanced users*: As they are familiar with the above chapters and sections, advanced users should also read the sections **Advanced Checks**.

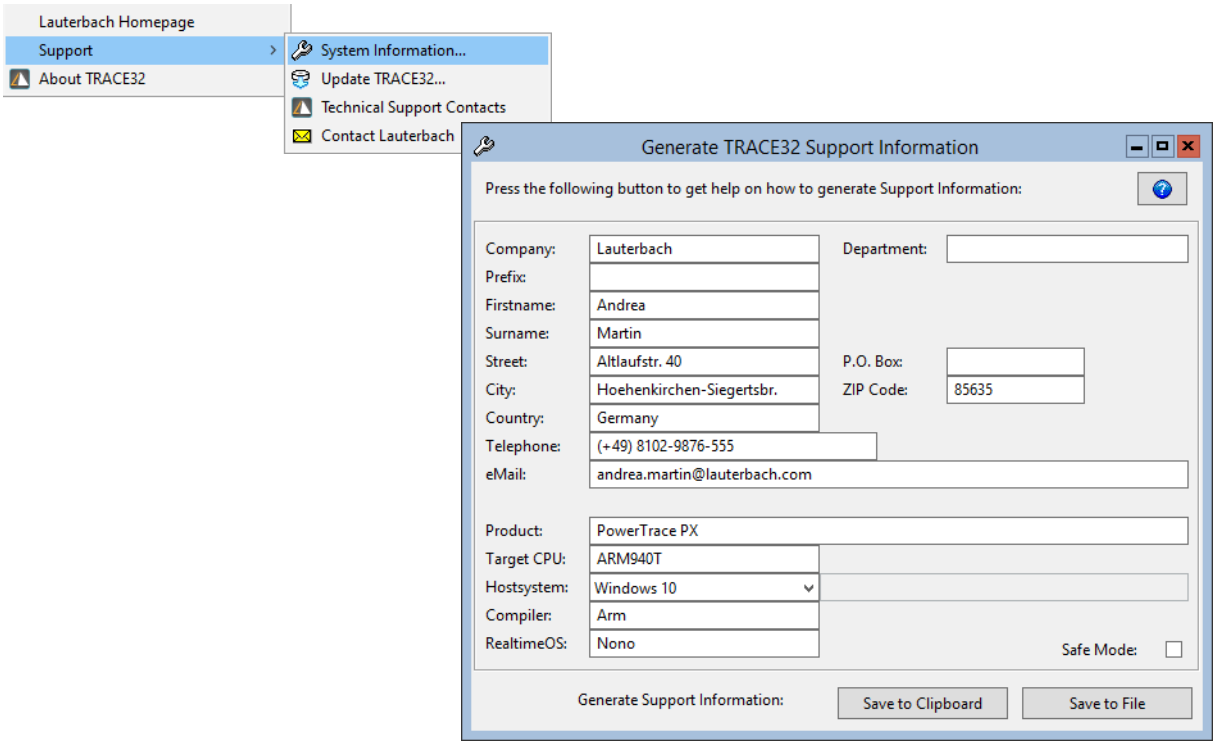
Prerequisites

- This document assumes that you already have the PowerTrace Serial hardware and a fully functional TRACE32 software installation.
- For some trace port types (e.g. PCIe), you need a different accessory set.

Contacting Support

Be sure to include detailed system information about your TRACE32 configuration.

1. To generate a system information report, choose **TRACE32** > **Help** > **Support** > **Systeminfo**.



NOTE:	Please help to speed up processing of your support request. By filling out the system information form completely and with correct data, you minimize the number of additional questions and clarification request e-mails we need to resolve your problem.

2. Preferred: click **Save to File**, and send the system information as an attachment to your e-mail.
3. Click **Save to Clipboard**, and then paste the system information into your e-mail.

Support Address

In addition to the system information report, please send the following information and files to support@lauterbach.com:

- Used start-up/configuration scripts
- List of all TRACE32 device, adapters and cables.
- A picture of your complete trace/debug hardware setup, if possible.
- The complete text of the error messages you get (**AREA.view**)
- Trace connector pinout (board schematics and layout)
- Is the problem lane speed or lane count dependent?

- A screenshot of each lane shown in **Trace.ShowFocusEye** (press SCAN before)

Installation

In this section:

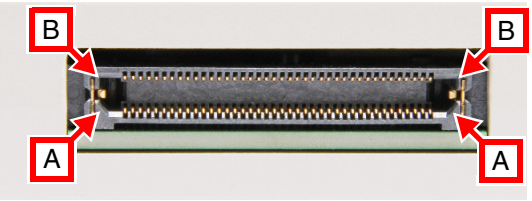
- Hardware Installation
- Software Installation

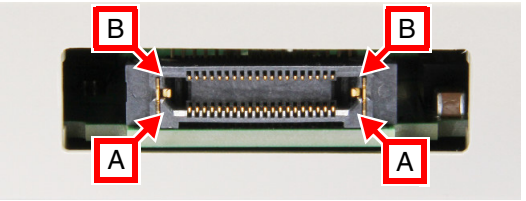
Hardware Installation

PowerTrace Serial

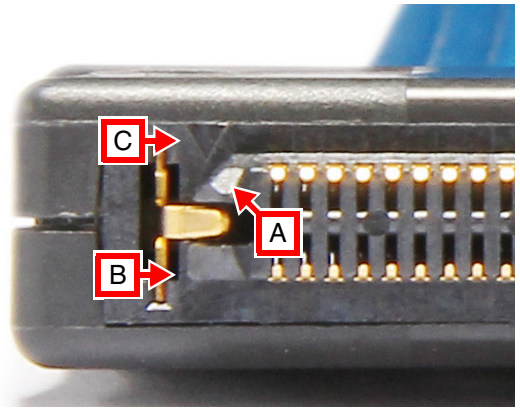
1. Connect the PODBUS EXPRESS IN connector to the PODBUS EXPRESS OUT connector of the PowerDebug II / PowerDebug Pro interface. Please ensure correct positioning. The connectors must be clean and without any damage.
2. Connect the flex cable and, if necessary, the debug cable to Serial Port 0 or Serial Port 1 of the **PowerTrace Serial.**

   Check the connector orientation of tool and target, there are two 45° corners [**A**] and two 90° corners [**B**]:





   The marker of pin 1 [**C**] of the plug is easy to miss, but is located close to one 45° corner [**A**]. For better illustration the picture below is rotated of 180°:



A 45° conners

B 90° conners

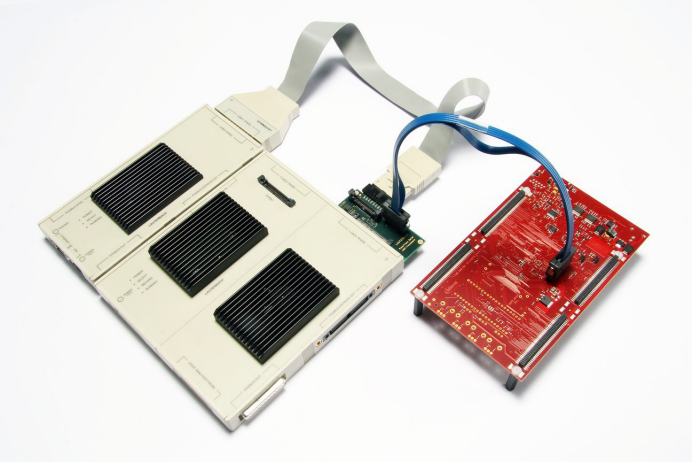
C Pin 1 marker

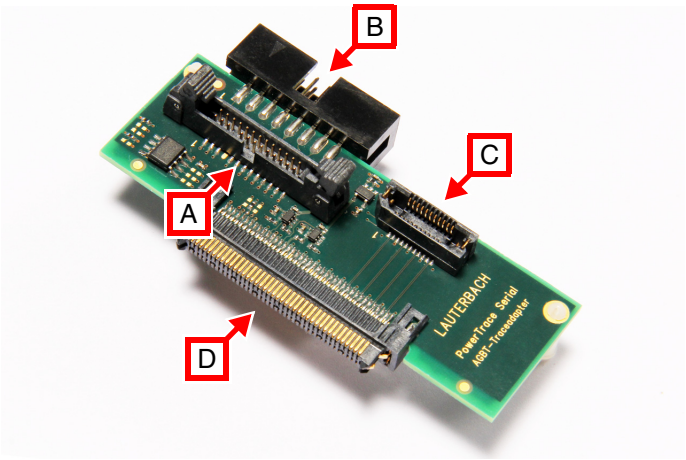
The flex cables are labeled to make the installation easier:

Serial Port 0: The debug cable can be connected to the PowerTrace Serial module or directly to the target depending on the target trace connector pin-out.

Serial Port 1: The debug cable must be connected to the target directly or

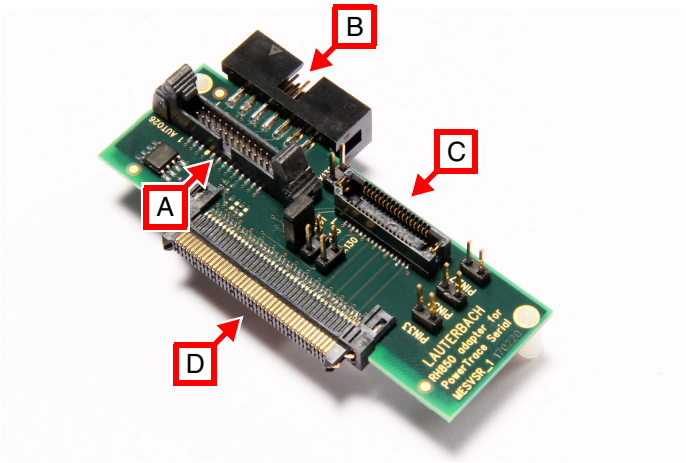
requires an adapter to merge debug and trace signals to a single cable:





Connector	Function
A	AUTO26 debug connector
B	JTAG16 debug connector
C	Target connector
D	PowerTrace Serial connector for Serial Port 1

	Both debug connectors, AUTO26 [**A**] and the JTAG16 [**B**] hold the same debug signals coming from the target connector [**C**]. Only one debug connector at the time must be used.



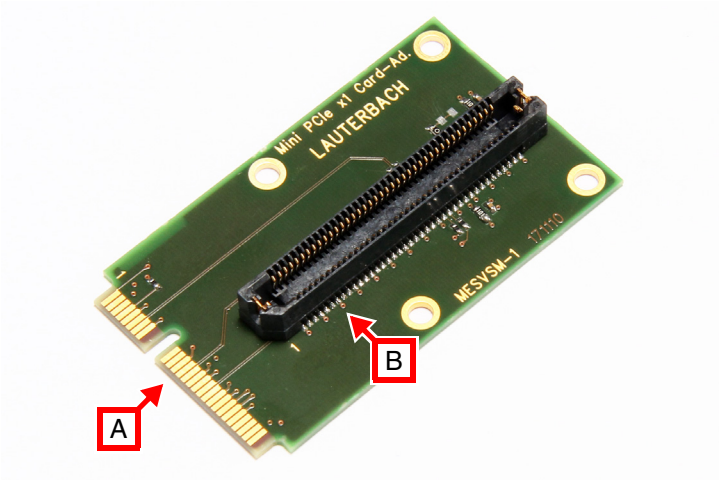
Connector	Function
A	AUTO26 debug connector
B	JTAG14 debug connector
C	Target connector
D	PowerTrace Serial connector for Serial Port 1

Jumper	Function
X130	**Set**: Connects pin 16 (EVTI) of the target connector to TRIGOUT of PowerTrace Serial **Open**: EVTI not connected
X131	**Set**: Connects pin 18 (EVTO) of the target connector to TRIGIN of PowerTrace Serial **Open**: EVTO not connected
X132	**DO NOT SET!** Pin 1: Connected to pin 34 of the target connector (RESOUT) Pin 2: GND

Jumper	Function
X113	**DO NOT SET!** Pin 1: Connected to pin 25 of the target connector Pin 2: GND
Pin27	**Set**: Connects pin 27 of the target connector to pin 14 (WD) of AUTO26 **Open**: pin 14 of Auto26 is open
Pin31	**Set**: Connects pin 31 of the target connector to pin 22 (BREQ) of AUTO26 **Open**: pin 22 of Auto26 is open
Pin33	**Set**: Connects pin 27 of the target connector to pin 24 (BGNT) of AUTO26 **Open**: pin 24 of Auto26 is open

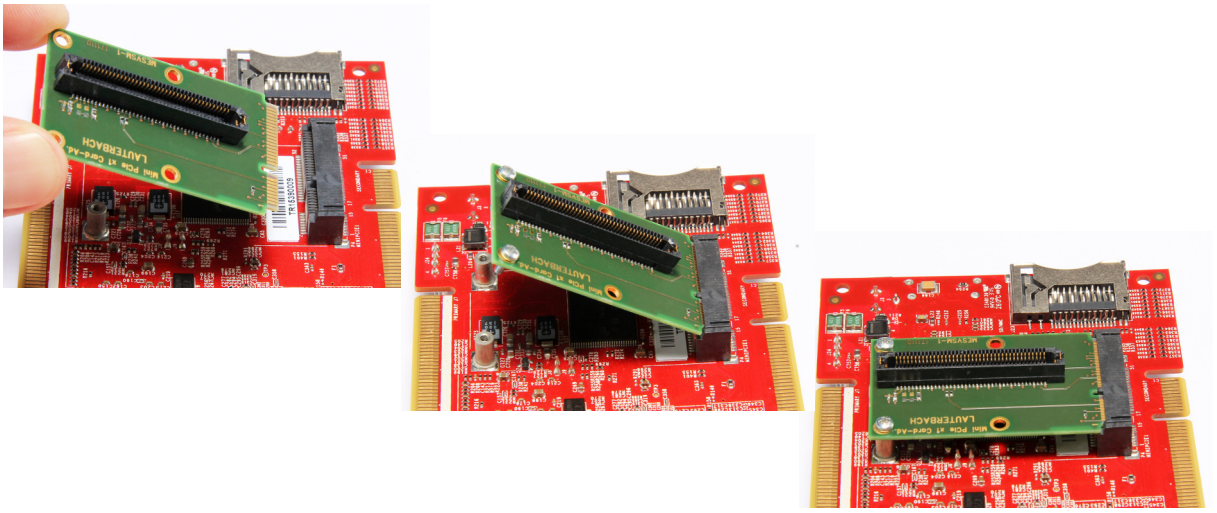
Both debug connectors AUTO26 [**A**] or the JTAG14 [**B**] hold the same debug signals coming from the target connector [**C**]. Only one debug connector must be used at the time.

The passive adapter card is used to adapt PC-like targets with trace data coming via MiniPCIe connectors.

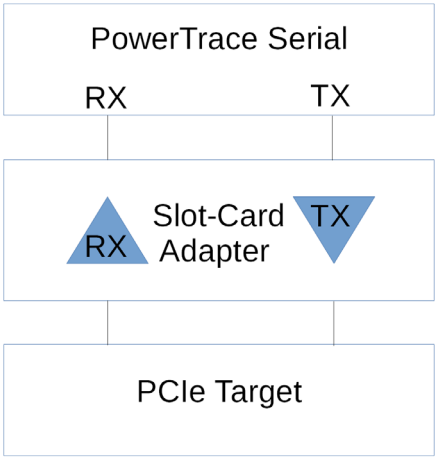


Connector	Function
A	MiniPCIe edge connector
B	PowerTrace Serial connector for Serial Port 1

Insert the adapter carefully.

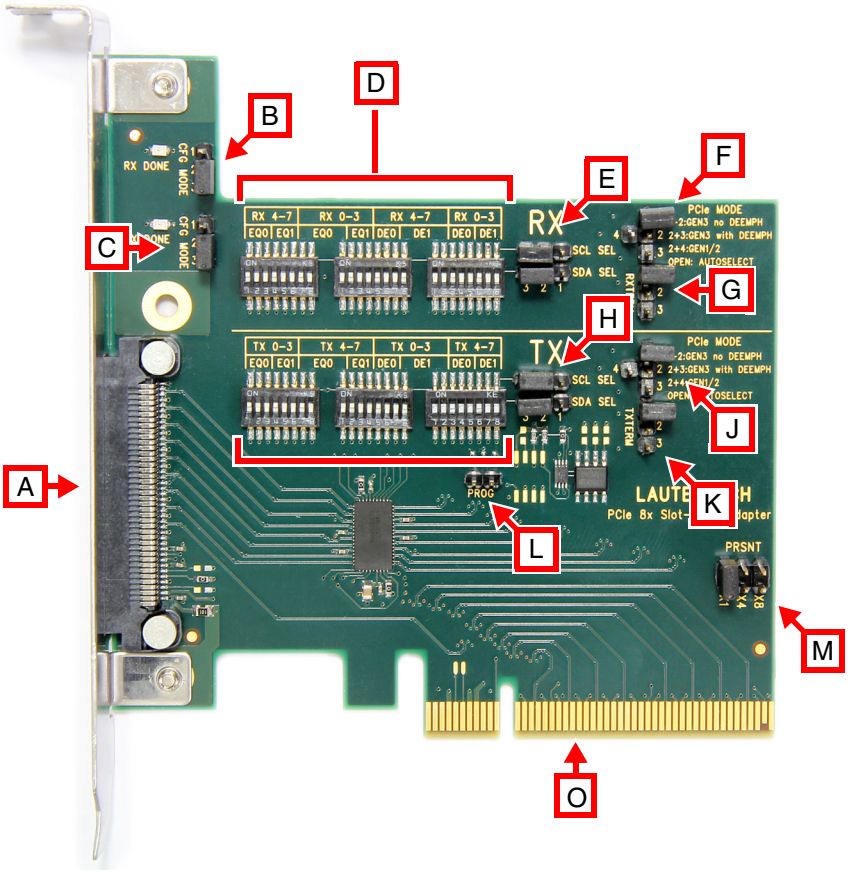


The active adapter card is used to adapt PC-like targets with trace data coming via PCIe. A special analog repeater component amplifies the electrical signal and improves the data eye. The component is transparent to target and tool. It can work in automatic configuration mode or manual configuration mode for PCIe Gen1,2 and 3.



The form factor follows the standard for PC slot-in cards. The card can be plugged into standard 16-lane-slots, 8-lane-slots and in case of special open-frame connectors also into 4-lane-slots and 1-lane-slots. If necessary the metal plate can be removed.

	The adapter will be shipped in auto-configuration-mode which will work for most of the targets. Configuration changes (e.g. deemphasis) should be done carefully and knowledge of high-speed signaling techniques is necessary.



A PowerTrace Serial tool connector

B Selector for adapter configuration mode for RX path

C Selector for adapter configuration mode for TX path

D Configuration switches for equalizer and deemphasis for groups of lanes (RX and TX)

E Selector for adapter configuration mode for RX path

F Selector for protocol specific behavior for RX path

G Selector for termination style for RX path

H Selector for adapter configuration mode for TX path

J Selector for protocol specific behavior for RX path

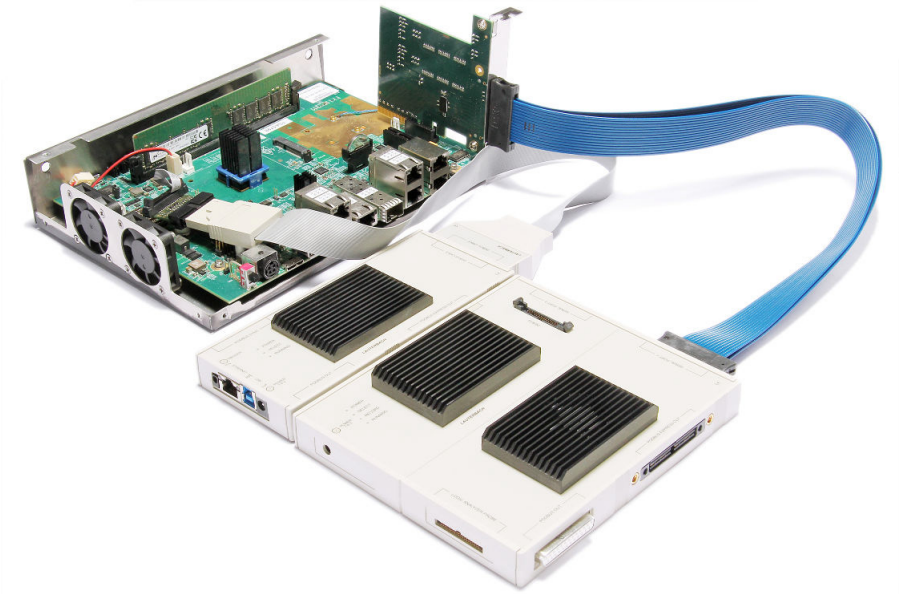
K Selector for termination style for TX path

L Selector for EEPROM configuration

M PCIe-device-present

O Target connector

A target setup could look like this:



NOTE:	It could be necessary to give mechanical support to the 80pin cable on its way from tool to target.

LED	Function
RX DONE	On: RX register configuration done Off: RX register configuration in progress, failed or target not powered
TX DONE	On: TX register configuration done Off: TX register configuration in progress, failed or target not powered

NOTE:	The adapter configuration of RX buffers and TX buffers is pipelined. The TX configuration can only succeed if RX configuration was successful.

CFG Jumper	Function
RX CFG MODE	**Must be set to 2-3** Set 1-2: SMB slave mode Set 2-3: Pin mode Open: SMB master mode (PROM mode)
TX CFG MODE	**Must be set to 2-3** Set 1-2: SMB slave mode Set 2-3: Pin mode Open: SMB master mode (PROM mode)
PROG	**Must be open** Set: PROM programming mode Open: normal mode

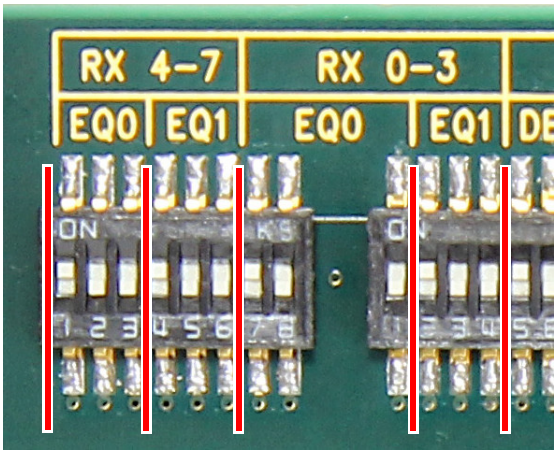
PRSNT Jumper	Function
PRSNT	Set number of lanes regarding to the PCIe standard. The target OS sometimes needs this information to configure the PCIe root complex correctly. Don't leave the jumper open. Select the next higher number of your lane count, e.g. x4 in case of 2 lanes. Set x1: select 1 lane Set x4: select 4 lanes Set x8: select 8 lanes

RX Jumper	Function
SCL SEL	**Must be set to 2-3.** Set 1-2: SMB mode Set 2-3: Pin mode
SDA SEL	**Must be set to 2-3.** Set 1-2: SMB mode Set 2-3: Pin mode

RX Jumper	Function
RXTERM	Termination behavior during signal detection. **Default: Open.** Open: unlimited cyclic RX test, Hi-Z during test then 50Ω Set 1-2: limited cyclic RX test, Hi-Z during test then 50Ω Set 2-3: manual test with 50Ω termination
PCIe MODE	Emphasis/deemphasis configuration **Default: Open** Set 1-2: PCIe Gen3 without Deemphasis Set 2-3: PCIe Gen3 with Deemphasis Set 2-4: PCIe Gen1+2 Open: Auto detection of driver parameter

TX Jumper	Function
SCL SEL	**Must be set to 2-3.** Set 1-2: SMB mode Set 2-3: Pin mode
SDA SEL	**Must be set to 2-3.** Set 1-2: SMB mode Set 2-3: Pin mode
TXTERM	Termination behavior during signal detection. **Default: Open.** Open: unlimited cyclic RX test, Hi-Z during test then 50Ω Set 1-2: limited cyclic RX test, Hi-Z during test then 50Ω Set 2-3: manual test with 50Ω termination
PCIe MODE	Emphasis/deemphasis configuration **Default: Open** Set 1-2: PCIe Gen3 without deemphasis Set 2-3: PCIe Gen3 with deemphasis Set 2-4: PCIe Gen1+2 Open: Auto detection of driver parameter

Equalization and Deemphasis can be configured separately for RX lane [0-3] and RX lane [4-7] and TX lane [0-3] and TX lane [4-7]. The configuration switches are grouped and can be set to on (1) and off (0). Three switches form a mutual group, e.g. switch[1..3]=> parameter EQ0. Within a group only one switch must be on.



	Driver settings for RX and TX will match in most cases.

#	EQ0	EQ1	dB at 1.5GHz	dB at 2.5GHz	dB at 4GHz	dB at 6GHz	Suggested for
1	**100**	**100**	2.5	3.5	3.8	3.1	FR4 and less than 5inch traces
2	**010**	**100**	3.8	5.4	6.7	6.7	FR4 5-10inch traces
3	**000**	**100**	5.0	7.0	8.4	8.4	FR4 10inch traces
4	**001**	**100**	5.9	8.0	9.3	9.1	FR4 15-20inch traces
5	**100**	**010**	7.4	10.3	12.8	13.7	FR4 20-30inch traces
6	**010**	**010**	6.9	10.2	13.9	16.2	FR4 25-30inch traces

#	EQ0	EQ1	dB at 1.5GHz	dB at 2.5GHz	dB at 4GHz	dB at 6GHz	Suggested for
7	**000**	**010**	9.0	12.4	15.3	15.9	FR4 25-30inch traces
8	**001**	**010**	10.2	13.8	16.7	17.0	8m cable, 30AWG
9	**100**	**000**	8.5	12.6	17.5	20.7	more than 8m cables
10	**010**	**000**	11.7	16.2	20.3	21.8	
11	**000**	**000**	13.2	18.3	22.8	23.6	
12	**001**	**000**	14.4	19.8	24.2	24.7	
13	**100**	**001**	14.4	20.5	26.4	28.0	
14	**010**	**001**	16.0	22.2	27.8	29.2	
15	**000**	**001**	17.6	24.4	30.2	30.9	
16	**001**	**001**	18.7	25.8	31.6	31.9	
	else	**else**					**Not allowed**

Equalizer configuration #1 is default.

#	DE0	DE1	Vp-p	DEM dB	internal Vp-p	Suggested for
1	**100**	**100**	0.8	0	0.8	FR4 and less than 5inch 4mil traces
2	**010**	**100**	0.9	0	0.9	FR4 and less than 5inch 4mil traces
3	**000**	**100**	0.9	-3.5	0.6	FR4 and 10inch 4mil traces

#	DE0	DE1	Vp-p	DEM dB	internal Vp-p	Suggested for
4	**001**	**100**	1.0	0	1.0	FR4 and less than 5inch 4mil traces
5	**100**	**010**	1.0	-3.5	0.7	FR4 and 10inch 4mil traces
6	**010**	**010**	1.0	-6	0.5	FR4 and 15inch 4mil traces
7	**000**	**010**	1.1	0	1.1	FR4 and less than 5inch 4mil traces
8	**001**	**010**	1.1	-3.5	0.7	FR4 and 10inch 4mil traces
9	**100**	**000**	1.1	-6	0.6	FR4 and 15inch 4mil traces
10	**010**	**000**	1.2	0	1.2	FR4 and less than 5inch 4mil traces
11	**000**	**000**	1.2	-3.5	0.8	FR4 and 10inch 4mil traces
12	**001**	**000**	1.2	-6	0.6	FR4 and 15inch 4mil traces
13	**100**	**001**	1.3	0	1.3	FR4 and less than 5inch 4mil traces
14	**010**	**001**	1.3	-3.5	0.9	FR4 and 10inch 4mil traces
15	**000**	**001**	1.3	-6	0.7	FR4 and 15inch 4mil traces
16	**001**	**001**	1.3	-9	0.5	FR4 and 20inch 4mil traces
	else	**else**				**Not allowed**

Deemphasis configuration #1 is default.

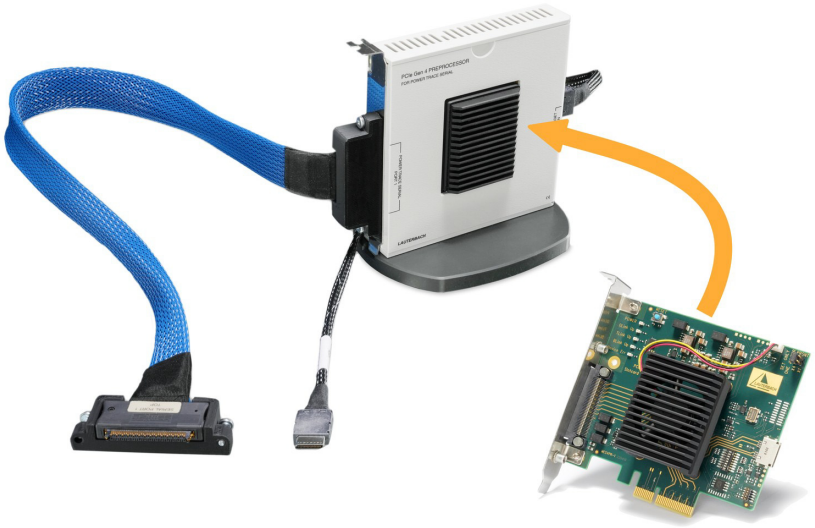
Target-specific adapter configuration might be necessary, if

- it is not possible to establish a PCIe link
- the lane speed is below the maximum possible
- the PCIe error rate is high => high bandwidth reduction

Start with the first configuration (#1) and increase step by step (#2 => #3 ...). To verify the configuration the data eye scanner should be used (**Analyzer.ShowFocusEye**).

Always first disable the switches before you enable the next configuration. Don't have more than one switch enabled within a group. Changing the configuration must be done like this:

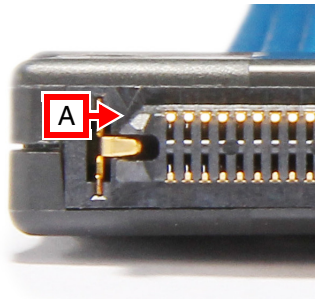
100 : current configuration
000 : set temporary configuration
001 : set final configuration



The preprocessor is used to adapt targets with trace data coming via PCIe. A special PCIe bridge component converts the x4 PCIe-Gen 4-stream coming from the target hardware to a x8 PCIe-Gen 3-stream going to the PowerTrace Serial module. Due to the downward compatibility of PCIE the preprocessor supports PCIe Gen1,2,3 and 4. The bridge component is not transparent to target and tool. A correct bus enumeration is necessary to address the attached bus devices correctly.

Hardware Installation

The blue 80pin Samtec cable is labeled with “Serial Port 1” at one end, which must be connected to the corresponding port of PowerTrace Serial module. The other end of the cable must be connected to the metal slot blade if the preprocessor. Check the connectors orientation (A) and insert carefully:



Finally, tighten the screws of the retainers lightly with a suitable screwdriver. If the trace module does not have corresponding screw holes then leave the screws loose in this case.

The PCIe-Gen4-Preprocessor uses special high-speed cables for data transmission. They consist of solid core wires and a double layer of metal foil for shielding which makes them very sensitive to bending. Avoid tight bending radii, make large bends instead:

	Don’t crimp or kink the cables! Keep the bending radius greater than 3cm (OCuLink) and 7cm (Samtec).

Note:	The PCIe-Gen4-Preprocessor is powered from the trace module. Now power is drawn from the target hardware.

Target adaptation via OcuLink

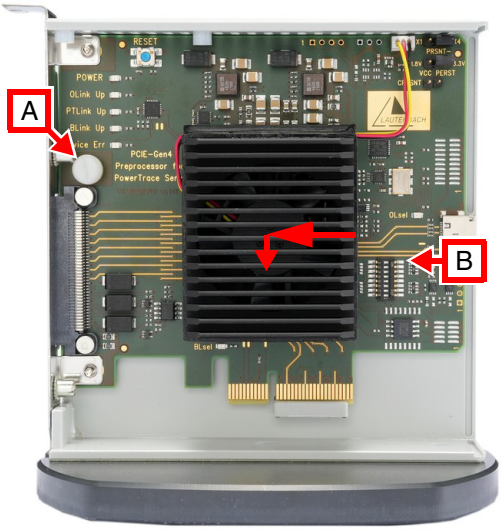
The preprocessor supports two connector types, OCuLink and edge-card. When delivered, the preprocessor is installed in the housing and already configured for OCuLink.

Connect the OCuLink-cable to the preprocessor and then to the target board. Ensure stable placement on the table for both, the preprocessor and the trace & debug modules. Take care of the cables.

Target adaptation as slot card

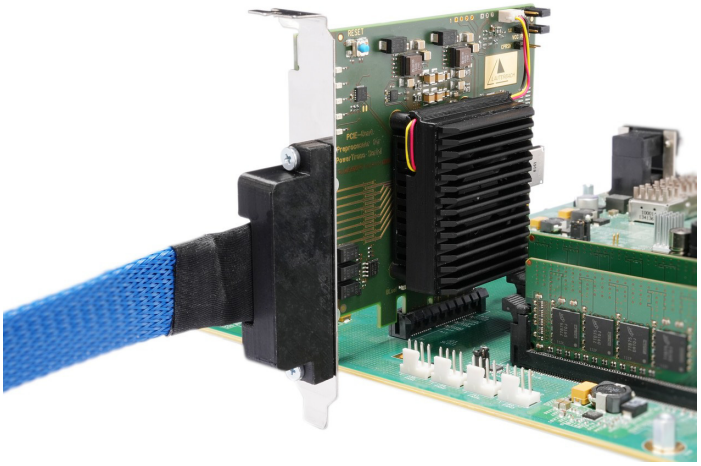
When the preprocessor should be used as plug-in card, it must be removed from the housing. The removal can be done without tools:

1. Remove all cables from the preprocessor.
2. Lift the top of the housing slightly near the recess and then pull off the entire front panel.
3. Remove the small plastic screw (A) on the left side of the PCB by hand.
4. First, slide the circuit board to the left, then move it a little downwards to release. Take it out of the box.
5. Set DIP switch #1 to ON (B) to select slot-card mode.



	Handle the printed circuit board with care, it is sensitive to electromagnetic discharges.

Place the preprocessor card in a standard PCIe-slot for x4, x8 or x16 cards. The preprocessor also fits in x1 slots with open-frame design. Finally, connect the preprocessor with the PowerTrace Serial module using the 80pin Samtec cable.



Target Connector Pinout

Only two sideband signals are used for the PCIe communication, PERST# and CPRSNT#.

CPRSNT# is voltage controlled by the preprocessor. That means, the signal will be pulled down only if TRACE32 PowerView has been started. Otherwise, the preprocessor is not powered and CPRSNT# is in tri-state mode.

PERST# is pulled high to 3.3V or 1.8V depending on the jumper setting. The signal is used to reset the port of the PCIe-bridge component.

Note:	The sideband signals PERST# and CPRSNT# are specified for 3.3V-level. Anyway, the preprocessor can be configured for 1.8V-level via jumper, if necessary. Other values are not supported.

OCuLink (root port)

Signal	Pin	Pin	Signal
N/C	A1	B1	N/C
GND	A2	B2	GND
PERp0	A3	B3	PETp0
PERn0	A4	B4	PETn0
GND	A5	B5	GND
PERp1	A6	B6	PETp1
PERn1	A7	B7	PETn1
GND	A8	B8	GND
N/C	A9	B9	N/C
N/C	A10	B10	N/C
GND	A11	B11	GND
REFCLKp	A12	B12	PERST#
REFCLKn	A13	B13	CPRSNT#
GND	A14	B14	GND
PERp2	A15	B15	PETp2
PERn2	A16	B16	PETn2
GND	A17	B17	GND
PERp3	A18	B18	PETp3
PERn3	A19	B19	PETn3
GND	A20	B20	GND
N/C	A21	B21	N/C

PCIe Slot / CEM (root port)

Signal	Pin	Pin	Signal
N/C	B1	A1	PRSNT1#
N/C	B2	A2	N/C
N/C	B3	A3	N/C
GND	B4	A4	GND
N/C	B5	A5	N/C
N/C	B6	A6	N/C
GND	B7	A7	N/C
+3.3V (optional)	B8	A8	N/C
N/C	B9	A9	(optional) +3.3V
N/C	B10	A10	(optional) +3.3V
N/C	B11	A11	PERST#
N/C	B12	A12	GND
GND	B13	A13	REFCLKp
PETp0	B14	A14	REFCLKn
PETn0	B15	A15	GND
GND	B16	A16	PERp0
PRSNT2#	B17	A17	PERn0
GND	B18	A18	GND
PETp1	B19	A19	N/C
PETn1	B20	A20	GND
GND	B21	A21	PERp1
GND	B22	A22	PERn1
PETp2	B23	A23	GND
PETn2	B24	A24	GND
GND	B25	A25	PERp2
GND	B26	A26	PERn2
PETp3	B27	A27	GND
PETn3	B28	A28	GND
GND	B29	A29	PERp3
N/C	B30	A30	PERn3
PRSNT2#	B31	A31	GND
GND	B32	A32	N/C

Startup sequence

TRACE32 start-up

- Connect TRACE32 system modules
- Connect debug and trace port (OCULink or slot)
- Start TRACE32 PowerView
  LED POWER => on
  Signal CPRSNT# => LOW

TRACE32 setup

- CPU (selection, system mode,...)
- Trace logic (base address, filter,...)
- Traceport configuration (select PCIe)
  LED PTLink => flashes

Target setup

- Configure PCIe root complex and trace source
  LED OLINK => on/flashes
- Start PCIe bus enumeration
  Lauterbach vendor-ID: 0x2031

Trace port test

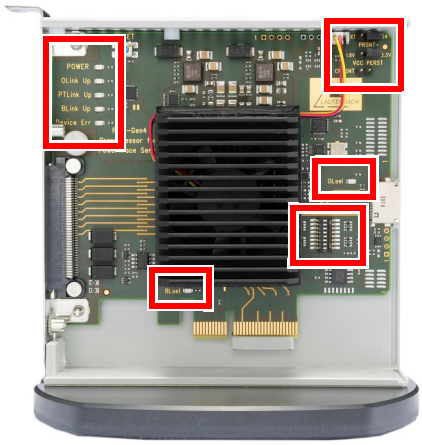
- Check data recording

.

Note:	A helper script to check the preprocessor status can be found in the DEMO-directory of your TRACE32 installation: ~~/demo/etc/diagnosis/powertraceserial/pcie_prepro_status.cmm The script is not self-updating. Just press the Update-button to do this.

Hardware Status and Link Information

The picture below shows different interesting areas of the preprocessor which will be described afterwards.



LED	Function
POWER	Power status of the preprocessor On: Preprocessor is powered, TRACE32 software is running Off: Preprocessor is not powered, TRACE32 software is not running
OLink Up	Link status of OCulink-port The PCIe-link is established when the LED lights up. Blink codes report the link speed: Gen 4: static on Gen 3: 2.0Hz blinking Gen 2: 1.0Hz blinking Gen 1: 0.5Hz blinking The LED is off if the PCIe-link is down.
TLink Up	Status of PCIe-link between PCIe-Gen4 Preprocessor and the PowerTrace Serial module The PCIe-link is well established when the LED is blinking at 2.0Hz. Deviating blink codes report reduced link speed: Gen 3: 2.0Hz blinking Gen 2: 1.0Hz blinking Gen 1: 0.5Hz blinking The LED is off if the PCIe-link is down.
BLink Up	Link status of edge-card port The PCIe-link is established when the LED lights up. Blink codes report the link speed: Gen 4: static on Gen 3: 2.0Hz blinking Gen 2: 1.0Hz blinking Gen 1: 0.5Hz blinking The LED is off if the PCIe-link is down.
Device Err	PCIe-Gen4 Preprocessor system status On: Hardware configuration of PCIe-Gen4 Preprocessor failed Off: Device configuration OK

LED	Function
BSEL	The LED is ON if the slot card connector is selected.
OSEL	The LED is ON if the OCuLink connector is selected.

If the LED nearby the OCuLink connector lights up then this connector is selected for connecting the target board. Otherwise, if the LED nearby the edge-card connector lights up then this connector is selected for connecting the target board.

Configuration Jumpers and Switches

Jumper	Function
PRSNT-	Defines the port size reported to the target board connected to the edge-card connector (PRSNT1# connected to pin 17 or pin 31 (PRSNT2#) Set to 1-2: 1-lane mode (x1) Set to 2-3: 4-lane mode (x4, default) Open: Signals PRSNT1# and PRSNT2# (pin 17 or pin 31) are floating
VCC PERST	Defines the supply voltage of the input buffer and the pullup-resistor connected to the PERST-signal. Typically, the PERST-signal is driven from target via open-drain buffer. Set to 1-2: 1.8V Set to 2-3: 3.3V (default) Open: Not allowed
CPRSNT	Controls behavior of CPRSNT-signal of the OCuLink-connector Set: Connects CPRSNT of directly to GND. Open: If TRACE32 software is running then CPRSNT=GND, else CPRSNT=HIGH.

DIP Switch	Function
1	Select target port connector On: Edge-Card connector Off: OCuLink connector (default)
2-4	Factory test (default: **Off**)

DIP Switch	Function
5	Factory test (default: **On**)
6-8	Factory test (default: **Off**)

Replacement parts

Order Number	
LA-3509	Retainer for Samtec 80
LA-1211	Flex Ext. for SAMTEC 80 pin ERM8-ERM8 500mm B
LA-1990	OCuLink Cable 500mm

This page is under construction.

Software Installation

TRACE32 PowerView includes debug and trace tool support. An extra software installation is not required. Additional license keys might be required, if architectures other than architecture covered by the included module license wants to be used.

Recommendation for the Software Start

1. Power off Target and TRACE32 hardware.
2. Disconnect the debug cable and trace cable from the target.
3. Ensure that any required adapters (e.g. **LA-3556**) and cables are connected to the PowerTrace Serial.
   From functional point of view the adapter **LA-3525** belongs to the target side and should not be connected to the PowerTrace Serial.
4. Start the TRACE32 software.
5. Connect the debug and trace cables to the target.
   - If there is no appropriate jack on your target, you can also connect the debug cable to the debug port of Serial Port 0 if this port is planned to be used (for ARM/Mipi conform devices only).
   - Alternatively, connect the debug cable to an additional converter PCB, which splits debug and trace signals. The converter PCB needs to be ordered separately at Lauterbach.
6. Connect the Serial Port 0 to your target's trace port by using the flex extension cable delivered with your accessory set. For some architectures, port sizes greater than 6 lanes or other trace protocols like PCIe, you need to use Serial Port 1.
7. Switch the target power on.
8. Run your start-up script.

NOTE:	Cable/adapter sets called "accessory set" have to be ordered additionally.

Recommendation for Power Down

1. Switch off the target power.
2. Disconnect the debug cable and trace flex extension cable from the target.
3. Shut down the TRACE32 PowerView GUI.
4. Switch off the TRACE32 hardware.

Trace Port Utilization

In this section:

- HSSTP-based Trace Port
- AGBT-based Trace Port
- Serial NEXUS-based Trace Port
- PCIe-based Trace Port

HSSTP-based Trace Port

The PowerTrace Serial can be used to trace many different architectures. Each architecture requires its own configuration on the target side to configure the CPU to send trace information.

A common trace setup consists of the following parts:

TRACE32 start-up

- Connection of system modules
- Establishing host connection
- Starting TRACE32 PowerView

TRACE32 setup

- Debug and trace port connection
- CPU (selection, system mode,...)
- Peripheral (external buffer enable,...)
- GPIO/trace port (PLL, pin manager,...)
- Trace logic (base address, filter,...)
- Application (frequency, I/O setup,...)
- Trace port (speed,...) including channel training

Trace port test

- Check trace channel state and correct data recording

TRACE32 PowerView offers a powerful feature called PRACTICE. The Lauterbach script language PRACTICE is used for automating tests, configuring the TRACE32 PowerView GUI and your debug environment.

The following example shows how to configure a trace port (ETM) and how to save the configuration.

Example

(A) The following trace port settings are required:

- Define the lane count of the trace port with the command **TRACEPORT.LaneCount**.
- Define the lane speed of the trace port with the command **TRACEPORT.LaneSpeed**.
- Definition of the endianness of the trace port is device dependent:
  TRACEPORT.MsgBItEndian
  TRACEPORT.MsgBYteEndian
  TRACEPORT.MsgWOrdEndian
  TRACEPORT.MsgLOngEndian.
- Definition of skipping bytes is device dependent:
  TRACEPORT.StartsKiP
  TRACEPORT.EndsKiP
- The need of a tool-sourced reference clock is target dependent:
  TRACEPORT.RefCLocK
- Turn on the ETM with the command **ETM.ON**.

(B) The target must be configured:

- Set up the I/O-ports (PHYs). To find the required information, refer to your CPU manual.
- Set up the board (buffers, jumpers, etc.). To find the required information, refer to your target user guide.
- Configure the operating frequency. The frequency depends on your own application.

(C) Finally, the PowerTrace Serial needs to be trained to get the channel up correctly:

- Start training sequence on target side (e.g. via script)
- Check the trace channel is up with **PRINT Analyzer.ISCHANNELUP()**.

This example is made for an CortexR5 target:

```
; JTAG DEBUGGER SETUP
SYStem.RESet                                    ; Initialize system
SYStem.JtagClock 10Mhz                          ; Select JTAG clock
SYStem.CPU CORTEXR5                             ; Select CPU type
SYStem.Up                                       ; Start debugger

; TARGET SETUP
Data.Set SD:0x10000014 %LE %L 0a05f             ; Unlock target reg
Data.Set SD:0x10000008 %LE %LONG 20             ; Set target frequency
SYStem.Option.BigEndian OFF                     ; Set endianism

; PROGRAM SETUP
Data.LOAD.ELF armle.axf /SPATH /LPATH           ; Load example PRACTICE
Register.Set PC main                            script
                                                ; Set program counter to
                                                ; program start
```

```
;(A) TRACE PORT SETUP
                                                ; Assign Coresight:
SYStem.CONFIG.ETM.BASE APB:0x8000E000           ; ETM Base Address
SYStem.CONFIG.FUNNEL1.BASEAPB:0x80004000        ; Funnel Base Address
SYStem.CONFIG.FUNNEL1.ATBSOURCE ETM 0 DTM 2
SYStem.CONFIG.ETF1.BASE APB:0x8000C000          ; ETF Base Address
SYStem.CONFIG.ETF1.ATBSOURCE FUNNEL1
SYStem.CONFIG.TPIU.BASE APB:0x80003000          ; TPIU Base Address
SYStem.CONFIG.TPIU.ATBSource ETF1
SYStem.CONFIG.ETR1.BASE APB:0x8000D000          ; ETR Base Address
SYSTEM.CONFIG.ETR1.ATBSOURCE ETF1

ETM.CORE 0                                      ; Assign core 0

ETM.PortMode CONTINUOUS                         ; Set the trace mode to
                                                ; Continuous mode

ETM.DataTrace Both                              ; Trace Address and Data

ETM.ON                                          ; Turn ETM on

SYStem.CONFIG.TRACEPORT.Type.AURORA             ; Set traceport type to
                                                ; Aurora

SYStem.CONFIG.TRACEPORT1.TraceSource TPIU       ; Connect TPIU with
                                                ; traceport

TRACEPORT.LaneSpeed 6000Mbps                    ; Set lane speed

TRACEPORT.LaneCount 4lane                       ; Set number of lanes

TRACEPORT.MsgBItEndian BigEndian                ; Change bit endianness
                                                ; of Aurora payload

TRACEPORT.EndsKiP 2                             ; Skip the last two
                                                ; bytes of an Aurora
                                                ; frame (=>skip CRC)
```

```
;(B) Target configuration
Data.Set EAPB:0x80009028 %LE %LONG 0x00219582    ; Remove PLL Bypass

Data.Set EAPB:0x8000902C %LE %LONG 0x80000000    ; Release Traceclkout
                                                 ; divider

Data.Set EAPB:0x80009000 %LE %LONG 0xF0002018    ; Set transfer reg

Data.Set EAPB:0x80009000 %LE %LONG 0xF000201C    ; Release reset

Data.Set EAPB:0x80001220 %LE %LONG 0x00000001    ; CRC enable

Data.Set EAPB:0x80001208 %LE %LONG 0x3FDFEFFF    ; Set Verify, bond and
                                                 ; alignment lengths

Data.Set EAPB:0x80001214 %LE %LONG 0xf           ; Configure lane count

;(C) Train Serial Channel and Test
&ad=0x80001204
TIMEOUT 2s WHILE !(Analyzer.ISCHANNELUP())       ; Repeat channel
(                                                ; training until succeed

Data.Set EAPB:0x80001200 %LE %LONG 0x00000000    ; Set STP link reset
Data.Set EAPB:0x80001200 %LE %LONG 0x0000000C    ; Remove STP link reset
Data.Set EAPB:0x80001200 %LE %LONG 0x0000000D    ; Enable channel init
WAIT CONV.INTTOBOOL(D.long(EAPB:&ad)&0x4f) 1s    ; Wait for pattern done

)

IF !(TIMEOUT())                                  ; Check for time-out
(
Data.Set EAPB:0x80001200 %LE %LONG 0x0000000F    ; Enable STP Link
                                                 transmit
)
ELSE
(
 PRINT "Failed to get channel up after 2s"       ; Report error
)

PRINT Analyzer.ISCHANNELUP()                     ; print channel status

ENDDO                                            ; End of script
```

AGBT-based Trace Port

The trace port setup is fully integrated in TRACE32. No separate configuration is required for common TRICORE architectures.

Serial NEXUS-based Trace Port

The trace port setup is fully integrated in TRACE32. No separate configuration is required for common NEXUS architectures like QorIQ, MPC57xx and RH850.

PCIe-based Trace Port

The PowerTrace Serial can be used to trace many different architectures. Each architecture requires its own configuration on the target side to configure the target device to send trace information via PCIe.

A common trace setup consists of the following parts:

TRACE32 start-up

- Connection of system modules
- Establishing host connection
- Starting TRACE32 PowerView

TRACE32 setup

- Debug and trace port connection
- CPU (selection, system mode,...)
- Trace logic (base address, filter,...)
- Trace port

Target setup

- Configuration of PCIe root complex and trace source
- Peripheral (external buffer enable,...)
- GPIO/trace port (PLL, pin manager,...)
- Application (frequency, I/O setup,...)

Trace port test

- Check trace channel state and correct data recording

TRACE32 offers a powerful feature called PRACTICE. The Lauterbach script language PRACTICE is used for automating tests, configuring the TRACE32 PowerView GUI and your debug environment.

The following example shows how to configure a trace port (PCIe) and how to save the configuration.

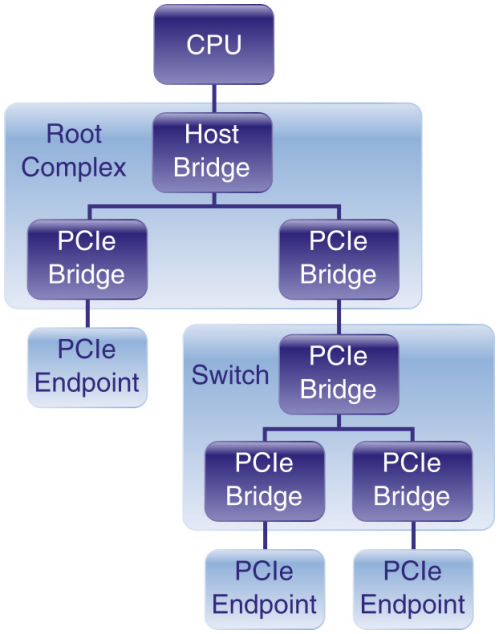
Example

(A) The following settings are required:

- Configuration trace environment with the command **SYStem.CONFIG TRACEPORT**.
- Init analyzer with the command **Analyzer.Init**.

(B) The target must be configured:

- Set up the I/O-ports (PHYs). To find the required information, refer to your CPU manual.
- Set up the board (buffers, jumpers, etc.). To find the required information, refer to your target user guide.
- Configure PCIe root complex. PowerTrace Serial is always an endpoint



- Set up trace source.

(C) The PowerTrace Serial will be trained by the root complex. Finally the channel state should be checked:

This example is based on an LS1046 target

```
; JTAG DEBUGGER SETUP
SYStem.RESet
SYStem.JtagClock ...
SYStem.CPU ...
SYStem.Up

; TARGET PERIPHERAL SETUP
Data.Set ...
SYStem.Option ...

; TARGET APPLICATION SETUP
Data.LOAD.ELF ...
Register.Set PC ...
Go.direct main


;(A) TRACE32 TRACE SETUP
SYStem.CONFIG.TRACEPORT.Type.PCIE.
Analyzer.Init


;(B) TARGET PCIe SETUP (script)
DO setup_ls_pcie "0x03600000" "0x5000000000"

;(C) TARGET TRACE SETUP (script)
&EtrAxiBase=0x5049000000
DO etr_utility.cmm ETR1 set DAB &EtrAxiBase
DO etr_utility.cmm" ETR1 set RSZ 0x8
DO etr_utility.cmm" ETR1 set AXICTL 0x00000F00 0x00000FBF

Trace.METHOD Analyzer

ETM.Trace ON

ETM.ON


;Test the state of the trace channel
PRINT Analyzer.ISCHANNELUP()

ENDDO
```

FAQ

Please refer to https://support.lauterbach.com/kb.

Device LED Codes

POWER

This LED gives information about the power state of PowerTrace Serial. It can blink or be off or on. The blink codes give further information about the issue:

Code	Note	Comment
1111111111111111	permanent on	power good
0000000000000000	permanent off	power off
1010100000000000	3x blinking	power connector used, but under/over voltage detected
1010000000000000	2x blinking	power fail on Serial Port 1
1000000000000000	1x blinking	main power regulation fail or external power connector is used, but no power sourced (=>0V)

SELECT

There can be multiple devices connected to the TRACE32 PodbusExpress. The LED reports which device is currently selected. It can blink or be off or on.

RECORD

The trace tool is armed and ready to receive data from target trace port. It can be off or on.

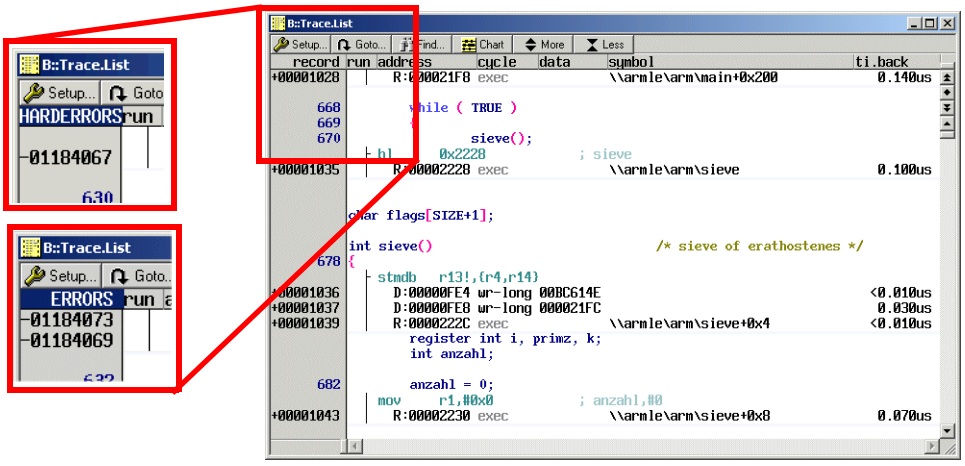
RUNNING

There are trace data coming in. The trace tool is receiving and the trace buffer gets filled. It can blink or be on or off.

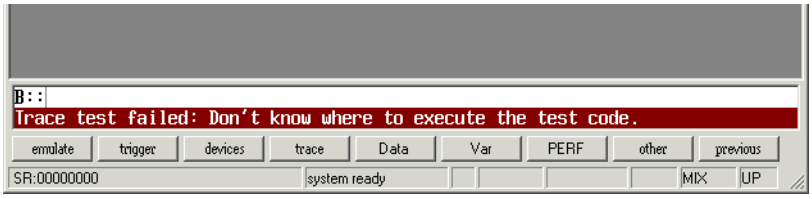
Advanced trace analysis commands like **Trace.STATistic.Func**, **Trace.STATistic.TASK** or **PERF.List** display only accurate results if the trace recording works error free.

Error messages are displayed:

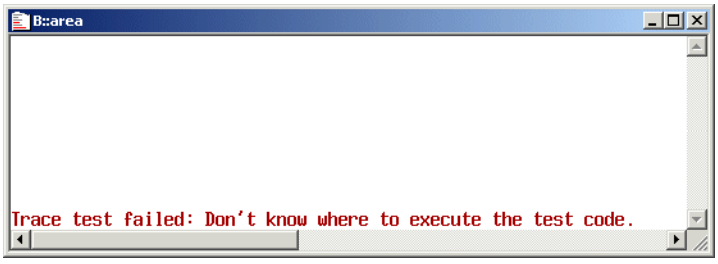
- In the upper left corner of the **Trace.List** window:



- In the message line:



- In the **Area.view** window:

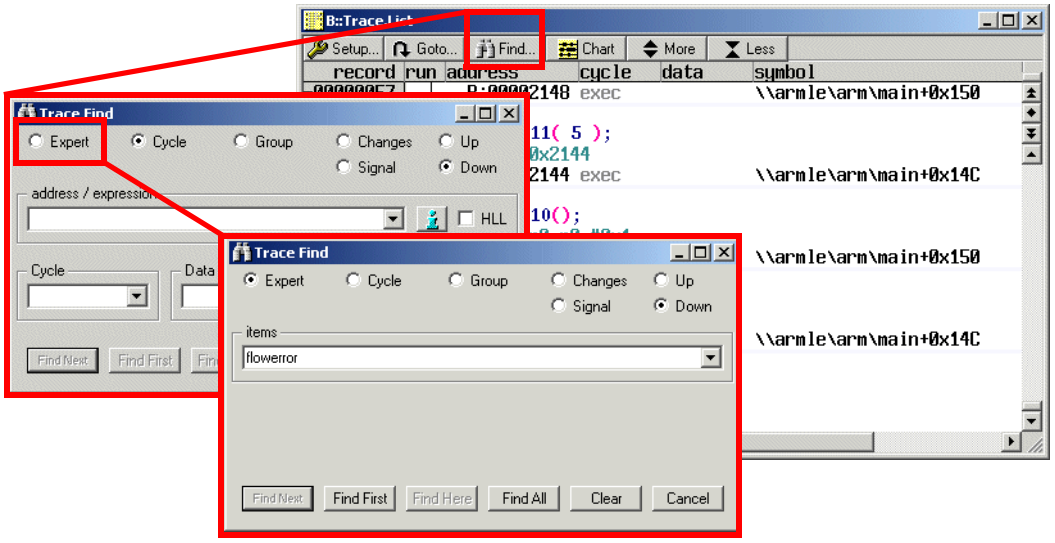


TRACE32 uploads only the requested trace information to the host to provide a quick display of the trace information. Consequently, errors that are out of the uploaded range are not immediately visible.

There are several ways to search for errors within the trace, all of them will force TRACE32 to upload the complete trace information to the host:

1. The **Trace Find** dialog:

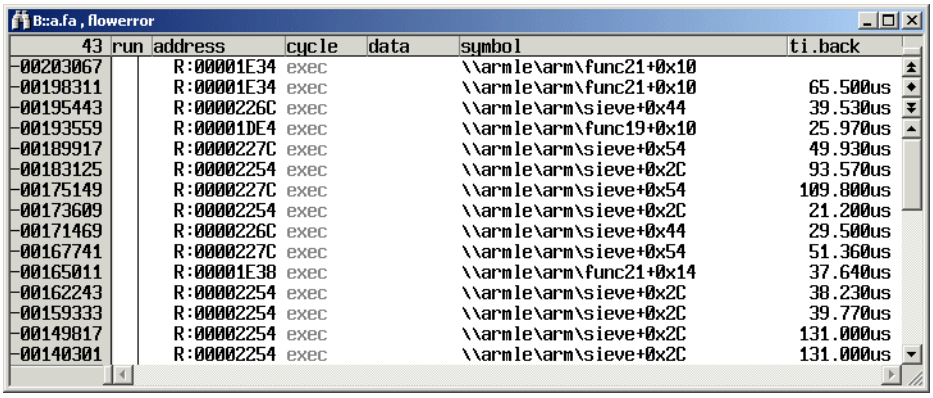
   - In the **Trace.List** window, click **Find** to open a dialog with special search options:



   - Select **Expert** and enter “flowerror” in the **items** field. The item entry is not case sensitive.
   - Use the **Find First** and **Find Next** buttons to jump to the next flowerror within the trace.
   - Click **Find All** to open a window listing all flowerrors.

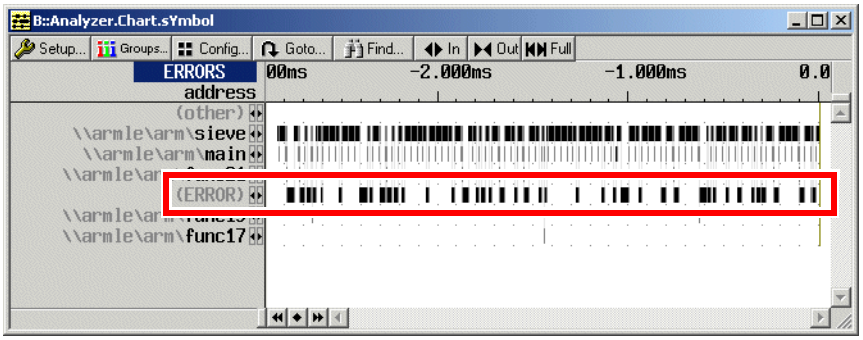
2. The command **Trace.FindAll , FLOWERROR**

This command searches for **errors** within the entire trace buffer. The records are listed in a window. The command corresponds to **Find All** described above.



3. The command **Trace.Chart.sYmbol**

This command starts a statistical analysis. An additional symbol (ERROR) is shown if errors were found.



The search could take a long time depending on the used memory size of the trace module and the type of host interface. Check the status to estimate the time.

Types of Trace Decoder Errors

One of the following errors may occur:

- **Harderror**
- **Flowerror**
- **Fifofull**

Harderror

There is no valid trace data. Possible reasons are:

- Wrong lane order
- Wrong endianness configuration (bit/byte/word/long)
- Skipping wrong amount of bytes (CRC or not)

Please see **Diagnosis Check List**.

Flowerror

The traced data is not consistent with the code in the target memories. Possible reasons are:

- Memory contents have changed (e.g. self modifying code).
- Wrong endianness configuration (bit/byte/word/long)
- Skipping wrong amount of bytes at the end of each Aurora frame (CRC or not)

Please see **Diagnosis Check List**.

Fifofull

The trace output Fifo has overflowed. The amount of trace data generated by the trace logic was greater than the trace port band width. To reduce the risk of a Fifo overflow:

- Increase the lane count if possible.
- Increase the lane speed if possible.
- Restrict the **DataTrace to read cycles** (write accesses can be reconstructed via **CTS**).
- Restrict the **DataTrace to write cycles**, a Fifo overflow becomes less likely.
- Reduce amount of trace data by using filters: use the filter **TraceEnable** or **TraceData**
- STALL the CPU if a Fifo overflow is likely to happen, if supported.
- Suppress the output of the data flow information if a Fifo overflow is likely to happen, if supported (**ETM.DataSuppress**).

Trace macrocells are not always able to prevent overflows of the internal Fifo. Even when STALL is enabled overflows may occur.

Trace Test Failure Messages

Trace.TestFocus supports a built-in trace test. This command loads a short test program to the target memory (RAM) and traces its execution. Afterwards the recorded program flow and data pattern will be checked for any errors.

“Analyzer data capture o.k.” will be shown if the test was successful.

Test failures might be caused by a variety of reasons, usually error messages such as:
“Trace test failed: not enough samples in the trace” will give you a clue as to what might have caused the failure.

For explanations of the error messages, refer to **”Error Message Emulator”** in **“Error Messages”** (error.pdf).

Diagnosis Check List

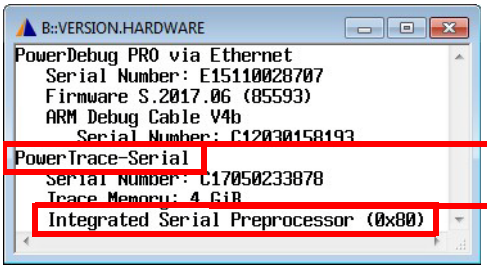
In this section:

- Basic Checks
- Advanced Checks

Basic Checks

1. Check

Check the preprocessor seen by the TRACE32 software. **VERSION.HARDWARE** shows all detected hardware.



ID	Preprocessor
0x80	Integrated Serial
0x82	Aurora 2
0x8a	PCIe-Gen4

2. Check

Check the trace port pinout. Lauterbach tools assume that the trace port pinout follows exactly the trace specification of the used architecture (please refer to **Connector Layout**).

Did the PowerTrace Serial work for other targets? If yes, what has changed on your new target board? Often messages such as **Trace test failed: not enough samples in the trace** or **Trace test failed: pin connection error** might indicate the source of the error.

Error message	Possible reason
Trace test failed: not enough samples in the trace	• Trace port does not send data
Trace test failed: pin connection error	• Trace port data format differs from the expected
Serial trace connection had errors	• The serial link is not stable

Also check the voltage level of the reference voltage. It is used as a reference for trigger and Vendor-I/O signals. It should correspond to the amplitude of your trigger signal.

3. Check

Double-check that the trace connector is properly connected to your target and that the orientation is correct. For trace port sizes with more than 6 lanes, Serial Port 1 must be used.

4. Check

Did your debugger remain in control over the target at all times when attempting to capture a trace? Error messages such as "**emulation debug port fail**" indicate that the debugger lost control over the target. In case your debugger lost control over the target:

- Is there a separate JTAG connector on your target?
  If available, connect the debug cable directly to this connector. Ensure that the JTAG signals are routed to only one connector, either trace connector or debug connector. Y-routing of signals will decrease signal integrity by reflections.
- Are all supply voltage levels stable when the trace port is active?
  Double check your targets supply voltage.
- A debug port fail can also happen in case of trace decoder errors based on inconsistent or damaged trace data. Clear the trace buffer and try to single step instructions. Do you see them displayed in the trace list window correctly?

5. Check

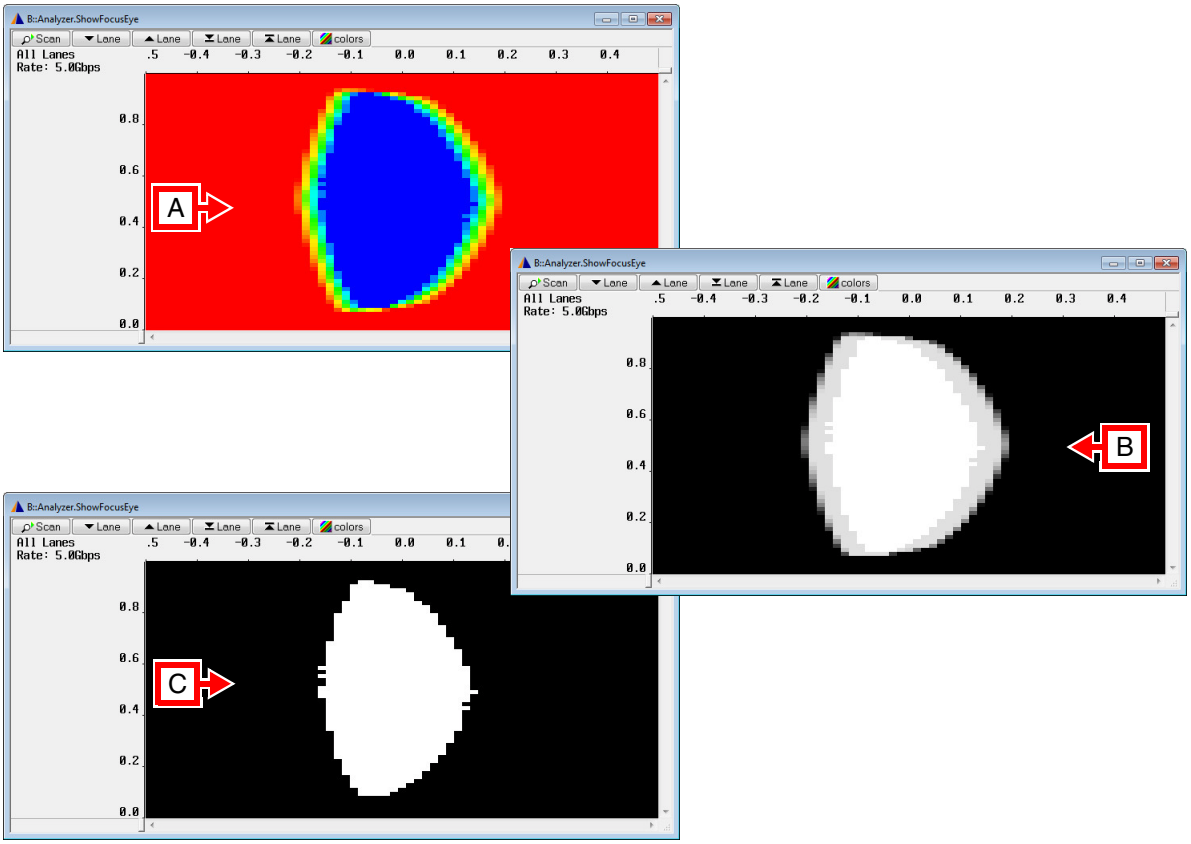
Some CPU types do not have dedicated trace port pins. Instead trace signals are multiplexed with other signals (e.g. SATA and Aurora). A special port pin setup may be required to get trace functionality. Check your CPU manual for the correct port pin configuration.

Advanced target applications might use more than one initialization procedure or the setup might change during run time again. Make sure that the trace port is actually enabled when attempting to trace.

6. Check

In case of shared trace pins, additional buffers (e.g. PCIE switches) may be used on the target hardware. Make sure that these buffers are enabled and configured correctly.

TRACE32 offers a powerful feature called **Analyzer.ShowFocusEye** to visualize the signal integrity of the trace port. It opens the window with a analog view of the data eyes:



A Rainbow style

B Gray style

C Black&white style

The horizontal axis reflects the time line in UI (unit intervals). The vertical axis reflects the voltage range of the input buffer. Press SCAN to update the measurement. With the button COLOR you can change the coloring of the graph. Rainbow scale [**A**] and grey scale [**B**] give an idea about the margin (darkblue = stable). Black and white scaling [**C**] shows the real usable range (white = stable/usable, black = unstable). The window gives you an idea about the signal integrity of your lanes. Each lane can be displayed separately by using the up/down buttons.
Mostly, if you only see a black or red screen (maybe with stripes) then the receiver did not lock. Either there is no trace signal or the target lane speed is not as expected.

	The graph shown in the **Analyzer.ShowFocusEye** window is similar to a sampling scope measurement. It reflects the trace signal seen by PowerTrace Serial. The absolute values cannot be guaranteed, but give an suitable impression of the signal integrity and what the tools does “see”.

8. Check

Is your TRACE32 software up to date? You can check by using **Help -> Support -> Update TRACE32**

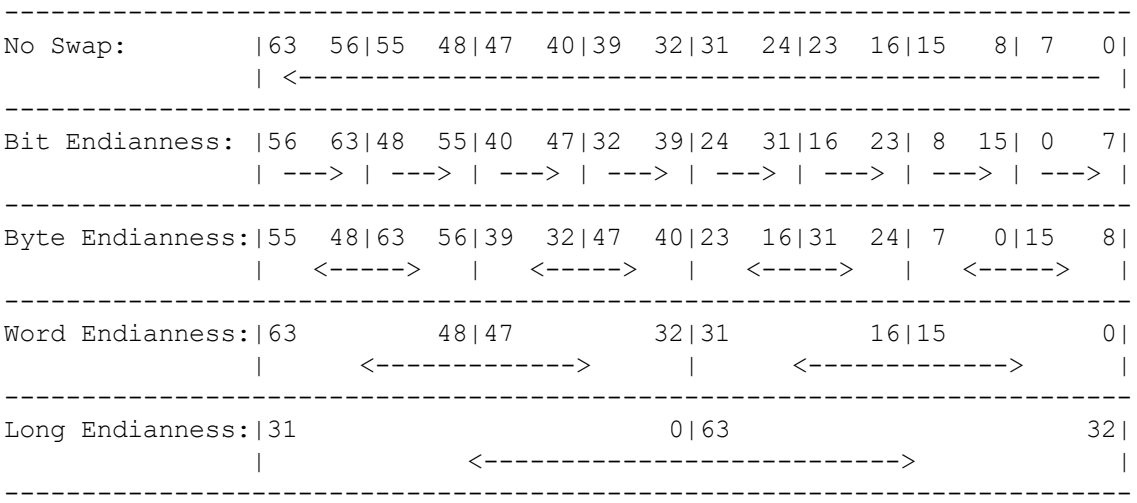
```
VERSION.SOFTWARE
```

1. Check

The bit order of incoming trace data might be wrong. This can happen by user fault (wrong configuration) or by manufacturer fault (chip bug). The PowerTrace Serial allows to change bit, byte, word and double-word (long) order by the following commands:

- **TRACEPORT.MsgBItEndian**
- **TRACEPORT.MsgBYteEndian**
- **TRACEPORT.MsgWOrdEndian**
- **TRACEPORT.MsgLOngEndian**

The following table shows the influence of each swap to the bit order:



Changing the endianness will have effect to the payload (trace data) only. The Aurora protocol data (e.g. /SOF/ or /IDLE/) are not influenced.

- Place the trace connector close to the target processor. Make sure that the ground and the latch pins are connected to your target's ground plane.

  For information on Samtec ERF8 connectors please refer to:
  www.samtec.com

  Additional information on flex extension cables can be obtained by contacting support@lauterbach.com.

- Keep lane length short to keep attenuation caused by PCB material low.
- Lane matching below 1cm (half inch) is sufficient, because it can be handled by the transmission protocol.
- +/- signal-matching must be done as good as possible. Make it on every direction-change even if it is a very small difference. Further, the matching must be done before (seen from target to tool) any layer change (via) happens.This keeps the best signal integrity.
- Cut out the GND-plane below pads of connectors and components to reduce parasitic capacitance.
- Avoid stubs e.g. by using only top or bottom layers. Use back-drilled vias if middle layers are used.
- Ideally lanes should have a 100 Ω impedance, but 5% tolerance is suitable. Use impedance controlled PCBs. FR4-material is suitable for short distances (10cm/ 4inches).
- Prevent use of signal vias. Ideally stay on one layer.
- Place GND-stitching vias close to signal vias.
- Don’t change the reference plane of the signals like e.g. GND->VCC->GND. Use only GND as signal reference.
- An adequate number of bypass capacitors for DUT are crucial to keep the supply voltage stable when the trace port is driven by your application. If your supply voltages are not stable, the high-speed trace signals might be instable too.
- Capacitor vias should never be shared, each capacitor requires its own vias close to the pads.
- The target voltage (VCCSENSE) has to be within the specified range, see **Electrical requirements**. For other voltage levels, contact **support@lauterbach.com**.

Technical Data

In this section:

- PowerTrace Serial Accessory Sets
- Electrical Specification for PowerTrace Serial

PowerTrace Serial Accessory Sets

Product number	Product code	Includes
LA-3521	ACC-PTSERIAL-ETM 1-6	LA-1235 (40pin flex cable) LA-2770 (Mipi34-ARM20-adapter) LA-2114 (34pin Half-size-Cable
LA-3522	ACC-PTSERIAL-ETM 7-8	LA-1239 (80pin flex cable)

LA-3521 - ACC-PTSERIAL-ETM 1-6

LA-1235 - 40pin Flex Cable

Manufacturer	Samtec
Type	30 AWG Twinax
Impedance	100 Ω
Capacitance	48.9 pF/m
Inductance	0.518 uH/m
Propagation Delay	4.79 ns/m
Shield	Braid, Silver Plated Copper
Dielectric	FEP
Jacket	PVC
Length	53 cm

Manufacturer	Lauterbach GmbH
Connector 34pin	Samtec, FTSH-117-01-L-DV-EJ-SLA
Connector 20pin	Standard 2.54mm, pin header with shroud
PCB	FR4

LA-2114 - 34pin Half Size Cable

Manufacturer	Yamaichi
Type	Flat ribbon cable
Impedance	75 Ω
Capacitance	22.5pF/ft.
Inductance	0.317µH/ft.
Propagation Delay	1.7 ns/ft.
Conductor	30 AWG
Insulation Material	PVC
Contact Current Rating	0.5A
Contact Resistance	30 mΩ
Length	45mm
Pitch	1.27mm x 1.27mm

LA-1239 - 80pin Flex Cable

Manufacturer	Samtec
Type	30 AWG Twinax
Impedance	100 Ω
Capacitance	48.9 pF/m
Inductance	0.518 uH/m
Propagation Delay	4.79 ns/m
Shield	Braid, Silver Plated Copper
Dielectric	FEP
Jacket	PVC
Length	53 cm

Electrical Characteristics

Serial Port 0

VCCSENSE	1.2-3.3V
Vendor IO	1.2-3.3V
JTAG	1.2-3.3V
TRIGIN	1.2-3.3V
TRIGOUT	1.2-3.3V
Synchronous reference clock (bit clock)	CML output, fraction of lane speed (1/1, 1/2, 1/10, 1/11, 1/12, 2/25, 1/20, 1/22, 1/24, 1/25, 1/30, 1/34, 1/40, 1/50), max. 6.25GHz

Serial Port 1

VCCSENSE	1.2-3.3V
TRIGIN	1.2-3.3V
TRIGOUT	1.2-3.3V
Asynchronous reference clock (oscillator clock)	output HCSL (10MHz-500MHz) or input LVDS/LVPECL/HCSL 100MHz-500MHz
Synchronous reference clock (bit clock)	CML output, fraction of lane speed (1/1, 1/2, 1/10, 1/11, 1/12, 2/25, 1/20, 1/22, 1/24, 1/25, 1/30, 1/34, 1/40, 1/50), max. 6.25GHz

Absolute Maximum Ratings

VCCSENSE	0-3.6V
I/O (JTAG,TRIG,Vendor)	0-3.6V

Switching Characteristics

Lane speed	625Mbps-8000Mbps 9000Mbps-12500Mbps

Timing Characteristics

Maximum Lane Skew

Lane skew	10UI

Additional Data

PCB

Impedance	100 Ω differential
Er	4.7
Material	FR4

Device Connector

Samtec	40 pin, ERF8-020-01-L-D-RA-L-TR 80 pin, ERF8-040-01-L-D-RA-L-TR

Simulation Data Request

Connector	www.samtec.com
Flex cable	support@lauterbach.com
Receiver XC7K325T-FFG900, GTX	www.xilinx.com
PCB	support@lauterbach.com

PowerTrace Serial

Models of the FPGA and the connector are required to simulate the input stage of PowerTrace Serial. Please contact Xilinx and Samtec for device simulation models:

Xilinx FPGA: Kintex7 XC7K320T-3, FF900, GTX-transceiver
Samtec connector: ERF8 series, 80pin, right-angle

Dimensions

PowerTrace Serial: **https://www.lauterbach.com/size/si_3520.pdf**

Adapter for 1lane PCIe-slot: **https://www.lauterbach.com/size/si_3527.pdf**

Adapter for 4lane PCIe-slot: **https://www.lauterbach.com/size/si_3524.pdf**

Adapter for 8lane PCIe-slot: **https://www.lauterbach.com/size/si_3525.pdf**

Adapter for Mini-PCIe: **https://www.lauterbach.com/size/si_3526.pdf**

Adapter for AGBT: **https://www.lauterbach.com/size/si_3556.pdf**

Adapter for RH850: **https://www.lauterbach.com/size/si_3561.pdf**

80pin flex extension cable: **https://www.lauterbach.com/size/si_1239.pdf**

PCIe-Gen4 Preprocessor: **https://www.lauterbach.com/size/si_3529.pdf**

Connector Layout

SerialPort 0

The Serial Port 0 is designed for HSSTP trace ports. The use of the flex extension cable (LA-1235) let the Serial Port 0 follow the ARM-HSSTP standard. The debug signals will be routed to the 34pin debug connector on the top of the PowerTrace Serial module and can be accessed for debug cable.

The Serial Port 0 will also be used for non-HSSTP targets. In this case additional adapters could be necessary.

SerialPort 1

The Serial Port 1 is designed to support multiple interfaces. The function of the signals change depending on the targets trace port. The pin-out of this connector should be seen as Lauterbach proprietary. If you want to connect non-supported data sources please contact the Lauterbach support.

Logic Analyzer Probe (only PowerTrace Serial V1)

The IProbe timing analyzer module is a part of the PowerTrace Serial (PTS) units. It is a subset of the TRACE32 timing analyzer solutions (PowerProbe and PowerIntegrator). It offers a minimum of timing capture capabilities and is not intended to replace or substitute these units. Just a simple signal qualifier can be used to trigger the analyzer. There is no external clock source and there is no complex trigger capability to qualify input signals. However, IProbe offers transient recording which allows an optimal usage of the available trace memory space.

The IProbe offers either a 17 bit digital data trace or a 7 * 12 bit subsequent analog input trace. Depending on if a timing trace or an analog trace is required, either a timing input probe or an analog input probe must be connected to the trace port of the PTS unit. If the required probe is attached, IProbe works as a digital or an analog trace module.

The timing probe basically consists on a high speed buffer for 17 signals, whereas the analog probe is equipped with a 7 channel AD-Converter.

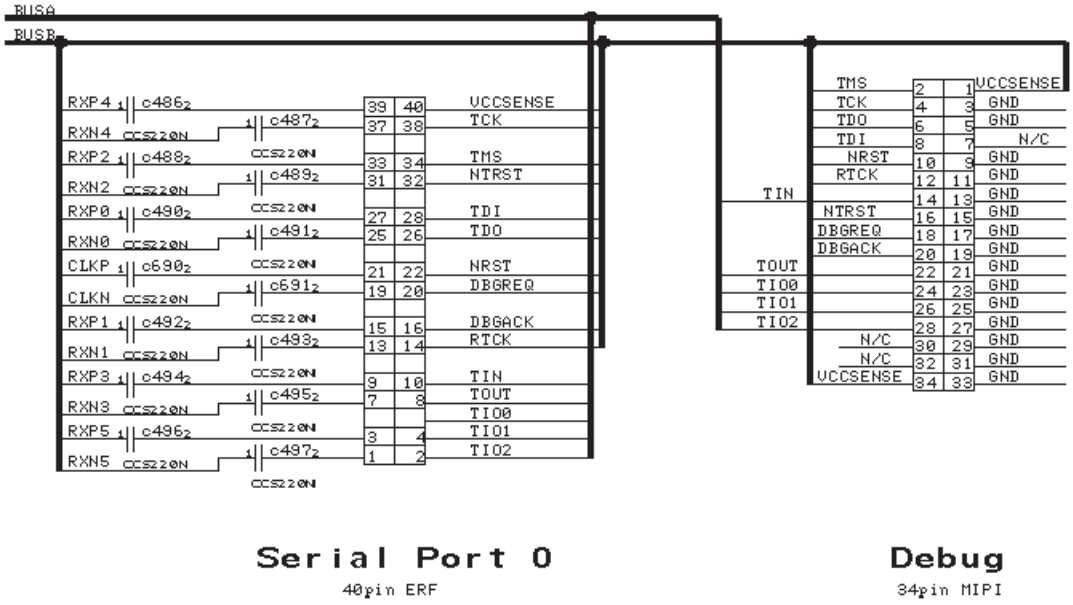
There is a timestamp unit with a resolution of 5ns.

Timing or analog trace date can easily be correlated to the regular program trace. The features of the protocol analysis can also be used.

Aux Port V1 (only PowerTrace Serial V2)

tbd

The 34pin debug connector should be seen as extension to Serial Port 0. The ARM-HSSTP standard defines a target connector which includes debug signals and trace signals. The debug signals are not used by the PowerTrace Serial, but they can be accessed via the MIPI-conform 34pin connector on the top of the module.



	The pinout of the SerialPort0-connector does not match the target trace connector pinout! The necessary flex extension cable will change the signal order to match the HSSTP standard. The correct target connector pinout for ARM-HSSTP can be found here: **https://www.lauterbach.com/adetmhsstp.html**