VX1000 Manual

Version 5.1 | English

Imprint

Vector Informatik GmbH
Ingersheimer Straße 24
D-70499 Stuttgart

The information and data given in this user manual can be changed without prior notice. No part of this manual may be reproduced in any form or by any means without the written permission of the publisher, regardless of which method or which instruments, electronic or mechanical, are used. All technical information, drafts, etc. are liable to law of copyright protection.

© Copyright 2026, Vector Informatik GmbH. All rights reserved.

Contents

1 Introduction **12**

1.1 About this User Manual 13

1.1.1 Warranty 14

1.1.2 Registered Trademarks 14

1.2 Important Notes 15

2 VX1000 System Overview **16**

2.1 Overview 17

2.2 Supported Microcontrollers 19

2.3 Features 20

2.4 VX1000 Application Driver and VX1000If 21

2.4.1 Usage in Development Software 21

2.4.2 Usage in Production Software 21

2.5 Impact on ECU Safety 23

2.6 XCP Slaves 24

2.6.1 General Information 24

2.6.2 XCP Protocol Parameters 25

2.6.3 Supported XCP Commands 26

3 VXtools **28**

3.1 VXupdate 29

3.1.1 General Information 29

3.1.2 Menu Bar 29

3.1.3 Toolbar 31

3.1.4 VX Device List 32

3.1.5 VX Configuration Selector 33

3.1.6 Image List 34

3.1.7 Flash Update 34

3.1.8 Status Bar 35

3.2 VXconfig 36

3.2.1 General Information 36

3.2.2 Menu Bar 37

3.2.3 Toolbar 40

3.2.4 Device List 41

3.2.5 Status Bar 42

3.3 VXconsole 43

3.3.1 General Information 43

3.3.2 Toolbar 44

3.3.3 LEDs ... 45
3.3.4 Message Window ... 46
3.3.5 Command Bar ... 46
3.3.6 Status Bar ... 47
3.4 HowTo ... 48
3.4.1 VXtools Setup ... 48
3.4.2 VX Device First Steps ... 49
3.4.3 VX device IP Address Configuration ... 50
3.4.4 FPGA and Firmware Update ... 53
3.4.5 Write VXparameter File to VX device ... 54
3.4.6 VXtools Automation ... 56

4 Base Modules ... 57

4.1 VX1060 Serial Base Module ... 59
4.1.1 Connectors ... 59
4.1.2 LEDs ... 61
4.1.3 Technical Data ... 65
4.1.4 Accessories ... 66
4.2 VX1134 Base Modules ... 67
4.2.1 VX1134B ... 67
4.2.2 VX1134C ... 76
4.2.3 Accessories ... 85
4.3 VX1135 Base Modules ... 86
4.3.1 VX1135A ... 86
4.3.2 VX1135C ... 96
4.3.3 VX1135D ... 106
4.3.4 VX1135E ... 116
4.3.5 VX1135F ... 126
4.3.6 Accessories ... 136
4.4 VX1152 Video Interface ... 137
4.4.1 General Information ... 137
4.4.2 Lossless Compression ... 139
4.4.3 CRC Algorithm ... 140
4.4.4 Connectors ... 141
4.4.5 Vehicle Battery Only (Closed System) ... 144
4.4.6 LEDs ... 145
4.4.7 Video on VXmodule60 ... 148
4.4.8 Automotive Ethernet on VNmodule60 (optional) ... 165
4.4.9 Device Stacking ... 167
4.4.10 Standby Operation ... 168
4.4.11 Technical Data ... 169

4.4.12 Accessories ... 170
4.5 VX1161 Multi Base Module ... 171
4.5.1 General Information ... 171
4.5.2 Inserting or Removing VX1161 Cards ... 172
4.5.3 VX1161.01A Base ... 174
4.5.4 VX1161.11 Power Supply ... 175
4.5.5 VX1161.22A Host Uplink (obsolete) ... 178
4.5.6 VX1161.22B Host Uplink ... 184
4.5.7 VX1161.22C Host Uplink ... 190
4.5.8 VX1161.31A Serial ... 196
4.5.9 VX1161.32B HSSL ... 200
4.5.10 VX1161.32C HSSL2 ... 204
4.5.11 VX1161.41A 6xCAN ... 208
4.5.12 VX1161.41B 6xCAN 1xFR ... 212
4.5.13 VN1150 CAN/LIN/UART Interface ... 216
4.5.14 VN5130 Ethernet Interface ... 217
4.5.15 VX1161.51A/.52A 2xTAP FPDLINK3 TI954/TI953 ... 219
4.5.16 VX1161.51B 4xRX FPDLINK3 TI954/TI954 ... 244
4.5.17 VX1161.51C/.52C 2xTAP GMSL2 MX9296A/MX9295A ... 255
4.5.18 VX1161.51D/.52D 4xRX GMSL2 MX9296 ... 285
4.5.19 VX1161.52G 2xTAP FPDLINK4 TI9702/TI971 ... 300
4.5.20 VX1161.52H 2xTAP GMSL2 MX96718A/MX96717 ... 331
4.5.21 VX1161.52K 2xTAP APHY VA7044/VA7031 ... 363
4.5.22 VX1161.52L 2xTAP GVIF CXD4966/CXD4967 ... 386
4.5.23 VX1161.51/52 Bandwidth Limits ... 407
4.5.24 VXconfig for VX1161.51/.52 ... 410
4.5.25 Accessories ... 427

5 PODs ... 428

5.1 VX1452 HSSL POD (obsolete) ... 431
5.1.1 General Information ... 431
5.1.2 Connectors ... 432
5.1.3 LEDs ... 433
5.1.4 Technical Data ... 433
5.1.5 Target Interface Connectors ... 434
5.2 VX1453 HSSL POD ... 440
5.2.1 General Information ... 440
5.2.2 Connectors ... 441
5.2.3 LEDs ... 442
5.2.4 Technical Data ... 442
5.2.5 Evalboard Extension Kit ... 443

5.2.6 Target Interface Connectors 444
5.3 VX1454.01 HSSL POD 456
5.3.1 General Information 456
5.3.2 Connectors 457
5.3.3 LEDs 458
5.3.4 Technical Data 458
5.3.5 Evalboard Extension Kit 459
5.3.6 Target Interface Connectors 460
5.4 VX1461B HSSL POD PCIe 463
5.4.1 General Information 463
5.4.2 Connectors 464
5.4.3 LEDs 465
5.4.4 Technical Data 465
5.4.5 Target Interface Connectors 466
5.5 VX1463.02 HSSL POD AUR+1xRIF 469
5.5.1 General Information 469
5.5.2 Connectors 470
5.5.3 LEDs 471
5.5.4 Technical Data 471
5.5.5 Target Interface Connectors 472
5.6 VX1463.20 HSSL POD AUR+2xRIF 475
5.6.1 General Information 475
5.6.2 Connectors 476
5.6.3 LEDs 477
5.6.4 Technical Data 477
5.6.5 Target Interface Connectors 478
5.7 VX1511.04 HSCT/RHSIF/Zipwire Repeater 481
5.7.1 General Information 481
5.7.2 Connectors 482
5.7.3 LEDs 482
5.7.4 Technical Data 482
5.7.5 Target Interface Connectors 484
5.8 VX1522.01 GMSL2 POD LVDS+DAP 486
5.8.1 General Information 486
5.8.2 Connectors 487
5.8.3 LEDs 488
5.8.4 Technical Data 489
5.8.5 Target Interface Connector 490
5.9 VX1543A Serial POD (Obsolete) 493
5.9.1 General Information 493

5.9.2 Connectors 494
5.9.3 LEDs 494
5.9.4 Technical Data 495
5.9.5 Target Interface Connector 496
5.10 VX1543B Serial POD 502
5.10.1 General Information 502
5.10.2 Connectors 504
5.10.3 LEDs 504
5.10.4 Technical Data 505
5.10.5 Target Interface Connector 506
5.11 VX1543C Serial POD 515
5.11.1 General Information 515
5.11.2 Connectors 516
5.11.3 LEDs 516
5.11.4 Technical Data 517
5.11.5 Target Interface Connector 518
5.12 VX1544A Serial POD 524
5.12.1 General Information 524
5.12.2 Connectors 525
5.12.3 LEDs 526
5.12.4 Technical Data 526
5.12.5 Target Interface Connector 527
5.13 VX1544B Serial POD 535
5.13.1 General Information 535
5.13.2 Connectors 537
5.13.3 LEDs 537
5.13.4 Technical Data 538
5.13.5 Target Interface Connector 539
5.14 VX1544C Serial POD 541
5.14.1 General Information 541
5.14.2 Connectors 543
5.14.3 LEDs 543
5.14.4 Technical Data 544
5.14.5 Target Interface Connector 545
5.15 VX1544D Serial POD 547
5.15.1 General Information 547
5.15.2 Connectors 548
5.15.3 LEDs 548
5.15.4 Technical Data 549
5.15.5 Target Interface Connector 550

5.16 VX1621A XPOD ... 559
5.16.1 General Information ... 559
5.16.2 Connectors ... 560
5.16.3 Technical Data ... 561
5.16.4 Target Interface Connectors ... 562

6 Scripting Language (CASL) ... 564

6.1 Calculation and Scripting ... 565
6.2 Script Execution Flow ... 567
6.3 Configuring CASL Script Execution ... 569
6.4 Device Specific Functions ... 570
6.4.1 SetConfiguration ... 570
6.4.2 WriteIO (Standard Write Mode) ... 573
6.4.3 WriteIO (Read-modify Write Mode) ... 574
6.4.4 ReadIO ... 575
6.4.5 GetMicroseconds ... 576
6.4.6 WaitMicroseconds ... 576
6.4.7 ElementsOf ... 577
6.4.8 PrintString ... 577
6.4.9 PrintChar ... 578
6.4.10 PrintSignedDec ... 578
6.4.11 PrintDec ... 578
6.4.12 PrintHex ... 578
6.4.13 PrintBin ... 579
6.5 Script Templates and Examples ... 579

7 Accessories ... 580

7.1 Adapters ... 582
7.1.1 VX1912 PowerPC JTAG Adapter ... 582
7.1.2 VX1922 x850 JTAG Adapter ... 584
7.1.3 VX1923A XC2000 DAP OCDS-L1 Adapter ... 586
7.1.4 VX1923B TriCore DAP OCDS-L1 Adapter ... 588
7.1.5 VX1926 V850E2 JTAG 20-Pin Adapter ... 590
7.1.6 VX1936 V850E2 JTAG 16-Pin Adapter ... 592
7.1.7 VX1944 EMEM Power Supply ... 594
7.1.8 VX1927A JTAG-PT FlexPCB for VX154x ... 596
7.1.9 VX1927B DAP-PT FlexPCB for VX154x ... 599
7.1.10 VX1927C DAP-IFX FlexPCB for VX154x (obsolete) ... 602
7.1.11 VX1927D Zipwire/HSCT FlexPCB for VX154x ... 605
7.1.12 VX1927E DAP-IFX FlexPCB for VX154x (obsolete) ... 607

7.1.13 VX1927H DAP-IFX FlexPCB for VX154x 610
7.1.14 VX1927J DAP-IFX FlexPCB for VX154x 613
7.1.15 VX1903.01A DAP Flex-PCB for XPOD 616
7.1.16 VX1903.01B DAP Flex-PCB for XPOD 617
7.2 EEK Heads 618
7.2.1 VX1902.02A EEK Head Aurora (Infineon) 618
7.2.2 VX1902.02B EEK Head AUR PCB (Infineon) 620
7.2.3 VX1902.03A EEK Head Aurora (NXP/Renesas) 622
7.2.4 VX1902.03B EEK Head AUR PCB (NXP/Renesas) 624
7.2.5 VX1902.04A EEK Head Aurora (ARM) 626
7.2.6 VX1902.04B EEK Head AUR PCB (ARM) 628
7.3 CAN Cables 630
7.3.1 VX1361C CAN Cable LEMO/D-SUB 0.75m 630
7.3.2 VX1361D CAN Cable LEMO/D-SUB 0.75m 631
7.3.3 VX1362A CAN Cable LEMO 1.5m 632
7.3.4 VX1365A CAN Cable LEMO 5m 633
7.3.5 VX1362B CAN Cable LEMO/Banana Plugs 1.5m 634
7.3.6 CANcable 2Y 635
7.4 FAKRA Cables 636
7.4.1 VX1382D/E/F/G FAKRA Cables 636
7.4.2 VX1382H FAKRA Extention 0.1m Cable 637
7.4.3 VX1382M.01 FAKRA Panelmount F0N100N0F (U.FL) 638
7.5 HSSL Cables 639
7.5.1 VX137x HSSL Cables 639
7.6 HSSL2 Cables 642
7.6.1 VX1381 HSSL2 Cables 642
7.7 Power/Sync Cables 645
7.7.1 Vector Power Supply 12V/1.5A 645
7.7.2 Vector Power Supply ODU Mini-Snap 646
7.7.3 Desktop Type Power Supply 649
7.7.4 Cable Vehicle Input <> ODU Mini-Snap 650
7.7.5 Power Cable Molex Mini-Fit 4mm² 650
7.7.6 Cable Banana Plug <> Binder 3-pin 651
7.7.7 Power Sync Cable 2xBinder <> Banana Plugs 652
7.7.8 Cable Banana Plug <> ODU Mini-Snap 653
7.7.9 Car Power Supply Cable 12V with Binder 654
7.7.10 Cable Binder 3pol Connector with Pigtail 655
7.7.11 Vector SYNCcableXL 656
7.8 Serial Cables 657
7.8.1 VX134x Serial Cables 657

7.8.2 VX1311 Serial2 Cables 659
7.9 Ethernet Cables 661
7.9.1 AEcable 2Y Family 661
7.9.2 Ethernet Cable 10Gbps 0.5m 667
7.9.3 Ethernet Cable 10Gbps 2m 668
7.9.4 Ethernet Cable 10Gbps 7.5m 669
7.9.5 Ethernet Cable Robust 10Gbps 0.5m 670
7.9.6 Ethernet Cable Robust 10Gbps 2.0m 671
7.9.7 Ethernet Cable Robust 10Gbps 7.5m 672
7.10 FlexRay Cables 673
7.10.1 FRcable AB 673
7.10.2 FR/CANcable 2Y 674
7.11 Mounting Equipment 675
7.11.1 Fix Kit 32mm Device 675
7.11.2 Fix Kit Large Device 676
7.11.3 Serial POD Enclosure Gen2 677
7.11.4 VSH Cable Guard 216 678
7.11.5 VSH Cable Guard 216 Clip 681
7.11.6 VSH Connection Kit 19" 682
7.11.7 VSH Connection Kit Horizontal 684
7.11.8 VSH Connection Kit Vertical 685
7.11.9 VSH Equipment Foot Kit 686
7.11.10 VSH Mounting Flange Kit 687

8 Hardware Integration Guide 688

8.1 Introduction 689
8.1.1 Getting Started 689
8.2 POD Usage Guideline 690
8.3 Interfaces and Bus Systems 691
8.3.1 Introduction 691
8.3.2 AUR (Aurora) 692
8.3.3 DAP 694
8.3.4 HSCT/RHSIF/Zipwire 696
8.3.5 HSSL 698
8.3.6 JTAG 699
8.3.7 RIF (LVDS Radar Interface) 701
8.3.8 SWD 703
8.3.9 SYSTEM 705
8.4 Layout Recommendations 707
8.4.1 Introduction 707

8.4.2 PCB Layer Stack-Up 707
8.4.3 Impedance Matching 708
8.4.4 Routing of Traces 709
8.4.5 Reference Planes 711
8.4.6 Differential Signals 712
8.5 Galvanic Isolation and Ground Concept 713
8.5.1 Understanding the VX1000 Ground Concept 713
8.5.2 Grounds of a VX1000 Setup 714
8.5.3 Grounds inside a VX Base Module 718
8.5.4 Grounds Inside a POD 719
8.5.5 Disconnecting an ECU from its Power Supply 720
8.6 Mechanical Integration 724
8.6.1 Introduction 724
8.6.2 Basic Thoughts about Integrating a POD 725
8.6.3 Heat Dissipation of a POD 726
8.7 Wiring 728
8.7.1 Cable Pigtail 728
8.7.2 VX1381 HSSL2 Pigtail Installation 729
8.8 Advanced Topics 732
8.8.1 Aurora Reference Clock 732

9 Important Notes - Details 736

9.1 Safety Instructions and Hazard Warnings 737
9.1.1 Proper Use and Intended Purpose 737
9.1.2 Hazards 738
9.1.3 Warning of Hot Surface 738
9.1.4 Electrostatically Sensitive Components (ESD) 738
9.2 Disclaimer 740
9.3 Disposal of Vector Hardware 741

10 Wichtige Hinweise 742

10.1 Sicherheits- und Gefahrenhinweise 743
10.1.1 Sach- und bestimmungsgemäßer Gebrauch 743
10.1.2 Gefahren 744
10.1.3 Warnung vor heißen Oberflächen 744
10.1.4 Elektrostatisch gefährdete Bauelemente (ESD) 744
10.2 Haftungsausschluss 746
10.3 Entsorgung von Vector Hardware 747

1 Introduction

In this chapter you find the following information:

1.1 About this User Manual **13**
1.1.1 Warranty 14
1.1.2 Registered Trademarks 14
1.2 Important Notes **15**

1.1 About this User Manual

Conventions

In the two following charts you will find the conventions used in the user manual regarding utilized spellings and symbols.

Style	Utilization
bold	Blocks, surface elements, window and dialog names of the software. Accentuation of warnings and advices. **[OK]** Push buttons in brackets **File|Save** Notation for menus and menu entries
`Source Code`	File name and source code.
Hyperlink	Hyperlinks and references.
<CTRL>+<S>	Notation for shortcuts.

Symbol	Utilization
	This symbol calls your attention to warnings.
	Warning of damages by electrostatic discharge (ESD = Electrostatically Sensitive Device).
	Warning of hot surface! Do not touch without appropriate precautions.
	Here you can obtain supplemental information.
	Here you can find additional information.
	Here is an example that has been prepared for you.
	Step-by-step instructions provide assistance at these points.
	Instructions on editing files are found at these points.
	This symbol warns you not to edit the specified file.

1.1.1 Warranty

Restriction of warranty

We reserve the right to change the contents of the documentation and the software without notice. Vector Informatik GmbH assumes no liability for correct contents or damages which are resulted from the usage of the documentation. We are grateful for references to mistakes or for suggestions for improvement to be able to offer you even more efficient products in the future.

1.1.2 Registered Trademarks

Registered trademarks

All trademarks mentioned in this documentation and if necessary third party registered are absolutely subject to the conditions of each valid label right and the rights of particular registered proprietor. All trademarks, trade names or company names are or can be trademarks or registered trademarks of their particular proprietors. All rights which are not expressly allowed are reserved. If an explicit label of trademarks, which are used in this documentation, fails, should not mean that a name is free of third party rights.

- Windows, Windows 7, Windows 8.1, Windows 10, Windows 11 are trademarks of the Microsoft Corporation.

1.2 Important Notes

WARNING!
We provide our important notes and safety instructions in several languages, including Englisch (EN) and German (DE). For more details, see the relevant section:

- EN: Important Notes - Details
- DE: Wichtige Hinweise

2 VX1000 System Overview

In this chapter you find the following information:

2.1 Overview **17**

2.2 Supported Microcontrollers **19**

2.3 Features **20**

2.4 VX1000 Application Driver and VX1000If **21**

2.4.1 Usage in Development Software 21

2.4.2 Usage in Production Software 21

2.5 Impact on ECU Safety **23**

2.6 XCP Slaves **24**

2.6.1 General Information 24

2.6.2 XCP Protocol Parameters 25

2.6.3 Supported XCP Commands 26

2.1 Overview

At a glance

The VX1000 system is a scalable solution with top performance for your measurement and calibration tasks. It can be used in the vehicle – both in the interior and in the engine compartment – on test benches and in the laboratory.Especially when developing ADAS ECUs, this allows you to control raw data captured by high-resolution radar sensors in combination with XCP data, e. g. object/tracking lists.

The system forms the interface between the ECU and a measurement and calibration tool such as CANape.For high data throughput with minimal impact on ECU run-time, data is accessed over the microcontroller-specific data trace and debug ports.

The VX1000 Base Module is connected to the PC over XCP on Ethernet, an OEM-independent ASAM standard (www.asam.net) that is widely used in the automotive industry. The VX1000 measurement hardware is connected to the ECU via a POD (Plug-On device).Depending on the available microcontroller interface, either the data trace or a copying method can be used to acquire measurement data.

Figure 1: Modular concept

Modular concept

The modular concept of the VX1000 system allows connecting to different microcontrollers and consists of the following components:

- VX1000 Base Modules
- VX13xx Cables (HSSL/HSSL2/Serial Cables)
- VX14xx/VX15xx PODs (Plug-On Devices)

To support longer distances between the VX1000 Base Module and the target ECU, e. g. under the hood in a car, the Base Module is connected to the ECU via a POD and a cable that is available in different lengths. PODs and cables are designed to withstand rough automotive environmental conditions.

Figure 2: VX1000 Family examples

XPOD

With the XPOD, the VX1000 system also offers a standalone measurement and calibration interface without the need of a VX1000 Base Module.

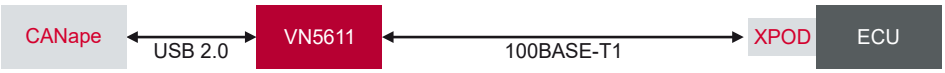


Figure 3: Example with XPOD

Advantages

Advantages of the VX1000 system are:

- Powerful measurement and calibration access to internal ECU data with maximum transmission rates
- Very small POD to connect to the ECU debug interface
- Easy and quick integration into the ECU software
- No impact on ECU run-time with data trace measurement method
- Interface to numerous development tools by third-party suppliers via the standardized ASAM protocol XCP on Ethernet
- Special functions for engine controllers such as Calibration Wake-Up and Calibration RAM Supply

2.2 Supported Microcontrollers

Reference
The VX1000 supports a broad range of automotive microcontrollers from different vendors. For the latest list, visit www.vector.com/vx1000-targets.

2.3 Features

This section contains a selection of features of the VX1000 system. Please contact Vector VX support (VXsupport@vector.com) for detailed information on all features.

Overview

- Very high measurement data throughput of more than 100 Mbyte/s for XCP data and radar raw data when using data trace for measurement
- Measurement of fast signal cycles (7 µs for data trace, 50 µs for copying method)
- Measurement configurations of more than 200,000 signals can be processed
- Precise generation of DAQ time stamps in the ECU
- ECU cold start measurement (First Loop DAQ)
- Calibration of ECU parameters without address range limitations
- Calibration memory page switching
- Automatic overlays when calibrating parameters in flash memory
- Stimulation or bypassing with short latency times
- 100 Mbps/1 Gbps/10 Gbps Ethernet connection to the PC
- Galvanically isolated power supply with wide input voltage range
- POD power supply via the VX1000 Base Module
- Optional: Flash programming, even for “brain-dead” ECUs
- Optional: FlexRay and CAN channels via XL Driver Library for CANape/CANalyzer/CANoe and custom applications
- PC tools for easy configuration and for software updates
- Target debugging over XCP

2.4 VX1000 Application Driver and VX1000If

2.4.1 Usage in Development Software

VX1000 Measurement and Calibration access to the ECU requires not only the appropriate VX1000 hardware but also the target specific VX1000 Application Driver (in short: VX1000 AppDriver) which must be integrated into the ECU software.

WARNING!
The VX1000 AppDriver has been developed for test and development purposes but not for use in production software. Activating the VX1000 AppDriver in production software is always prohibited!

The VX1000 AppDriver is available through the Vector Download Center free of charge. This download also contains the VX1000If, which encapsulates access from the application to the VX1000 AppDriver, and the technical reference required for the integration into the ECU development software.

WARNING!
For VX1000If and the VX1000 AppDriver provided this way, Product-Specific Special Terms apply, which can be downloaded here:
http://www.vector.com/TC-VX1000If-VX1000AppDriver

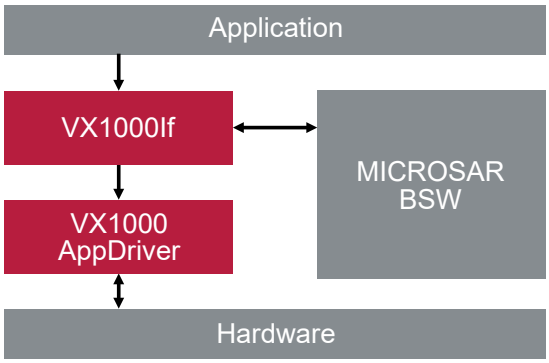


Figure 4: Application components

2.4.2 Usage in Production Software

For production usage, the VX1000 AppDriver must be disabled completely to avoid any runtime impact of the module. This is required as the VX1000 AppDriver has been developed for test and development purposes only but not for use in production software.

To disable VX1000 AppDriver in production software different approaches are possible. It remains the sole responsibility of the ECU integrator to choose an appropriate variant:

- Use VX1000If, which is provided free of charge together with the VX1000 AppDriver. It remains the responsibility of the ECU integrator to ensure correct function of the provided software, including VX1000If. As Vector does not perform any version management on thus delivered software, there will be no issue reporting and Vector rejects to take over any liability requests.
- Request a managed MICROSAR Classic VX1000If software delivery from Vector through embedded@vector.com or a respective request in the MICROSAR Classic eQuestionnaire. Depending on your needs, VX1000If can either be delivered as a standalone package (VX1000If Mini-CSP) or it can be added to your MICROSAR Classic package.
- Encapsulate VX1000If and the VX1000 AppDriver in a way that disables these components during production usage.
- Remove VX1000 AppDriver and VX1000If from production software.

WARNING!
Activating the VX1000 AppDriver in production software is always prohibited!

2.5 Impact on ECU Safety

The VX1000 hardware provides a very efficient way for measurement and calibration access to ECUs via XCP on Ethernet. It features an external and direct read/write access (DMA) to the ECU and/or its microcontroller (MCU), without involving the CPU or application, thus minimizing impact on ECU runtime.

Unlike an XCP driver embedded in the ECU application, the XCP slave is implemented in the VX1000 hardware itself. The hardware is accompanied by the VX1000 AppDriver, which is integrated into the application and provides a software API for measurement and calibration purposes.

The ECU or its MCU is accessed by the VX1000 via hardware interfaces, typically debug ports of the MCU. The VX1000 can communicate with the VX1000 AppDriver via DMA, but can also directly read or modify memory contents such as calibration parameters, special function registers and even non-volatile memory.

WARNING!
Some MCUs do not provide means to restrict this DMA, meaning that the VX1000 hardware can access all memory areas. This can lead to unpredictable ECU behaviour if used incorrectly! This risk exists regardless of whether the VX1000 AppDriver has been safely integrated into the application.

WARNING!
It is strongly recommended that the safety concept of the ECU is considered before use, and that any use of calibration tools and the VX1000 hardware is in accordance with this safety concept!

2.6 XCP Slaves

2.6.1 General Information

XCP slaves

The VX1000 system provides 2 XCP slaves on configurable TCP/UDP ports. The **Fully-Featured XCP slave** supports CAL, DAQ, PGM, PAG and STIM for the ECU. The **Limited XCP slave** supports CAL, DAQ and STIM for the ECU.

Network ports

The network ports of all XCP slaves can be configured.

Note
It is recommended to use the default TCP and UDP ports for the XCP slaves.
Fully-Featured XCP slave (TCP/UDP): 5555
Limited XCP slave (TCP/UDP): 5554

2.6.2 XCP Protocol Parameters

Reference
All settings described in this section can be configured using VXconfig.

XCP protocol parameters

Some XCP protocol parameters can be configured to adapt the VX1000 system to Measurement tools with limited XCP protocol support.

MAX_CTO
MAX_DTO

The XCP MAX_CTO and XCP MAX_DTO parameters configure the maximum size of CTO and DTO packets supported by the XCP slave.

SERV_TEXT

XCP SERV_TEXT messages are used by the slaves to give additional information or user guidance if problems or errors are encountered. If a tool cannot process SERV_TEXT messages, this feature can be disabled.

DTO ID

For the DTO ID field type only relative ODT numbers are supported by the VX1000 XCP slaves.

DAQ Overload Indication

A DAQ Overload can be indicated in the MSB of the DTO PID or as additional XCP event or not at all. The recommended setting is XCP event.

Priority Threshold

This is not exactly a XCP protocol parameter but rather a parameter to either reduce Ethernet network load or enable low network latencies.

The DTOs of all DAQ lists bound to XCP events with a priority greater than the Priority Threshold are immediately sent to the Measurement tool. Whereas DTOs of DAQ lists bound to events with low priority are collected for up to 20 ms to reduce network load.

Normal measurement data should be measured with low priority while DAQ data which is used as bypassing input should be handled at high priority.

2.6.3 Supported XCP Commands

Command	STD	CAL	DAQ	PGM	PAG	Remark
CONNECT	•					Mode=1 used for coldstart data acquisition
DICONNECT	•					
GET_STATUS	•					
SYNC	•					

Command	STD	CAL	DAQ	PGM	PAG	Remark
GET_COMM_MODE_INFO	•					
GET_ID	•					
SET_REQUEST						
GET_SEED	•					
UNLOCK	•					
SET_MTA	•					
UPLOAD	•					
SHORT_UPLOAD	•					
BUILD_CHECKSUM	•					Uses CRC16_CCITT
TRANSPORT_LAYER_CMD						
USER_CMD						

Command	STD	CAL	DAQ	PGM	PAG	Remark
DOWNLOAD		•				
DOWNLOAD_MAX						
DOWNLOAD_NEXT						
SHORT_DOWNLOAD		•				
MODIFY_BITS						

Command	STD	CAL	DAQ	PGM	PAG	Remark
SET_CAL_PAGE					•	Page 0 = FLASH, Page 1 = RAM
GET_CAL_PAGE					•	

Command	STD	CAL	DAQ	PGM	PAG	Remark
GET_PAG_PROCESSOR_ INFO						
GET_SEGMENT_INFO						
GET_PAGE_INFO						
SET_SEGMENT_MODE						
GET_ SEGMENT_MODE						
COPY_CAL_PAGE						

Command	STD	CAL	DAQ	PGM	PAG	Remark
CLEAR_DAQ_LIST			•			
SET_DAQ_PTR			•			
WRITE_DAQ			•			
WRITE_DAQ_MULTIPLE			•			Highly recommended to speed up measurement setup
SET_DAQ_LIST_MODE			•			
GET_DAQ_LIST_MODE			•			
START_STOP_DAQ_LIST			•			
START_STOP_SYNCH			•			

Command	STD	CAL	DAQ	PGM	PAG	Remark
GET_DAQ_CLOCK			•			
READ_DAQ						
GET_DAQ_PROCESSOR_INFO			•			
GET_DAQ_RESOLUTION_INFO			•			
GET_DAQ_LIST_INFO						
GET_DAQ_EVENT_INFO						

Command	STD	CAL	DAQ	PGM	PAG	Remark
FREE_DAQ			•			
ALLOC_DAQ			•			
ALLOC_ODT			•			
ALLOC_ODT_ENTRY			•			

Command	STD	CAL	DAQ	PGM	PAG	Remark
PROGRAM_START				•		
PROGRAM_CLEAR				•		
PROGRAM				•		
PROGRAM_RESET				•		

Command	STD	CAL	DAQ	PGM	PAG	Remark
GET_PGM_PROCESSOR_INFO						
PROGRAM_PREPARE				•		Used for some controllers
PROGRAM_FORMAT						
PROGRAM_NEXT						
PROGRAM_MAX						
PROGRAM_VERIFY						

3 VXtools

In this chapter you find the following information:

3.1 VXupdate **29**
3.1.1 General Information 29
3.1.2 Menu Bar 29
3.1.3 Toolbar 31
3.1.4 VX Device List 32
3.1.5 VX Configuration Selector 33
3.1.6 Image List 34
3.1.7 Flash Update 34
3.1.8 Status Bar 35

3.2 VXconfig **36**
3.2.1 General Information 36
3.2.2 Menu Bar 37
3.2.3 Toolbar 40
3.2.4 Device List 41
3.2.5 Status Bar 42

3.3 VXconsole **43**
3.3.1 General Information 43
3.3.2 Toolbar 44
3.3.3 LEDs 45
3.3.4 Message Window 46
3.3.5 Command Bar 46
3.3.6 Status Bar 47

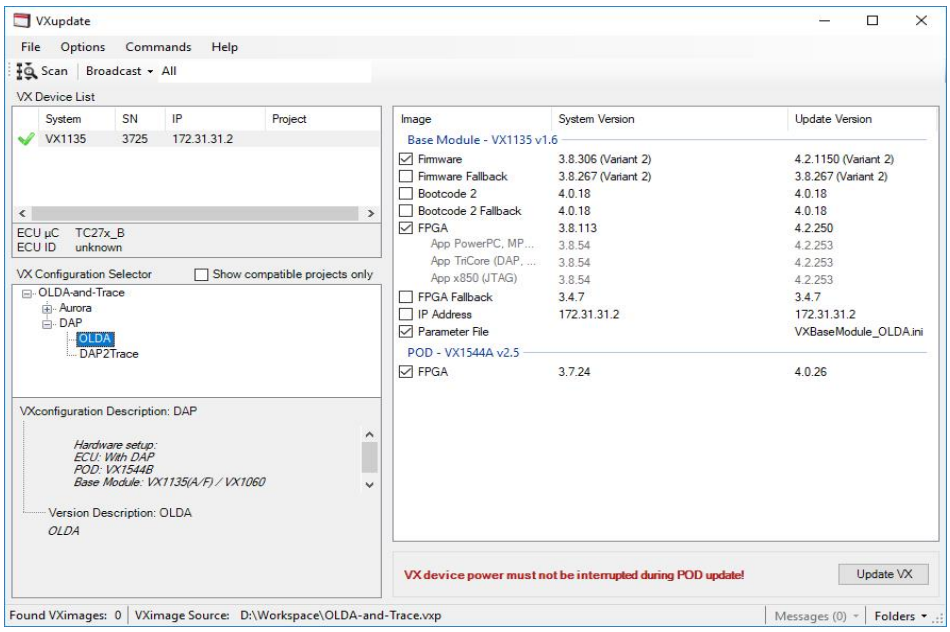
3.4 HowTo **48**
3.4.1 VXtools Setup 48
3.4.2 VX Device First Steps 49
3.4.3 VX device IP Address Configuration 50
3.4.4 FPGA and Firmware Update 53
3.4.5 Write VXparameter File to VX device 54
3.4.6 VXtools Automation 56

3.1 VXupdate

3.1.1 General Information

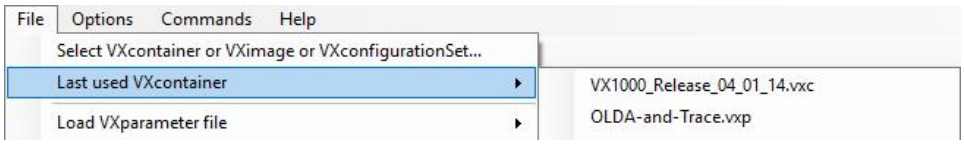
VXupdate

VXupdate is the tool for updating VX1000 systems by flashing VXimage files and VXparameter files. It can be found in **Start | Programs | Vector VXtools | VXupdate** after installation.



3.1.2 Menu Bar

File menu

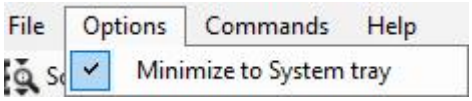


► **Select VXcontainer or VXimage or VXconfigurationSet…**
This dialog allows selecting an individual .vxi **VXimage**, a .vxc **VXcontainer** or a .vxp **VXconfigurationSet**. The VXimages and the parameter sets, which are found and apply to the selected VX device, will be displayed in the Image List view. If a **VXconfigurationSet** is selected the available VXconfigurations will be listed in the VX Configuration Selector (see section VXconfiguration Selector).

Note
File types for the VX1000 system are:
.vxp VXconfigurationSet containing one or more VXconfigurations consisting of VXimages and VXparameter files - can also be used to modify the IP address of a VX device.
.vxc VXcontainer with a set of VXimage files for the VX1000 system.
.vxi VXimage file for the VX1000 system containing an FPGA or Firmware image for a Base Module or a POD.

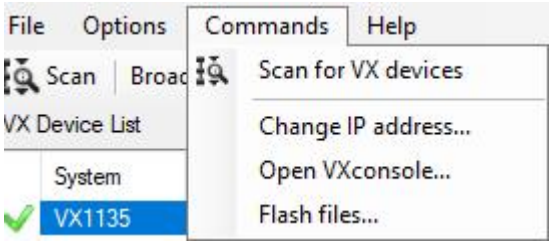
- **Last used VXcontainer**
  Lists already used VXcontainers or VXconfigurationSets. The user can select and load one of them.
- **Load VXparameter file**
  Loads a VXparameter set into the selected VX device.
- **Exit**
  Closes VXupdate.

Options menu



- **Minimize to System tray**
  If this option is enabled, an icon will be added to the System tray.
  When VXupdate is minimized, it will be removed from the Taskbar and can be restored via the System tray icon.

Commands menu



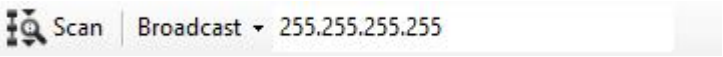
- **Scan for VX devices**
  Scans the network for VX devices. Found devices are shown in the Device List view.
- **Change IP address…**
  Changes the IP address of the selected VX device.
- **Open VXconsole…**
  Opens VXconsole to display VX1000 system debug messages of the selected VX device. This information is helpful for Vector support requests.
- **Flash files…**
  Starts the flash update process (see section Flash Update)

Help menu

- **Screenshot | Complete Desktop**
  Takes a screenshot of the whole Desktop and shows the location of the created png file.
- **Screenshot | VXupdate**
  Takes a screenshot of the VXupdate window and shows the location of the created png file.
- **Support mail…**
  Creates a debug information file and opens a new E-Mail window in your default E-Mail client addressed to the Vector VX support (VXsupport@vector.com). Please also include an issue description in the message body.
- **Folders | User Data**
  Opens the Windows Explorer with the VXupdate User Data folder.
- **Folders | Program Data**
  Opens the Windows Explorer with the VXupdate program folder.
- **Manual**
  Opens the VX1000 Manual PDF.
- **About…**
  Shows details of the VXupdate version.

3.1.3 Toolbar

Toolbar

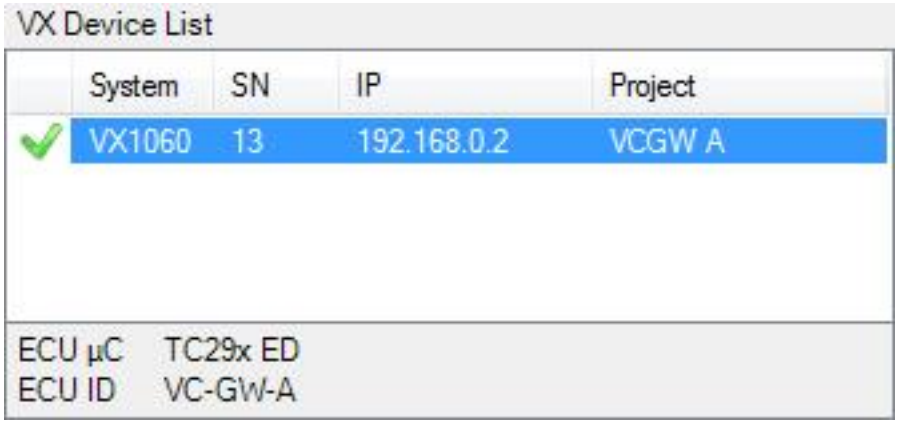


- **Scan**
  Scans the network for VX devices. Found devices are shown in the Device List view.
- **Broadcast**
  Sets the broadcast address for VX device scans. This parameter limits the scan to a subnet (e.g. to a single network adapter) or covers all network adapters when a value of 255.255.255.255 is used. Reasonable values can be found in the drop-down menu. If only one specific device should be found, its IP address can be entered.

3.1.4 VX Device List

General information

The Device List view displays all found VX devices in the network:



Device Status

- **Status**
  The following device states are possible, after a scan has been executed:

Status	Description / Possible problems
✔	▶ VX device in normal operation. ▶ VX device contains valid VXimages for all of its hardware components.

⚠	▶ VX device contains an incompatible or outdated Firmware or FPGA Image for at least one of its hardware components.
	  ▶ Update the FPGA and Firmware Images (see section FPGA and Firmware Update).
	▶ VX1000 Fallback Firmware is active.
	  ▶ Update the FPGA and Firmware Images (see section FPGA and Firmware Update).

⊘	▶ VX device has invalid network settings. ▶ Correct the network settings of the VX1000 system (see section VX1000 System IP Configuration). ▶ XCP slave is already in use by another XCP master (Measurement tool).

Device Info

- **System**
  Displays the type of the found VX device.
- **SN**
  Displays the serial number of the found VX device.
- **IP**
  Displays the IP address of the found VX device.
- **Project**
  Displays the Project ID of the POD connected to the VX device.

ECU Info

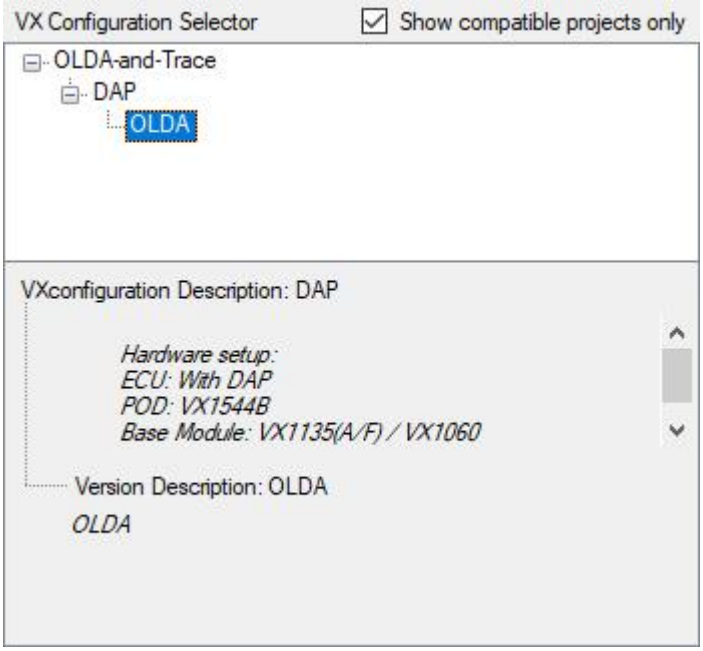
- **ECU µC**
  Displays the detected microcontroller type of the ECU connected to the selected VX device.

- **ECU ID**
  Displays the ECU-ID string defined in the VX1000 AppDriver of the ECU connected to the selected VX device.

3.1.5 VX Configuration Selector

Configuration Selector

When a VXconfigurationSet (.vxp) is loaded – via the File menu or via drag and drop – the available configurations and their descriptions are shown in this view. By clicking on one of the configuration sets the corresponding images and parameters are loaded into the Image List view (update version column) and are available for flash programming into the VX device.



3.1.6 Image List

General information

The Image List view displays the VXimages of the selected VX device which can be selected and flashed.

Image	System Version	Update Version
Base Module - VX1135 v1.4		
☑ Firmware	4.2.1055 (Variant 2)	4.1.971 (Variant 2)
☐ Firmware Fallback	3.8.267 (Variant 2)	3.8.267 (Variant 2)
☐ Bootcode 2	4.0.18	4.0.18
☐ Bootcode 2 Fallback	4.0.18	4.0.18
☑ FPGA	4.2.240	4.1.222
App PowerPC, MP...	4.2.232	4.1.206
App TriCore (DAP, ...	4.2.232	4.1.206
App x850 (JTAG)	4.2.232	4.1.206
☐ FPGA Fallback	3.4.7	3.4.7
☐ IP Address	172.31.31.2	172.31.31.2
☑ Parameter File		VXBaseModule_OLDA.ini
POD - VX1544B v2.5		
☐ FPGA	4.0.26	4.0.26

List view

- **Image**
  This column shows the compatible images and parameters of the loaded VXconfigurationSet, VXcontainer or VXimage for the selected VXdevice sorted by Base Module and POD. Also the Base Module and POD type with hardware revision are shown.
- **System Version**
  This column shows the current VXimage version of the selected VX device.
- **Update Version**
  This column shows the version of the VXimage file available for update.

3.1.7 Flash Update

Flash

VX device power must not be interrupted during POD update! [Update VX]

The **[Update VX]** button starts the update for the selected VX device, applying all selected images and parameters.

WARNING!
The power supply of the VX device must not be interrupted during a POD update procedure. The POD cannot recover from a failed update procedure.

3.1.8 Status Bar

Status bar

- **Found VXimages**
  The number of VXimage files (*.vxi / *.hex / *.mem) inside a .vxc **VXcontainer** or a directory containing VXimage files.
- **VXimage Source**
  Storage location of the found VXimage files.
- **Messages**
  VXupdate status messages will be grouped and available through the Messages drop-down field.
- **Folders | User Data**
  Opens the Windows Explorer with the VXupdate User Data folder.
- **Folders | Program Data**
  Opens the Windows Explorer with the VXupdate program folder.

3.2 VXconfig

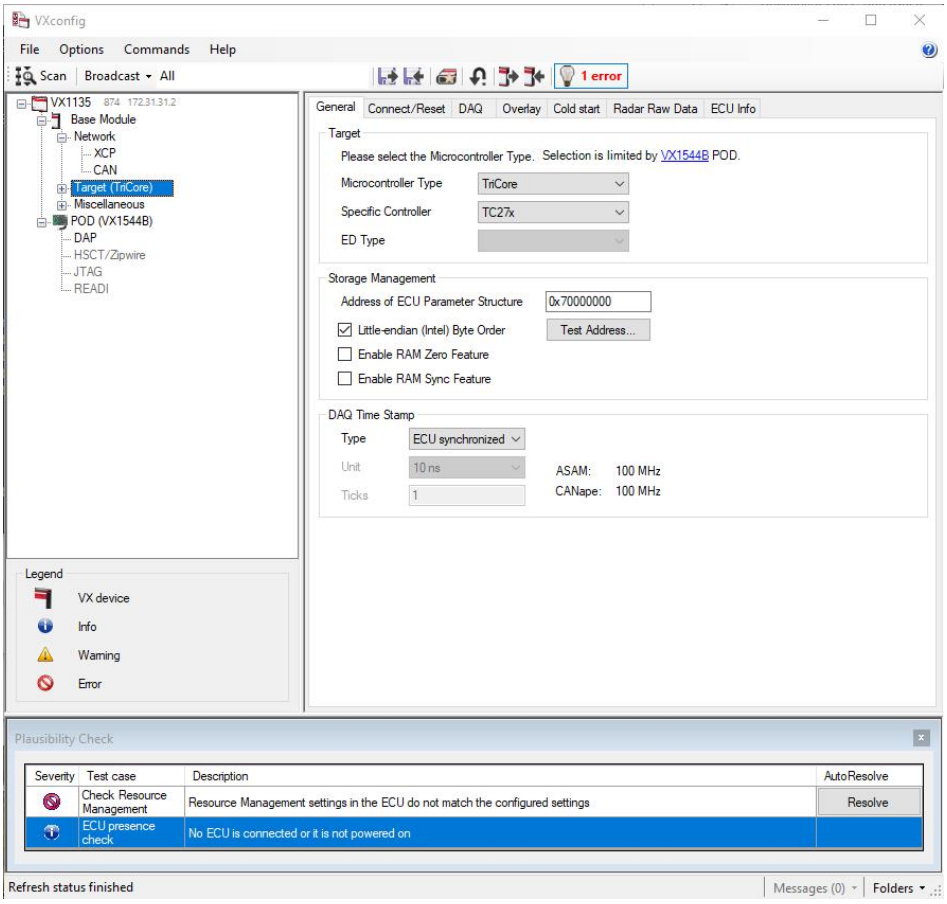
3.2.1 General Information

Configuration tool

VXconfig is the configuration tool for the VX1000 system.

This tool provides access to all VXparameters which control the behavior of the VX device. It also performs Firmware and FPGA updates using VXupdate.

VXconfig can be found in **Start | Programs | Vector VXtools | VXconfig** after installation.



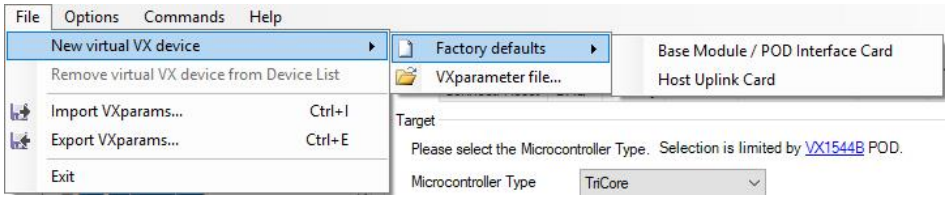
Note

The parameters shown in VXconfig are a working copy of the parameters of the selected VX device. These parameters can be edited in VXconfig without immediately affecting the VX device. If the parameters shall be written into the VX device the commands **Write configuration to VX1000 Base Module** or **Write configuration to POD** have to be used.

Importing VXparameters or POD parameters also just loads the parameters into VXconfig and does not write the parameters into the VX device immediately.

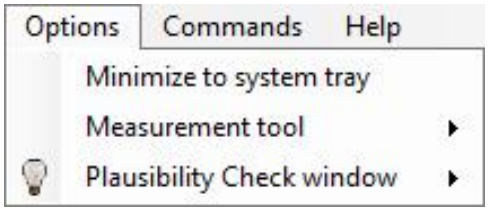
3.2.2 Menu Bar

File menu



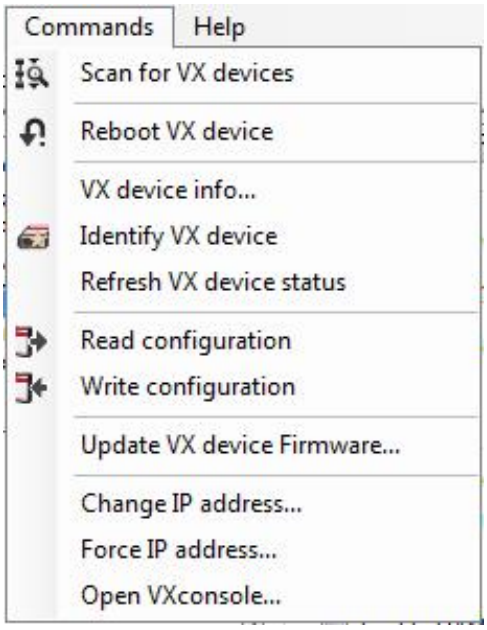
- **New virtual VX device…**
  Inserts a new virtual VX device into the device list, either by using VXparameter defaults or by importing VXparameters from a VXparameter file.
- **Remove virtual VX device from Device List**
  The selected virtual device is deleted.
- **Import VXparams…**
  Imports VXparameters from a VXparameter file into the selected VX device.
- **Export VXparams…**
  Saves the current configuration of the selected VX device into a VXparameter file.
- **Exit**
  Closes VXconfig.

Options menu



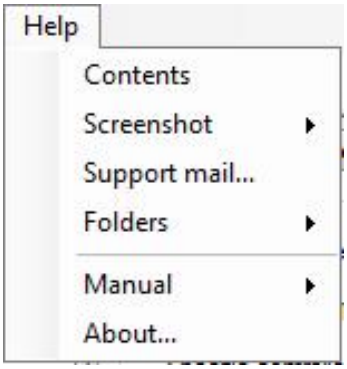
- **Minimize to System tray**
  Enables an icon for VXconfig in the System tray. When VXconfig is minimized, it will be removed from the Taskbar and can be restored via the System tray icon.
- **Measurement tool**
  Selects the Measurement tool that will be taken into account for the plausibility check.
- **Plausibility check window**
  Hides or shows the plausibility check results. Selects the position of the plausibility check window.

Commands menu



- **Scan for VX devices**
  Scans the network for VX devices. Found devices are shown in the Device List view.
- **Reboot VX device**
  Restarts the selected VX device.
- **VX device info…**
  Shows details of the selected VX device.
- **Identify VX device**
  Flashes the PWR LED of the VX device for 3 seconds. This feature can be used for visual identification of a VX1000 Base Module.
- **Refresh VX device status**
  Refreshes the status of the VX device (Scan, TCP, XCP, POD and ECU and executes the Identify VX device command.
- **Read configuration**
  Reads the current configuration from POD and Base Module into VXconfig.
- **Write configuration**
  Writes the current configuration from VXconfig into POD and Base Module. A restart is performed afterwards.
- **Update VX device Firmware...**
  Opens VXupdate so that the Firmware images of the selected VX device can be updated.
- **Change IP address…**
  Changes the IP address of the selected VX device
- **Force IP address…**
  Sets the IP address of a VX device even if it cannot be found by a scan e.g. due to a Firewall problem. The VX device must be specified by Base Module type, Serial Number, and Config XCP port.
- **Open VXconsole …**
  Opens VXconsole to display VX1000 system debug messages of the selected VX device. This information is helpful for Vector support requests.

Help menu



- **Contents**
  Opens the help file.
- **Screenshot | Complete Desktop**
  Takes a screenshot of the whole Desktop and shows the location of the created png file.
- **Screenshot | VXconfig**
  Takes a screenshot of the VXconfig window and shows the location of the created png file.
- **Support mail…**
  Creates a debug information file and opens a new E-Mail window in your default E-Mail client addressed to the Vector VX support (VXsupport@vector.com). Please also include an issue description in the message body.
- **Folders | User Data**
  Opens the Windows Explorer with the VXconfig User Data folder.
- **Folders | Program Data**
  Opens the Windows Explorer with the VXconfig program folder.
- **Manual**
  Opens the installed VX1000 Manual.
- **About…**
  Shows details of the VXconfig version.

3.2.3 Toolbar

Toolbar

- **Scan**

  Scans the network for VX devices. Found devices are shown in the Device List view.

- **Broadcast**

  Sets the broadcast address for VX device scans. This parameter limits the scan to a subnet (e.g. to a single network adapter) or covers all network adapters when a value of 255.255.255.255 is used. Reasonable values can be found in the drop-down menu. If only one specific device should be found, its IP address can be entered.

- **Import VXparams**

  Opens an existing VXparameter file for editing.

- **Export VXparams**

  Saves the current configuration as VXparameter file.

- **Identify VX device**

  Flashes the state LED of the VX device for 3 seconds. This feature can be used for visual identification of a VX1000 Base Module.

- **Reboot VX device**

  Restarts the selected VX device.

- **Read from VX device**

  Reads the configuration from the selected VX1000 Base Module into VXconfig.

- **Write to VX device**

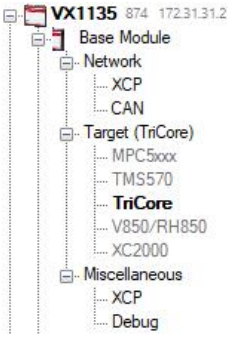
  Writes the current configuration from VXconfig to the selected VX1000 Base Module. The device will be restarted afterwards.

- **Check settings**

  Displays the Plausibility Check window below or right to the VXconfig main window.

3.2.4 Device List

General information



The Device List view displays all detected VX devices in the network after a scan. The tree structure provides a grouped access to all available VXparameters. Each top-level entry of the Device List view contains a device status symbol followed by Base Module type, serial number and network IP address. Tree entries marked in bold font indicate modified VXparameters that have not been saved or written to the VX device.

Device Status

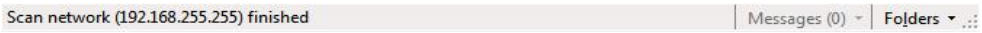
The following device states are possible and shown as a symbol on the left of the VX1000 Base Module type:

Status	Description / Possible problems
	► VX device in normal operation. ► VX device contains valid VXimages for all of its hardware components.
	► VX device contains an incompatible or outdated FPGA or Firmware Image for at least one of its hardware components. ► Update the FPGA and Firmware Images (see section FPGA and Firmware Update). ► VX1000 Fallback Firmware is active. ► Update the FPGA and Firmware Images (see section FPGA and Firmware Update).
	► VX device has invalid network settings. ► Correct the network settings of the VX1000 system (see section VX device IP Address Configuration on page 50). ► XCP slave is already in use by another XCP master (Measurement tool). ► Check the VX device Status.

- **Base Module type**
  The Base Module type (e.g. VX1132) is printed next to the Device State symbol.
- **Serial Number**
  The Serial Number of the VX1000 Base Module is shown next to the Base Module type.
- **IP address**
  The IP address of the VX1000 Base Module is shown after the Serial Number.

3.2.5 Status Bar

Status bar



- **Command Status Information**
  The Status bar shows information about the currently or last executed command.
- **Messages**
  VXconfig status messages will be grouped and available through the Messages drop-down field.
- **Folders | User Data**
  Opens the Windows Explorer with the VXconfig User Data folder.
- **Folders | Program Data**
  Opens the Windows Explorer with the VXconfig program folder.

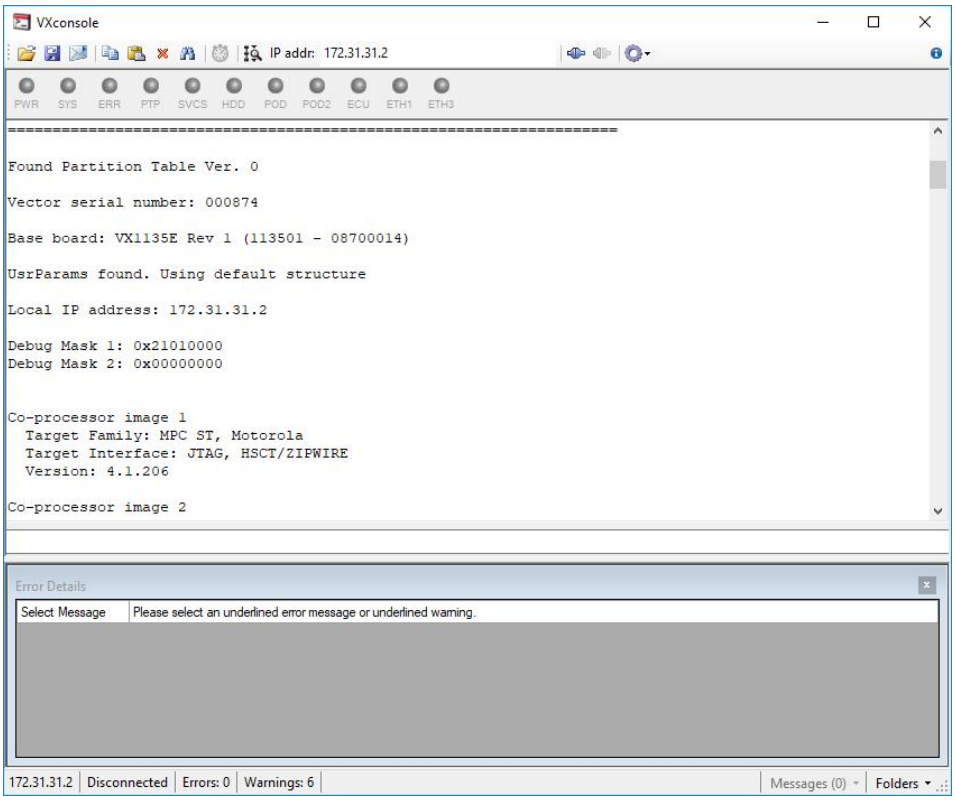
3.3 VXconsole

3.3.1 General Information

VXconsole

VXconsole is a tool for displaying system messages issued by a VX device. It can be started from both VXconfig and VXupdate via **Commands | Open VXconsole**.

The VXconsole window consists of a toolbar, LEDs, a message window, a command bar, and a status bar.



Note
The logged messages which were generated since the last restart of the VX device are helpful information for Vector support requests.

3.3.2 Toolbar

Toolbar

- **Open Log-File**
  Opens an existing log file for viewing.
- **Save Log-File**
  Saves the current logged messages as text/log file.
- **Send logged messages to Vector VX support**
  Copies the logged messages into a text file and opens an E-Mail window of the default E-Mail client addressed to the Vector VX support (VXsupport@vector.com). Please also include an issue description in the message body.
- **Copy logged messages**
  Copies the logged messages to the clipboard.
- **Insert clipboard into message window**
  Inserts content of the clipboard into the message window.
- **Clear logged messages**
  Clears the message window.
- **Search in messages**
  Opens the search bar. Also accessible via shortcut STRG+F.
- **Insert marker**
  Inserts a text line marker with the current time.
- **Scan**
  Scans the network for VX devices to determine the IP address of a specific VX device.
- **Connect**
  Opens a VXconsole connection to the VX device with the given IP address.
- **Disconnect**
  Disconnects from the VX device.
- **Options**
  Provides the following options to control the behavior of VXconsole:
  - **Minimize to System tray**
    Minimizes the VXconsole window and shows the VXconsole icon in the Windows System tray.
  - **Auto connect on startup**
    Automatically connect to the last used IP address when VXconsole is started.
  - **Presence check**
    Automatically reconnect when the connection to the VX device breaks down (like if the VX device is rebooted).
  - **Keep window always on top**
    VXconsole window is always shown on top of all other windows.

- **Show LED state**
  The state of the connected VX devices LEDs is shown in VXconsole (see section LEDs on page 45).
- **Show Message details**
  Opens the **Error Details** window below the message window to show detailed information on error messages.
- **Log to VXdeviceMessages.log**
  Write the debug messages into a ring buffer log file.
- **Enable logging time stamps**
  Enables timestamps for the debug messages.
- **Enable infinite logging**
  No ring buffer used for logging. All messages are logged as long as memory is available.

► **About**

Opens the details of the VXconsole version.

3.3.3 LEDs

Reference
The LED images below the toolbar mirror the state of the selected Base Module. Please find the specific LED descriptions in the according section (see Base Modules on page 57).

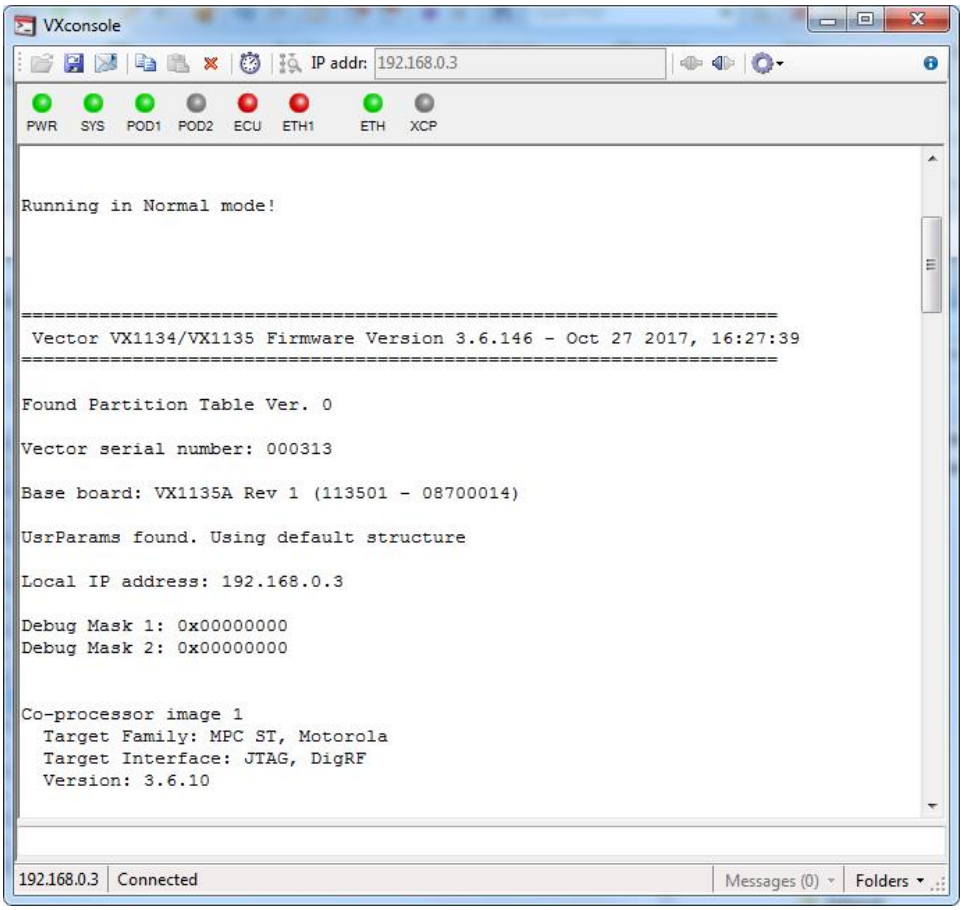
3.3.4 Message Window

The message window contains primarily the logged messages of the VX device in addition to connection status messages. Those messages are color-coded and have the following meaning:

Text Color	Description
Blue	Connection status messages or markers generated by VXconsole
Black	Standard/normal messages of the VX device
Orange	Warning messages of the VX device
Red	Error/Critical messages of the VX device

For underlined messages detailed information is shown in the **Error Details** window when clicking on the message.

Message window



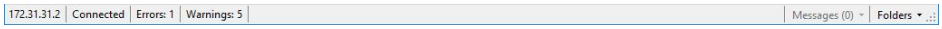
3.3.5 Command Bar

Command bar

The Command bar allows sending of command strings to the VX device. At the moment, only the “replay” command is supported.

3.3.6 Status Bar

Status bar



- **IP Address**
  IP address of the VXdevice to which the last connection (attempt) was made.
- **Connection Status**
  Status of the current connection between VXconsole and the VX device.
- **Errors**
  Number or Error messages in the message window. Clicking here will jump to the next error message. This can be used to comfortably cycle through all the error messages within the message buffer for inspection.
- **Warnings**
  Number or Warning messages in the message window. Clicking here will jump to the next warning message. This can be used to comfortably cycle through all the warnings within the message buffer for inspection.
- **Messages**
  VXconsole status messages will be grouped and available through the Messages drop-down field.
- **Folders | User Data**
  Opens the Windows Explorer with the VXconsole User Data folder.
- **Folders | Program Data**
  Opens the Windows Explorer with the VXconsole program folder.

3.4 HowTo

The following subsections contain step-by-step installation instructions and system procedures when working with the VX1000 system.

3.4.1 VXtools Setup

3.4.1.1 Overview

General information

`VXtoolsSetup_<version>_FullSetup.exe` is the setup program for the Vector VXtools containing:

- Firmware and FPGA images for the VX devices
- VXupdate (one-click tool to update images and apply preconfigs)
- VXconfig (expert tool to configure VX devices or make preconfigs)
- VXconsole (tool for monitoring VX devices)
- VXplugins for CANape (seamless integration of VX devices in CANape), also includes flash kernels for target programming via VX devices
- Network interface drivers (for bus interfaces)
- VX1000 manual

Reference
The latest version of VXtools Full Setup is available at www.vector.com/vxtools.

A **Minimal Setup** is also available in the Download Center. This version does not include the VXplugins for CANape or the network interface drivers.

3.4.1.2 Minimum Requirements

Hardware	Ethernet (100 Mbps, 1 Gbps, 10 Gbps recommended)
Operating system	Microsoft Windows 10 Microsoft Windows 11

Note
Please note that **Administrator rights** are required for the installation procedure.

3.4.1.3 Installations Instructions

VXtoolsSetup

The VXtools (VXconfig, VXconsole and VXupdate) can be installed by running the VXtools installation wizard. Please run the `VXtoolsSetup_<version>.exe` and follow the installation dialog.

3.4.2 VX Device First Steps

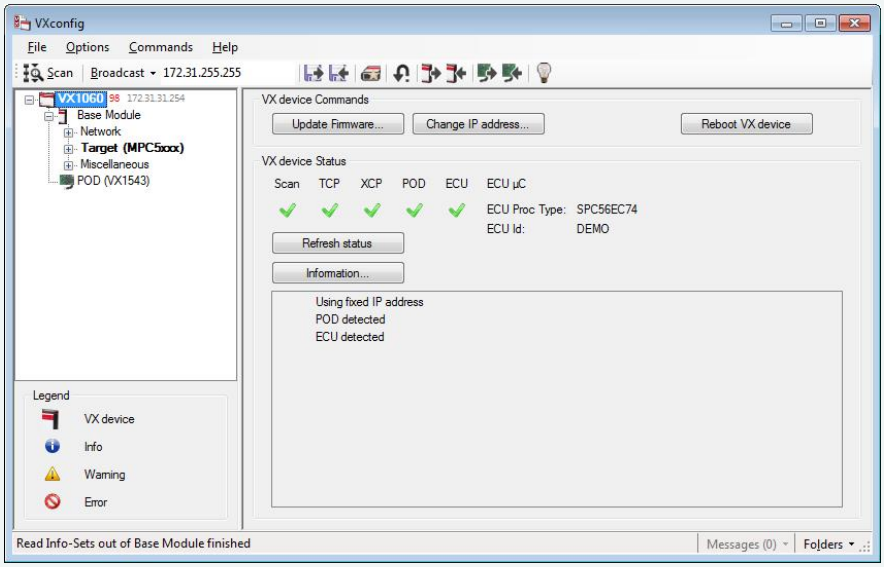
VXconfig

Establishing first contact with a VX device is easily done with VXconfig, which is part of the VXtools package.

Step by Step Procedure

Please follow these steps:

1. Connect an Ethernet cable to the VX device.
2. Connect this cable directly to a PC or to a network switch.
3. Connect the power supply to the connector labeled with “Power”. The Power LED should now be on.
4. Run VXconfig from the Start menu (**Start | Programs | Vector VXtools | VXconfig**).
5. Check if the connected VX device was recognized by VXconfig. It should be listed in the Device List view. Furthermore, the VX device status should display green symbols for Scan status, TCP status and XCP status.



Note

If the IP address of the VX device does not match the local subnet of the PC, it must be adjusted for the VX device to be accessible.

3.4.3 VX device IP Address Configuration

VXconfig

The VX device IP configuration is easily done with VXconfig, which is part of the VXtools package.

There are two ways to change an IP address. If the VX device was recognized by VXconfig, it can be changed using the command Change IP Address. Otherwise Force IP Address has to be used.

Note
The following default IP addresses are set at device delivery:

VX1060/VX113x:
192.168.0.2

VX1161:
- Slot 2: 192.168.0.12
- Slot 3: 192.168.0.13
- Slot 4: 192.168.0.14
- Slot 5: 192.168.0.15
- Slot 6: 192.168.0.16
- Slot 7: 192.168.0.17
- Slot 8: 192.168.0.18

WARNING!
An unused IP address, which matches your local subnet, has to be provided by your network administrator.

3.4.3.1 Change IP Address

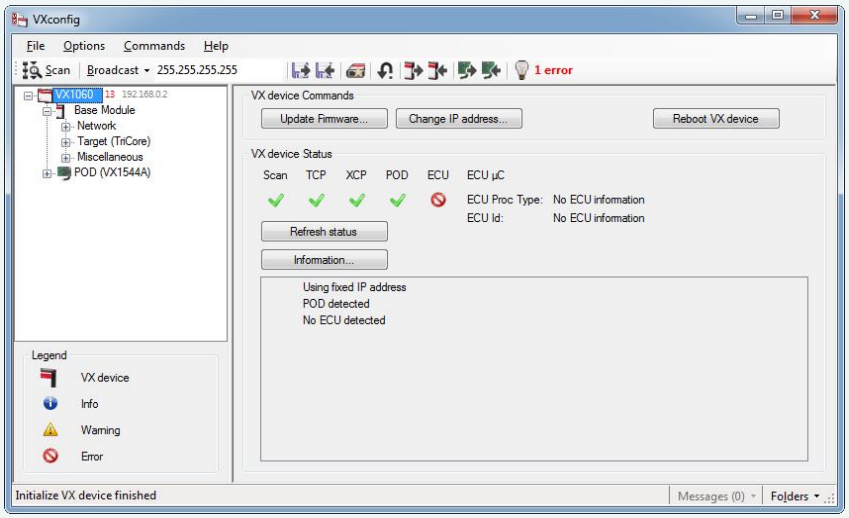
Change IP address

This procedure describes how to change the IP address of a VX device that was found during a Scan with VXconfig.

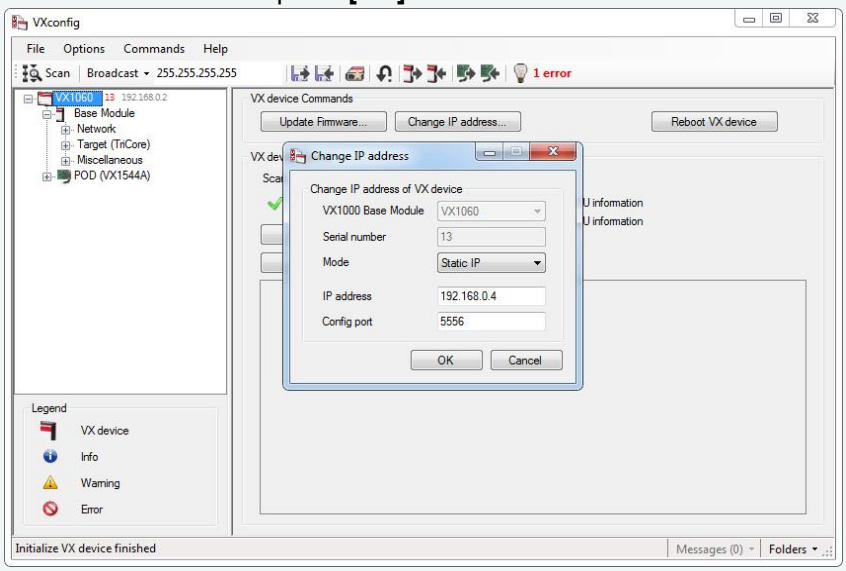
Step by Step Procedure

Please follow these steps:

1. Ensure that the VX device is correctly installed, connected to the PC and running.
2. Run VXconfig from the Start menu (**Start | Programs | Vector VXtools | VXconfig**).
3. Select the VX device whose IP address has to be changed in the Device List view.



4. Execute **Commands | Change IP address…** from the Menu bar to open the IP address change dialog. Enter the new IP address provided by your network administrator and press **[OK]**.



5. If the newly entered IP address resides within a different subnet than the previous address, please press **[Scan]**.

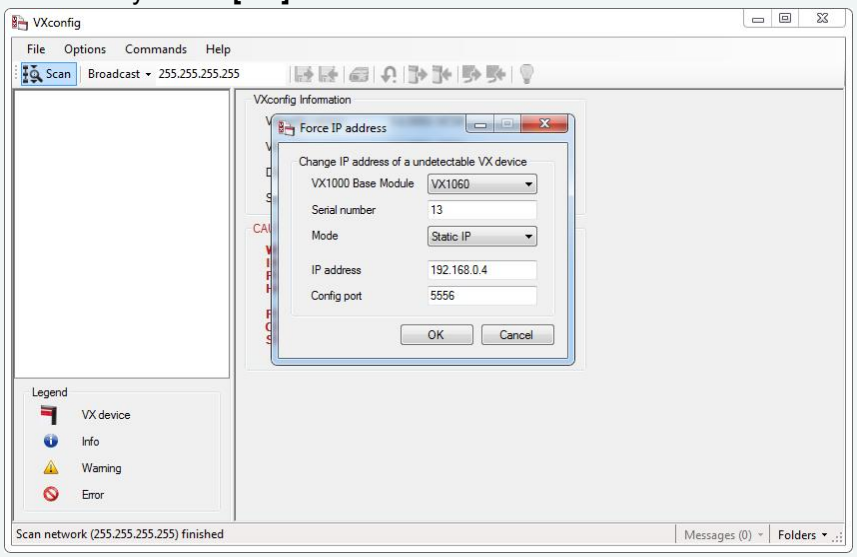
3.4.3.2 Force IP Address

This procedure describes how to change the IP address of a VX device that was not found during a Scan with VXconfig.

WARNING!
An unused IP address, which matches your local subnet, has to be provided by your network administrator.

Step by Step Procedure
Please follow these steps:

1. Ensure that the VX device is correctly installed, connected to the PC and running.
2. Run VXconfig from the Start menu (**Start | Programs | Vector VXtools | VXconfig**).
3. The VX device whose IP address has to be changed is not shown in the Device List view.
4. Execute **Commands | Force IP address…** from the Menu bar to open the IP address change dialog. Select the Base Module type and the serial number of the unrecognized VX device, enter the new IP address provided by your network administrator and change the Config XCP port (default: 5556), if necessary. Press **[OK]** afterwards.



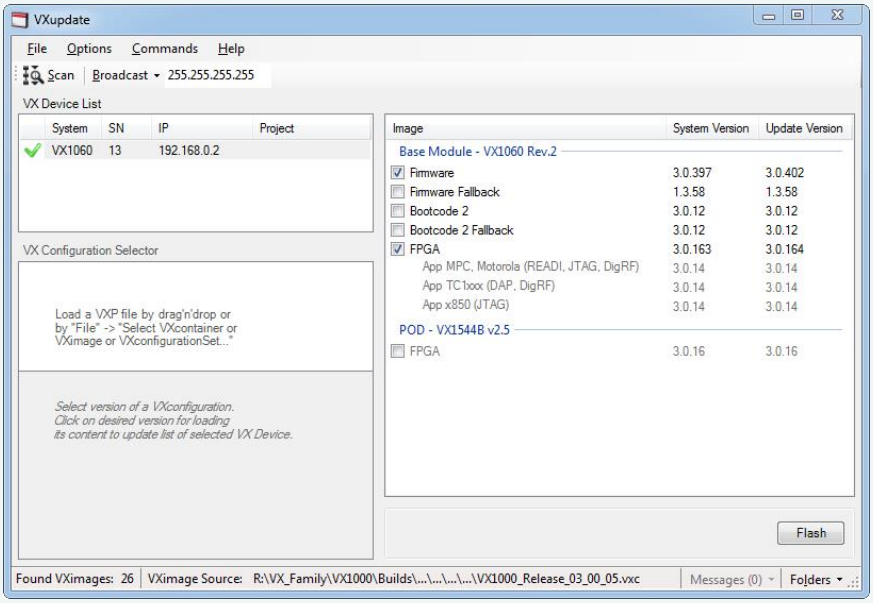
3.4.4 FPGA and Firmware Update

FPGA and Firmware updates of the VX1000 Base Module and the VX14xx/VX15xx PODs are easily done with VXupdate.

Note
The FPGA Image of the POD cannot be updated, if the present version is equal to the update version.

Step by Step Procedure
Please follow these steps:

1. Ensure that the VX device is connected to the PC and running.
2. Run VXupdate from the Start menu (**Start | Programs | Vector VXtools | VXupdate**).
3. Go to menu **File | Select VXcontainer** and select the container with the Firmware and FPGA files for update. Press **[Open]**.
4. Select the VX device that is to be updated, e.g. VX1060 with serial number 13.



5. Select the FPGA and Firmware Images that are to be updated. VXupdate will automatically select all components for which an update is available.
6. Press [Flash] to start the update process.

3.4.5 Write VXparameter File to VX device

The VX1000 Base Module can be easily configured with VXconfig or VXupdate by loading a prepared VXparameter file into the VX1000 Base Module.

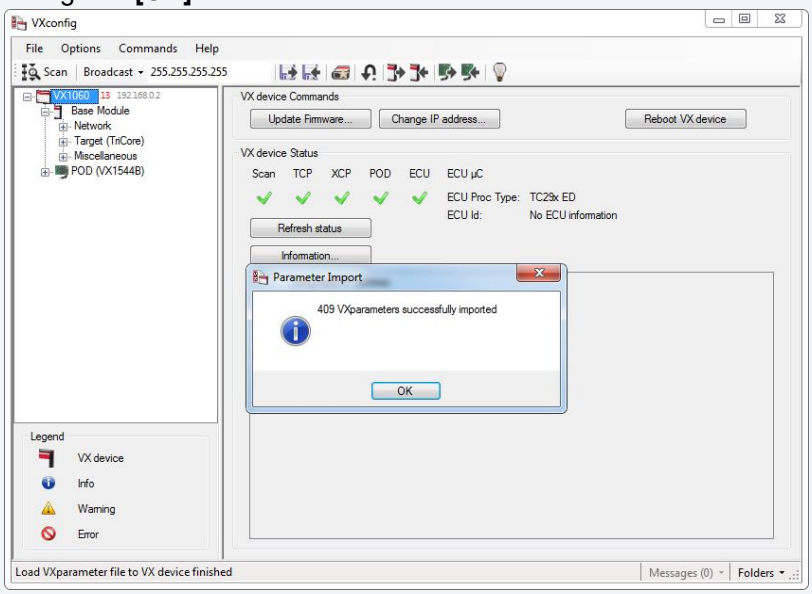
Note
VXupdate is recommended to download a valid VXparameter file without former modification of the parameters. VXconfig has to be used if the VXparameter file contains errors or if modifications have to be made before the parameters are downloaded into the VX device.

VXconfig procedure

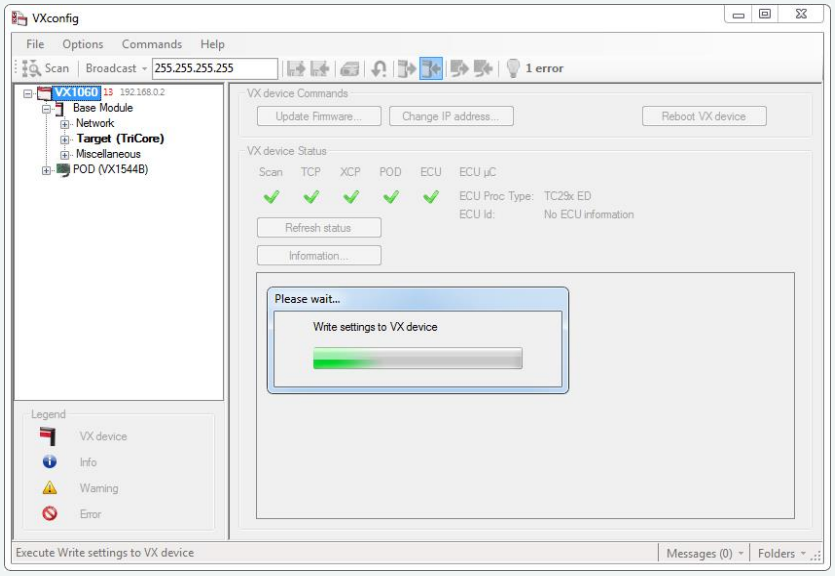
VXconfig allows importing a prepared VXparameter file, reviewing and changing the VXparameter set, correction of invalid VXparameter values and downloading those settings into an available VX1000 system.

Step by Step Procedure
Please follow these steps:

1. Ensure that the VX device is connected to the PC and running.
2. Run VXconfig from the Start menu (**Start | Programs | Vector VXtools | VXconfig**).
3. Select the VX device in the Device List view that shall receive the VXparameter file and open the Base Module page by clicking on the Base Module tree entry.
4. Press **[Import VXparams file…]** on the Base Module page or from the File menu and select the prepared configuration file. To load the settings into VXconfig, press **[Open]** in the file selection window and confirm the import dialog with **[OK]**.



5. To download the configuration into the VX device press **[Write to VX]**. The tool performs a scan once the configuration is written.



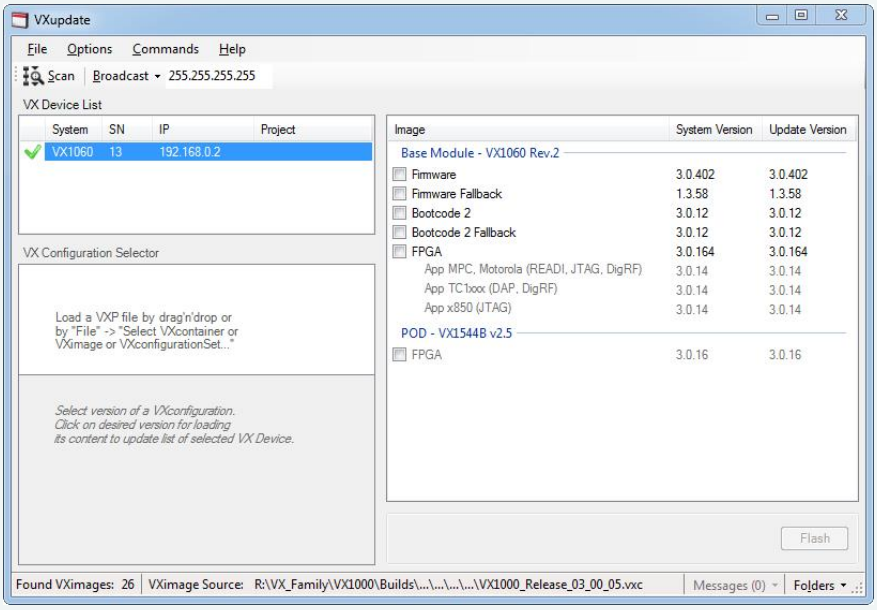
VXupdate procedure

The VXupdate tool allows downloading of a prepared and valid VXparameter file into an available VX device.

Step by Step Procedure

Please follow these steps:

1. Ensure that the VX device is connected to the PC and running.
2. Run VXupdate from the Start menu (**Start | Programs | Vector VXtools | VXupdate**).
3. Select the VX device in the Device List view that shall receive the VXparameter file (e.g. VX1060 with serial number 13).



4. Go to **File | Load VXparameter file | <<filename>>** on the Menu bar and select the prepared VXparameter file.
5. After the VXparameter file was downloaded into the VX device, press **[OK]** to close the “Update Process” dialog, if necessary. The tool performs a Scan afterwards.

3.4.6 VXtools Automation

Each of the VXtools supports a command line interface which can be used for scripted automation. For details on the available command line options, please invoke the tool with parameter -h from a command line.

Note

Calling any of the VXtools from the command line with parameter -h will list the available command line options:

- VXconfig -h
- VXupdate -h
- VXconsole -h

4 Base Modules

In this chapter you find the following information:

4.1 VX1060 Serial Base Module **59**
4.1.1 Connectors 59
4.1.2 LEDs 61
4.1.3 Technical Data 65
4.1.4 Accessories 66

4.2 VX1134 Base Modules **67**
4.2.1 VX1134B 67
4.2.2 VX1134C 76
4.2.3 Accessories 85

4.3 VX1135 Base Modules **86**
4.3.1 VX1135A 86
4.3.2 VX1135C 96
4.3.3 VX1135D 106
4.3.4 VX1135E 116
4.3.5 VX1135F 126
4.3.6 Accessories 136

4.4 VX1152 Video Interface **137**
4.4.1 General Information 137
4.4.2 Lossless Compression 139
4.4.3 CRC Algorithm 140
4.4.4 Connectors 141
4.4.5 Vehicle Battery Only (Closed System) 144
4.4.6 LEDs 145
4.4.7 Video on VXmodule60 148
4.4.8 Automotive Ethernet on VNmodule60 (optional) 165
4.4.9 Device Stacking 167
4.4.10 Standby Operation 168
4.4.11 Technical Data 169
4.4.12 Accessories 170

4.5 VX1161 Multi Base Module **171**
4.5.1 General Information 171
4.5.2 Inserting or Removing VX1161 Cards 172
4.5.3 VX1161.01A Base 174
4.5.4 VX1161.11 Power Supply 175
4.5.5 VX1161.22A Host Uplink (obsolete) 178
4.5.6 VX1161.22B Host Uplink 184
4.5.7 VX1161.22C Host Uplink 190
4.5.8 VX1161.31A Serial 196

4.5.9 VX1161.32B HSSL 200
4.5.10 VX1161.32C HSSL2 204
4.5.11 VX1161.41A 6xCAN 208
4.5.12 VX1161.41B 6xCAN 1xFR 212
4.5.13 VN1150 CAN/LIN/UART Interface 216
4.5.14 VN5130 Ethernet Interface 217
4.5.15 VX1161.51A/.52A 2xTAP FPDLINK3 TI954/TI953 219
4.5.16 VX1161.51B 4xRX FPDLINK3 TI954/TI954 244
4.5.17 VX1161.51C/.52C 2xTAP GMSL2 MX9296A/MX9295A 255
4.5.18 VX1161.51D/.52D 4xRX GMSL2 MX9296 285
4.5.19 VX1161.52G 2xTAP FPDLINK4 TI9702/TI971 300
4.5.20 VX1161.52H 2xTAP GMSL2 MX96718A/MX96717 331
4.5.21 VX1161.52K 2xTAP APHY VA7044/VA7031 363
4.5.22 VX1161.52L 2xTAP GVIF CXD4966/CXD4967 386
4.5.23 VX1161.51/52 Bandwidth Limits 407
4.5.24 VXconfig for VX1161.51/.52 410
4.5.25 Accessories 427

4.1 VX1060 Serial Base Module

4.1.1 Connectors

Figure 5: VX1060 Serial Base Module front side

- **DEBUG • IO**
  Using VX199x Debugger Adapters is no longer supported on this connector. Instead, the VX1060 Base Module now supports the ASAM Standard extension Software Debugging over XCP to allow third-party debugger software to use the VX1060 as debugger hardware to access the target.

Figure 6: VX1060 Serial Base Module back side

- **PWR • SYNC**
  Power is provided through a 3-pin Binder connector (labeled with PWR • SYNC) on the back side of the VX1060 Base Module. The power supply is protected against reverse polarity and over-current.Vector offers different accessories for supplying power with a mating 3-pin female Binder connector (type 711).

  Hardware time synchronization allows the Measurement and Calibration tool to synchronize the timestamps of different devices to a common time base with a precision of 1 µs.For synchronization of VX devices a two-wire connection (signals: Sync, GND) between all devices is required. At each falling edge on the Sync line the VX device generates a timestamp that is provided to the Measurement and Calibration tool via XCP.

Reference
Power supply cables and sync cables can be found in section Accessories on page 580.

ETH

The communication between a Measurement and Calibration tool like CANape and the VX1000 system is implemented by XCP over Ethernet. Therefore the VX1060 must be connected to a common PC network with a network cable (Cat 5 or better). As the Ethernet interface features Auto-MDI(X) a crossover cable or patch cable can be used equivalently.

The following protocols are supported for measurement and calibration:
- XCP on TCP
- XCP on UDP

Ensure that no Firewall on the network or on the PC blocks the traffic of the VX1000 system. The Ethernet connector contains two LEDs (100 and ACT). The description of both indicators is given in the following two tables.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off Y = yellow G = green

100
LED indicating the link speed.

Sequence	Description
□□□□□□□□	Link speed is 10 Mbit/s
G G G G G G G G	Link speed is 100 Mbit/s

ACT
LED indicating the Ethernet activity status.

Sequence	Description
□□□□□□□□	Ethernet link is down (see section VXconsole on page 43)
Y Y Y Y Y Y Y Y	Ethernet link is up
Y □ Y □ Y □ Y □	Ethernet link is up and activity is detected

POD (red or blue)

This connector is used to attach a VX154x Serial POD.

A blue connector isolation indicates, that the connector housing (and therefore the shield of a connected cable) is capacitively coupled to the internal system ground of the VX1060 Serial Base Module (starting with HW-Rev. 5.0).

A red connector isolation indicates, that the connector housing (and therefore the shield of a connected cable) is directly connected to the internal system ground of the VX1060 Serial Base Module (before HW-Rev. 5.0).

4.1.2 LEDs

Overview

On the front side, there are several LEDs that display the state of the Base Module, the attached POD and the target ECU.

The following tables list all available LED sequences and describe the condition at which they will be active.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off R = red G = green

PWR (Power)

Sequence	Description
R R R R R R R R	Device is powered but Firmware not running. The PWR LED will light up red at the beginning of the boot sequence of the Base Module. ► Please contact Vector VX support if the PWR LED stays red for more than 10 seconds. In this case a critical error has occurred.
G G G G G G G G	Firmware loaded successfully.
G □ G □ G □ G □	Response to ”Identify VX device command” (VXconfig)
R □ □ □ □ □ □ □	Standby Mode is active. A cyclical short red pulse on the PWR LED indicates that the Base Module is in Standby Mode. ► Please contact the VX support if the Standby Mode is not left even if the Ethernet Host link is up again for more than 10 seconds.

SYS (System)

Sequence	Description
□ □ □ □ □ □ □ □	Firmware not loaded.
R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Base Module, its system components are initialized. If a component fails to initialize properly, the SYS LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
G G G G G G G G	Device is in normal operating mode

POD

Sequence	Description
☐☐☐☐☐☐☐☐	Device is not in normal operating mode.
R R R R R R R R	POD not detected. After the Base Module boot sequence has been completed successfully, the system searches for an attached POD. As long as no POD is found, the POD LED will stay red. ► Please check if the cable between POD and Base Module is undamaged. ► Please also check if all connectors are correctly locked into their corresponding sockets.
G G G G G G G G	POD in normal operating mode
G G G G R R R R	POD detected but configuration incorrect The POD was detected by the Base Module, but the target interface configuration is incorrect. ► Please open VXconfig and adjust the POD settings according to the target ECU configuration.

ECU

Sequence	Description
☐☐☐☐☐☐☐☐	POD not in normal operating mode
R R R R R R R R	Target microcontroller not detected. The POD was detected by the VX1000 Base Module and is configured correctly, but the target ECU cannot be accessed. ► Please check if the POD is connected (via an appropriate cable) to the target microcontroller. ► Please check if the target system is powered correctly and operational. ► Please check if the target interface settings of the POD match the ECU configuration. For some targets, the target interface is disabled if the ECU parameter structure is not found. ► Please check if the ECU parameter structure (gVX1000) is present in the target RAM and that its address is correctly configured in VXconfig.
R R R R ☐☐☐☐	Inconsistent Measurement Data detected. The target microcontroller is accessible and was configured correctly to be accessed by the VX1000 system. The VX1000 detected a situation where the consistency of the measurement data cannot be guaranteed anymore. Therefore the measurement process was stopped. All data processed up to that event are valid and consistent. This problem can either be caused by: ► Overload Situations

ECU

Sequence	Description
	- Possible Solution: Increase target interface clock frequency of the POD, if possible. ► Technical issues of either ECU or VX1000 - See the VXconsole log or the messages sent via XCP to the XCP-Master (e.g. CANape) for details.
G G G G □ □ □ □	RAM synchronization in process
G G G G G G G G	Target microcontroller detected and configuration correct
G G G G G G G R	Target microcontroller detected but configuration incorrect The target microcontroller is accessible but was not configured correctly to be accessed by the VX1000 system. This means the settings made in VXconfig do not match the properties of the ECU Code Instrumentation. ► Please check if the ECU Parameter Structure address set in VXconfig matches the address in the Linker MAP file of the target ECU code. ► Please check if the byte order set in VXconfig matches the byte order of the target ECU.

ETH (Ethernet)

Sequence	Description
□ □ □ □ □ □ □ □	Ethernet link is down. The Ethernet link between the VX device and the next active network component (e.g. PC or switch) is down. ► Please verify that an appropriate Ethernet cable is properly connected to the Base Module. ► Please verify that all network components between the Base Module and the PC are operational. ► Please verify if the Ethernet cable is not damaged.
R R R R R R R R	Ethernet link up but no IP address assigned via DHCP
R □ R □ R □ R □	Ethernet link activity but no IP address assigned via DHCP . The Ethernet link is up and the IP address has not yet been assigned by a DHCP server. This might take a few seconds. As soon as a valid IP address has been assigned, the ETH LED color will change to green. The System is not accessible unless a valid IP address is configured. Depending on the network configuration there are different solutions: 1. Network with static IP address allocation ► Please set a valid IP address for the given network with VXconfig as described in section VX device IP Address Configuration on page 50. Please verify afterwards with VXconfig that the DHCP Mode is set to “Off”. 2. Network with dynamic IP address allocation (assigned via DHCP server)

ETH (Ethernet)

Sequence	Description
	► Please consult the IT systems administrator if the ETH LED color stays red.
G G G G G G G G	Ethernet link is up (network configured)
G ☐ G ☐ G ☐ G ☐	Ethernet link activity (network configured). The Base Module detected an active Ethernet connection and is configured with a valid IP address either via the DHCP server or the stored static IP address. If there is no network connection with the Base Module possible, the following items should be verified: ► Please check if the configured IP address of the System matches the subnet of the used network. ► Please check if the network traffic between the PC and the System is not filtered (blocked) by any Firewall present in the network.

XCP

Sequence	Description
☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	No XCP master connected to the device.
R R R R ☐ ☐ ☐ ☐	XCP connection to Config XCP slave (no ECU access). A XCP master is connected to the Config XCP port/slave of the Base Module. This port is reserved for the VXtools (VXconfig and VXupdate) and cannot be used by the Measurement tool as access to the target is suppressed. ► Please verify that the measurement port configuration of the Measurement tool (A2L) and the VX device are matching. The settings of the VX device can be checked and modified via VXconfig.
G G G G G G G G	XCP connection to Fully-Featured or Limited XCP slave
G G G G ☐ ☐ ☐ ☐	DAQ measurement running
G G G G R R R R	DAQ measurement running and DAQ overload detected The ECU generated too much measurement data in a short period of time which could not be processed by the System. In that case the measurement process will continue but there will be a data loss. All available data are valid and consistent. ► Please increase the target interface clock frequency of the POD, if possible. ► Please reduce the amount of generated measurement data by reducing the number of measurement signals in the measurement configuration. ► Please reduce the amount of generated measurement data by reducing the size of the data trace windows.

4.1.3 Technical Data

Dimensions (W/D/H)	106 x 115 x 32 mm
Weight	330 g
Power supply	5.5 VDC ... 50 VDC (operating voltage) 7.5 VDC ... 50 VDC (start-up voltage)
Power consumption	typ. 250 mA @ 12 V (without POD) Standby: 70 mA @ 12 V
Temperature range	-40 °C ... +70 °C
PC interface	Ethernet 100 Mbit/s
XCP data rate	up to 10 Mbyte/s DAQ data
Part number	22218

4.1.4 Accessories

- Vector Power Supply 12V/1.5A (page 645)
- Cable Banana Plug <> Binder 3-pin (page 651)
- Power Sync Cable 2x Binder <> Banana Plugs (page 652)
- Car Power Supply Cable 12V with Binder (page 654)
- Cable Binder 3polConnector with Pigtail (page 655)
- Vector SYNCcableXL (page 656)
- Fix Kit 32mm Device (page 675)

4.2 VX1134 Base Modules

4.2.1 VX1134B

4.2.1.1 Connectors

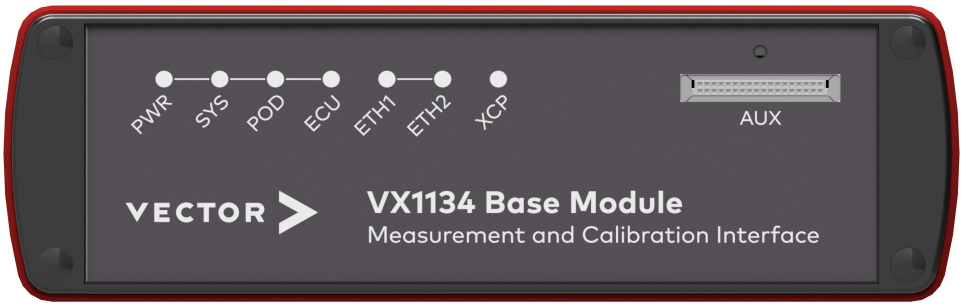


Figure 7: VX1134B Base Module front side

- **AUX**
  The AUX port is an interface to connect additional hardware to the VX1134B. For now, no auxiliary hardware is available.

Figure 8: VX1134B Base Module back side

- **POD1 (red or green)**
  This connector is used to attach a VX14xx HSSL POD with VX137x HSSL Cable.

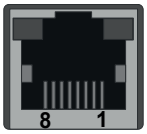
  A red connector isolation indicates that the connector housing (and therefore the shield of a connected cable) is directly connected to the internal system ground of the VX1134B Base Module.

  A green connector isolation indicates that the connector housing (and therefore the shield of a connected cable) is capacitively coupled to the internal system ground of the VX1134B Base Module. This variant has been discontinued.

- **ETH2 • TARGET**
  The ETH2 Ethernet port acts as a built-in switch port. It supports Ethernet IEEE802.3 (100 Mbit/s, 1000 Mbit/s) as well as 100BASE-T1 physical layer. The different physical layers are automatically detected. The ETH2 port is intended to connect a VX1621 XPOD or a cascaded VX1000 Base Module to the Host PC through the ETH HOST port.

Standard Ethernet connector (RJ45) for 100BASE-TX, 1000BASE-T or 100BASE-T1.

Pin	Assignment for 100BASE-T1
1	MDI+
2	MDI-
3...8	Not connected



WARNING!
Please note that VX1000 Ethernet ports supporting both 1000BASE-T and 100BASE-T1 are only intended as interface to measurement and calibration hardware, like an XPOD. They are not designed to be connected to automotive Ethernet networks.

These ports are part of an Ethernet switch connecting the VX device itself, and any network node or network segment attached to ETH ports of the VX device. Such, the ETH ports may be exposed to unexpected Ethernet traffic, like broadcasts, which might not be tolerable in an automotive Ethernet network. Also, due to supporting both, standard Ethernet, and 100BASE-T1 on the same port, the electrical circuit slightly differs from what is common in automotive Ethernet networks, which also might not be tolerable in an automotive Ethernet network.

To communicate with automotive Ethernet networks, please use dedicated Vector Network Interface Hardware.

The ETH2 Ethernet connector contains two LEDs (BR and ETH). The description of both indicators is given in the following two tables.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off Y = yellow G = green

BR/AE
LED indicating the 100BASE-T1 status.

Sequence	Description
G G G G G G G G	100BASE-T1 link is up with VX1134 as Master
G □ G □ G □ G □	100BASE-T1 activity
Y Y Y Y Y Y Y Y	100BASE-T1 link is up with VX1134 as Slave
Y □ Y □ Y □ Y □	100BASE-T1 activity

ETH
LED indicating the Ethernet status.

Sequence	Description
G G G G G G G G	Ethernet link is up with 100 Mbit/s
G □ G □ G □ G □	Ethernet Activity
Y Y Y Y Y Y Y Y	Ethernet link is up with 1000 Mbit/s
Y □ Y □ Y □ Y □	Ethernet Activity

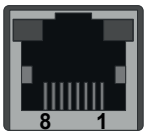
ETH1 • HOST

The communication between a Measurement and Calibration tool like CANape and the VX1000 system is implemented by XCP over Ethernet. Therefore the VX1134 must be connected to a common PC network with a network cable (Cat 5 or better). As the Ethernet interface features Auto-MDI(X) a crossover cable or patch cable can be used equivalently.

The following protocols are supported for measurement and calibration:
- XCP on TCP
- XCP on UDP

Standard Ethernet connector (RJ45) for 100BASE-TX, 1000BASE-T or 100BASE-T1.

Pin	Assignment for 100BASE-T1
1	MDI+
2	MDI-
3...8	Not connected



WARNING!
Please note that VX1000 Ethernet ports supporting both 1000BASE-T and 100BASE-T1 are only intended as interface to measurement and calibration hardware, like an XPOD. They are not designed to be connected to automotive Ethernet networks.

These ports are part of an Ethernet switch connecting the VX device itself, and any network node or network segment attached to ETH ports of the VX device. Such, the ETH ports may be exposed to unexpected Ethernet traffic, like broadcasts, which might not be tolerable in an automotive Ethernet network. Also, due to supporting both, standard Ethernet, and 100BASE-T1 on the same port, the electrical circuit slightly differs from what is common in automotive Ethernet networks, which also might not be tolerable in an automotive Ethernet network.

To communicate with automotive Ethernet networks, please use dedicated Vector Network Interface Hardware.

Ensure that no Firewall on the network or on the PC blocks the traffic of the VX1000 system. The Ethernet connector contains two LEDs (100 and ACT). The description of both indicators is given in the following two tables. The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off Y = yellow G = green

LNK
LED indicating the link speed.

Sequence	Description
□□□□□□□□	Ethernet link is down
G G G G G G G G	100 Mbit/s Ethernet link is up
Y Y Y Y Y Y Y Y	1000 Mbit/s Ethernet link is up

ACT

LED indicating the Ethernet activity status.

Sequence	Description
☐☐☐☐☐☐☐☐	No Ethernet activity
G☐G☐G☐G☐	Ethernet activity

- **PWR**

  Power is provided through a 2-pin ODU Mini-Snap male connector (labeled with PWR) on the back side of the VX1134B. The power supply is protected against reverse polarity and over-current. All power domains (input power supply, POD/ECU, FlexRay, CAN) are galvanically isolated to avoid stray electric currents. Vector offers different accessories for supplying power with a mating 2-pin ODU Mini-Snap female connector.

Pin	Assignment
2	GND
1	VBAT

- **SYNC**

  Hardware time synchronization allows the Measurement and Calibration tool to synchronize the timestamps of different devices to a common time base with a precision of 1 µs.For synchronization of VX devices a two-wire connection (signals: Sync, GND) between all devices is required. At each falling edge on the Sync line the VX device generates a timestamp that is provided to the Measurement and Calibration tool via XCP.

Reference

Power supply cables and sync cables can be found in section Accessories on page 580.

4.2.1.2 LEDs

Overview

On the front side, there are several LEDs that display the state of the Base Module, the attached POD and the target ECU.

The following tables list all available LED sequences and describe the condition at which they will be active.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off R = red G = green

PWR (Power)

Sequence	Description
R R R R R R R R	Device is powered but Firmware not running. The PWR LED will light up red at the beginning of the boot sequence of the Base Module. ► Please contact Vector VX support if the PWR LED stays red for more than 10 seconds. In this case a critical error has occurred.
G G G G G G G G	Firmware loaded successfully.
G □ G □ G □ G □	Response to ”Identify VX device command” (VXconfig)
R □ □ □ □ □ □ □	Standby Mode is active. A cyclical short red pulse on the PWR LED indicates that the Base Module is in Standby Mode. ► Please contact the VX support if the Standby Mode is not left even if the Ethernet Host link is up again for more than 10 seconds.

SYS (System)

Sequence	Description
□ □ □ □ □ □ □ □	Firmware not loaded.
R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Base Module, its system components are initialized. If a component fails to initialize properly, the SYS LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
G G G G G G G G	Device is in normal operating mode

POD

Sequence	Description
☐☐☐☐☐☐☐☐	Device is not in normal operating mode.
R R R R R R R R	POD not detected. After the Base Module boot sequence has been completed successfully, the system searches for an attached POD. As long as no POD is found, the POD LED will stay red. ► Please check if the cable between POD and Base Module is undamaged. ► Please also check if all connectors are correctly locked into their corresponding sockets.
G G G G G G G G	POD in normal operating mode
G G G G R R R R	POD detected but configuration incorrect The POD was detected by the Base Module, but the target interface configuration is incorrect. ► Please open VXconfig and adjust the POD settings according to the target ECU configuration.

ECU

Sequence	Description
☐☐☐☐☐☐☐☐	POD not in normal operating mode
R R R R R R R R	Target microcontroller not detected. The POD was detected by the VX1000 Base Module and is configured correctly, but the target ECU cannot be accessed. ► Please check if the POD is connected (via an appropriate cable) to the target microcontroller. ► Please check if the target system is powered correctly and operational. ► Please check if the target interface settings of the POD match the ECU configuration. For some targets, the target interface is disabled if the ECU parameter structure is not found. ► Please check if the ECU parameter structure (gVX1000) is present in the target RAM and that its address is correctly configured in VXconfig.
R R R R ☐☐☐☐	Inconsistent Measurement Data detected. The target microcontroller is accessible and was configured correctly to be accessed by the VX1000 system. The VX1000 detected a situation where the consistency of the measurement data cannot be guaranteed anymore. Therefore the measurement process was stopped. All data processed up to that event are valid and consistent. This problem can either be caused by: ► Overload Situations

ECU

Sequence	Description
	- Possible Solution: Increase target interface clock frequency of the POD, if possible. ► Technical issues of either ECU or VX1000 - See the VXconsole log or the messages sent via XCP to the XCP-Master (e.g. CANape) for details.
G G G G □ □ □ □	RAM synchronization in process
G G G G G G G G	Target microcontroller detected and configuration correct
G G G G G G G R	Target microcontroller detected but configuration incorrect The target microcontroller is accessible but was not configured correctly to be accessed by the VX1000 system. This means the settings made in VXconfig do not match the properties of the ECU Code Instrumentation. ► Please check if the ECU Parameter Structure address set in VXconfig matches the address in the Linker MAP file of the target ECU code. ► Please check if the byte order set in VXconfig matches the byte order of the target ECU.

ETH (Ethernet)

Sequence	Description
□ □ □ □ □ □ □ □	Ethernet link is down. The Ethernet link between the VX device and the next active network component (e.g. PC or switch) is down. ► Please verify that an appropriate Ethernet cable is properly connected to the Base Module. ► Please verify that all network components between the Base Module and the PC are operational. ► Please verify if the Ethernet cable is not damaged.
R R R R R R R R	Ethernet link up but no IP address assigned via DHCP
R □ R □ R □ R □	Ethernet link activity but no IP address assigned via DHCP . The Ethernet link is up and the IP address has not yet been assigned by a DHCP server. This might take a few seconds. As soon as a valid IP address has been assigned, the ETH LED color will change to green. The System is not accessible unless a valid IP address is configured. Depending on the network configuration there are different solutions: 1. Network with static IP address allocation ► Please set a valid IP address for the given network with VXconfig as described in section VX device IP Address Configuration on page 50. Please verify afterwards with VXconfig that the DHCP Mode is set to “Off”. 2. Network with dynamic IP address allocation (assigned via DHCP server)

ETH (Ethernet)

Sequence	Description
	► Please consult the IT systems administrator if the ETH LED color stays red.
G G G G G G G G	Ethernet link is up (network configured)
G ☐ G ☐ G ☐ G ☐	Ethernet link activity (network configured). The Base Module detected an active Ethernet connection and is configured with a valid IP address either via the DHCP server or the stored static IP address. If there is no network connection with the Base Module possible, the following items should be verified: ► Please check if the configured IP address of the System matches the subnet of the used network. ► Please check if the network traffic between the PC and the System is not filtered (blocked) by any Firewall present in the network.

XCP

Sequence	Description
☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	No XCP master connected to the device.
R R R R ☐ ☐ ☐ ☐	XCP connection to Config XCP slave (no ECU access). A XCP master is connected to the Config XCP port/slave of the Base Module. This port is reserved for the VXtools (VXconfig and VXupdate) and cannot be used by the Measurement tool as access to the target is suppressed. ► Please verify that the measurement port configuration of the Measurement tool (A2L) and the VX device are matching. The settings of the VX device can be checked and modified via VXconfig.
G G G G G G G G	XCP connection to Fully-Featured or Limited XCP slave
G G G G ☐ ☐ ☐ ☐	DAQ measurement running
G G G G R R R R	DAQ measurement running and DAQ overload detected The ECU generated too much measurement data in a short period of time which could not be processed by the System. In that case the measurement process will continue but there will be a data loss. All available data are valid and consistent. ► Please increase the target interface clock frequency of the POD, if possible. ► Please reduce the amount of generated measurement data by reducing the number of measurement signals in the measurement configuration. ► Please reduce the amount of generated measurement data by reducing the size of the data trace windows.

4.2.1.3 Technical Data

Dimensions (W/D/H)	172 x 147 x 55 mm
Weight	1050 g (VX1134 Base Module)
Power supply	5 VDC ... 34 VDC (operating voltage) 7 VDC ... 34 VDC (start-up voltage)
Power consumption	typ. 740 mA @ 12 V (without PODs) Standby: 1 mA @ 12 V
Temperature range	-40 °C ... +60 °C
PC interface	Ethernet 1 Gbit/s
Data rate	up to 100 MB/s overall for XCP DAQ data and radar raw data
Part number	► **VX1134B Base Module** Target interface: HSSL - 22388 VX1134 Base Modul - 22427 Interface Option B for VX1134

4.2.2 VX1134C

4.2.2.1 Connectors

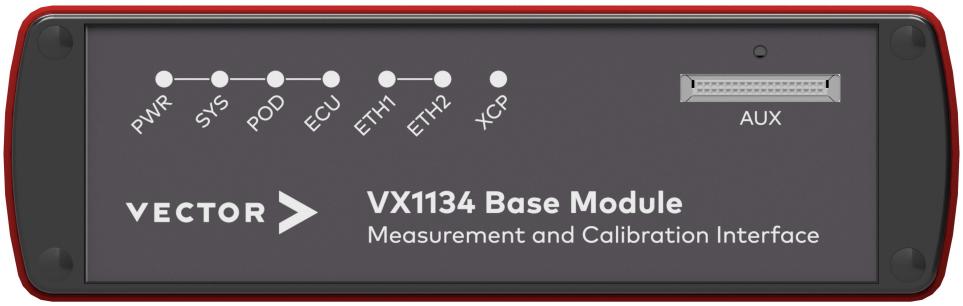


Figure 9: VX1134C Base Module front side

- **AUX**
  The AUX port is an interface to connect additional hardware to the VX1134C. For now, no auxiliary hardware is available.

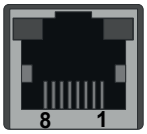
Figure 10: VX1134C Base Module back side

- **POD1 (yellow)**
  This connector is used to attach a VX14xx HSSL POD with VX1381 HSSL2 Cable. The connector housing (and therefore the shield of a connected cable) is capacitively coupled to the internal system ground of the VX1134C Base Module.

- **ETH2 • TARGET**
  The ETH2 Ethernet port acts as a built-in switch port. It supports Ethernet IEEE802.3 (100 Mbit/s, 1000 Mbit/s) as well as 100BASE-T1 physical layer. The different physical layers are automatically detected. The ETH2 port is intended to connect a VX1621 XPOD or a cascaded VX1000 Base Module to the Host PC through the ETH HOST port.

Standard Ethernet connector (RJ45) for 100BASE-TX, 1000BASE-T or 100BASE-T1.

Pin	Assignment for 100BASE-T1
1	MDI+
2	MDI-
3...8	Not connected



WARNING!
Please note that VX1000 Ethernet ports supporting both 1000BASE-T and 100BASE-T1 are only intended as interface to measurement and calibration hardware, like an XPOD. They are not designed to be connected to automotive Ethernet networks.

These ports are part of an Ethernet switch connecting the VX device itself, and any network node or network segment attached to ETH ports of the VX device. Such, the ETH ports may be exposed to unexpected Ethernet traffic, like broadcasts, which might not be tolerable in an automotive Ethernet network. Also, due to supporting both, standard Ethernet, and 100BASE-T1 on the same port, the electrical circuit slightly differs from what is common in automotive Ethernet networks, which also might not be tolerable in an automotive Ethernet network.

To communicate with automotive Ethernet networks, please use dedicated Vector Network Interface Hardware.

The ETH2 Ethernet connector contains two LEDs (BR and ETH). The description of both indicators is given in the following two tables.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off Y = yellow G = green

BR/AE
LED indicating the 100BASE-T1 status.

Sequence	Description
G G G G G G G G	100BASE-T1 link is up with VX1134 as Master
G □ G □ G □ G □	100BASE-T1 activity
Y Y Y Y Y Y Y Y	100BASE-T1 link is up with VX1134 as Slave
Y □ Y □ Y □ Y □	100BASE-T1 activity

ETH
LED indicating the Ethernet status.

Sequence	Description
G G G G G G G G	Ethernet link is up with 100 Mbit/s
G □ G □ G □ G □	Ethernet Activity
Y Y Y Y Y Y Y Y	Ethernet link is up with 1000 Mbit/s
Y □ Y □ Y □ Y □	Ethernet Activity

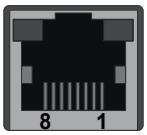
ETH1 • HOST

The communication between a Measurement and Calibration tool like CANape and the VX1000 system is implemented by XCP over Ethernet. Therefore the VX1134 must be connected to a common PC network with a network cable (Cat 5 or better). As the Ethernet interface features Auto-MDI(X) a crossover cable or patch cable can be used equivalently.

The following protocols are supported for measurement and calibration:
- XCP on TCP
- XCP on UDP

Standard Ethernet connector (RJ45) for 100BASE-TX, 1000BASE-T or 100BASE-T1.

Pin	Assignment for 100BASE-T1
1	MDI+
2	MDI-
3...8	Not connected



WARNING!
Please note that VX1000 Ethernet ports supporting both 1000BASE-T and 100BASE-T1 are only intended as interface to measurement and calibration hardware, like an XPOD. They are not designed to be connected to automotive Ethernet networks.

These ports are part of an Ethernet switch connecting the VX device itself, and any network node or network segment attached to ETH ports of the VX device. Such, the ETH ports may be exposed to unexpected Ethernet traffic, like broadcasts, which might not be tolerable in an automotive Ethernet network. Also, due to supporting both, standard Ethernet, and 100BASE-T1 on the same port, the electrical circuit slightly differs from what is common in automotive Ethernet networks, which also might not be tolerable in an automotive Ethernet network.

To communicate with automotive Ethernet networks, please use dedicated Vector Network Interface Hardware.

Ensure that no Firewall on the network or on the PC blocks the traffic of the VX1000 system. The Ethernet connector contains two LEDs (100 and ACT). The description of both indicators is given in the following two tables. The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off Y = yellow G = green

LNK
LED indicating the link speed.

Sequence	Description
□□□□□□□□	Ethernet link is down
G G G G G G G G	100 Mbit/s Ethernet link is up
Y Y Y Y Y Y Y Y	1000 Mbit/s Ethernet link is up

ACT

LED indicating the Ethernet activity status.

Sequence	Description
☐☐☐☐☐☐☐☐	No Ethernet activity
G☐G☐G☐G☐	Ethernet activity

- **PWR**

  Power is provided through a 2-pin ODU Mini-Snap male connector (labeled with PWR) on the back side of the VX1134C. The power supply is protected against reverse polarity and over-current. All power domains (input power supply, POD/ECU, FlexRay, CAN) are galvanically isolated to avoid stray electric currents. Vector offers different accessories for supplying power with a mating 2-pin ODU Mini-Snap female connector.

Pin	Assignment
2	GND
1	VBAT

- **SYNC**

  Hardware time synchronization allows the Measurement and Calibration tool to synchronize the timestamps of different devices to a common time base with a precision of 1 µs.For synchronization of VX devices a two-wire connection (signals: Sync, GND) between all devices is required. At each falling edge on the Sync line the VX device generates a timestamp that is provided to the Measurement and Calibration tool via XCP.

Reference

Power supply cables and sync cables can be found in section Accessories on page 580.

4.2.2.2 LEDs

Overview

On the front side, there are several LEDs that display the state of the Base Module, the attached POD and the target ECU.

The following tables list all available LED sequences and describe the condition at which they will be active.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off R = red G = green

PWR (Power)

Sequence	Description
R R R R R R R R	Device is powered but Firmware not running. The PWR LED will light up red at the beginning of the boot sequence of the Base Module. ► Please contact Vector VX support if the PWR LED stays red for more than 10 seconds. In this case a critical error has occurred.
G G G G G G G G	Firmware loaded successfully.
G □ G □ G □ G □	Response to ”Identify VX device command” (VXconfig)
R □ □ □ □ □ □ □	Standby Mode is active. A cyclical short red pulse on the PWR LED indicates that the Base Module is in Standby Mode. ► Please contact the VX support if the Standby Mode is not left even if the Ethernet Host link is up again for more than 10 seconds.

SYS (System)

Sequence	Description
□ □ □ □ □ □ □ □	Firmware not loaded.
R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Base Module, its system components are initialized. If a component fails to initialize properly, the SYS LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
G G G G G G G G	Device is in normal operating mode

POD

Sequence	Description
☐☐☐☐☐☐☐☐	Device is not in normal operating mode.
R R R R R R R R	POD not detected. After the Base Module boot sequence has been completed successfully, the system searches for an attached POD. As long as no POD is found, the POD LED will stay red. ► Please check if the cable between POD and Base Module is undamaged. ► Please also check if all connectors are correctly locked into their corresponding sockets.
G G G G G G G G	POD in normal operating mode
G G G G R R R R	POD detected but configuration incorrect The POD was detected by the Base Module, but the target interface configuration is incorrect. ► Please open VXconfig and adjust the POD settings according to the target ECU configuration.

ECU

Sequence	Description
☐☐☐☐☐☐☐☐	POD not in normal operating mode
R R R R R R R R	Target microcontroller not detected. The POD was detected by the VX1000 Base Module and is configured correctly, but the target ECU cannot be accessed. ► Please check if the POD is connected (via an appropriate cable) to the target microcontroller. ► Please check if the target system is powered correctly and operational. ► Please check if the target interface settings of the POD match the ECU configuration. For some targets, the target interface is disabled if the ECU parameter structure is not found. ► Please check if the ECU parameter structure (gVX1000) is present in the target RAM and that its address is correctly configured in VXconfig.
R R R R ☐☐☐☐	Inconsistent Measurement Data detected. The target microcontroller is accessible and was configured correctly to be accessed by the VX1000 system. The VX1000 detected a situation where the consistency of the measurement data cannot be guaranteed anymore. Therefore the measurement process was stopped. All data processed up to that event are valid and consistent. This problem can either be caused by: ► Overload Situations

ECU

Sequence	Description
	- Possible Solution: Increase target interface clock frequency of the POD, if possible. ► Technical issues of either ECU or VX1000 - See the VXconsole log or the messages sent via XCP to the XCP-Master (e.g. CANape) for details.
G G G G □ □ □ □	RAM synchronization in process
G G G G G G G G	Target microcontroller detected and configuration correct
G G G G G G G R	Target microcontroller detected but configuration incorrect The target microcontroller is accessible but was not configured correctly to be accessed by the VX1000 system. This means the settings made in VXconfig do not match the properties of the ECU Code Instrumentation. ► Please check if the ECU Parameter Structure address set in VXconfig matches the address in the Linker MAP file of the target ECU code. ► Please check if the byte order set in VXconfig matches the byte order of the target ECU.

ETH (Ethernet)

Sequence	Description
□ □ □ □ □ □ □ □	Ethernet link is down. The Ethernet link between the VX device and the next active network component (e.g. PC or switch) is down. ► Please verify that an appropriate Ethernet cable is properly connected to the Base Module. ► Please verify that all network components between the Base Module and the PC are operational. ► Please verify if the Ethernet cable is not damaged.
R R R R R R R R	Ethernet link up but no IP address assigned via DHCP
R □ R □ R □ R □	Ethernet link activity but no IP address assigned via DHCP . The Ethernet link is up and the IP address has not yet been assigned by a DHCP server. This might take a few seconds. As soon as a valid IP address has been assigned, the ETH LED color will change to green. The System is not accessible unless a valid IP address is configured. Depending on the network configuration there are different solutions: 1. Network with static IP address allocation ► Please set a valid IP address for the given network with VXconfig as described in section VX device IP Address Configuration on page 50. Please verify afterwards with VXconfig that the DHCP Mode is set to “Off”. 2. Network with dynamic IP address allocation (assigned via DHCP server)

ETH (Ethernet)

Sequence	Description
	► Please consult the IT systems administrator if the ETH LED color stays red.
G G G G G G G G	Ethernet link is up (network configured)
G ☐ G ☐ G ☐ G ☐	Ethernet link activity (network configured). The Base Module detected an active Ethernet connection and is configured with a valid IP address either via the DHCP server or the stored static IP address. If there is no network connection with the Base Module possible, the following items should be verified: ► Please check if the configured IP address of the System matches the subnet of the used network. ► Please check if the network traffic between the PC and the System is not filtered (blocked) by any Firewall present in the network.

XCP

Sequence	Description
☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	No XCP master connected to the device.
R R R R ☐ ☐ ☐ ☐	XCP connection to Config XCP slave (no ECU access). A XCP master is connected to the Config XCP port/slave of the Base Module. This port is reserved for the VXtools (VXconfig and VXupdate) and cannot be used by the Measurement tool as access to the target is suppressed. ► Please verify that the measurement port configuration of the Measurement tool (A2L) and the VX device are matching. The settings of the VX device can be checked and modified via VXconfig.
G G G G G G G G	XCP connection to Fully-Featured or Limited XCP slave
G G G G ☐ ☐ ☐ ☐	DAQ measurement running
G G G G R R R R	DAQ measurement running and DAQ overload detected The ECU generated too much measurement data in a short period of time which could not be processed by the System. In that case the measurement process will continue but there will be a data loss. All available data are valid and consistent. ► Please increase the target interface clock frequency of the POD, if possible. ► Please reduce the amount of generated measurement data by reducing the number of measurement signals in the measurement configuration. ► Please reduce the amount of generated measurement data by reducing the size of the data trace windows.

4.2.2.3 Technical Data

Dimensions (W/D/H)	172 x 147 x 55 mm
Weight	1050 g (VX1134 Base Module)
Power supply	5 VDC ... 34 VDC (operating voltage) 7 VDC ... 34 VDC (start-up voltage)
Power consumption	typ. 740 mA @ 12 V (without PODs) Standby: 1 mA @ 12 V
Temperature range	-40 °C ... +60 °C
PC interface	Ethernet 1 Gbit/s
Data rate	up to 100 MB/s overall for XCP DAQ data and radar raw data
Part number	► **VX1134C Base Module** Target interface: HSSL2 - 22388 VX1134 Base Modul - 22426 Interface Option C for VX1134

4.2.3 Accessories

- Vector Power Supply ODU Mini-Snap (page 646)
- Cable Vehicle Input <> ODU Mini-Snap (page 650)
- Cable Banana Plug <> ODU Mini-Snap (page 653)
- Vector SYNCcableXL (page 656)
- Fix Kit Large Device (page 676)

4.3 VX1135 Base Modules

4.3.1 VX1135A

4.3.1.1 Connectors

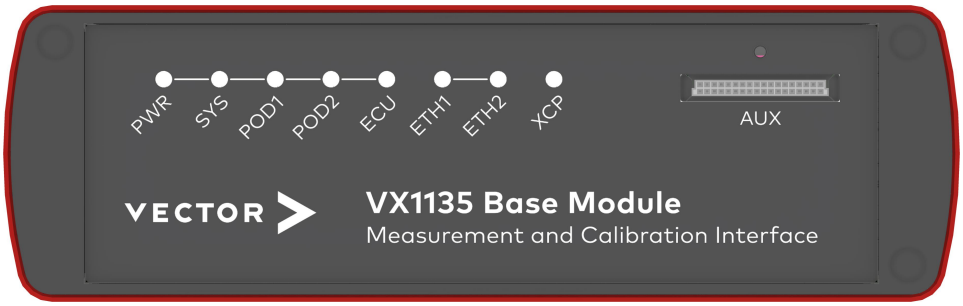


Figure 11: VX1135A Base Module front side

- **AUX**
  The AUX port is an interface to connect additional hardware to the VX1135. For now, no auxiliary hardware is available.

Figure 12: VX1135A Base Module back side

- **POD1 (blue)**
  This connector is used to attach a VX154x Serial POD. The connector housing (and therefore the shield of a connected cable) is capacitively coupled to the internal system ground of the VX1135A Base Module.
- **POD2 (red or green)**
  This connector is used to attach a VX14xx HSSL POD with VX137x HSSL Cable.

  A red connector isolation indicates that the connector housing (and therefore the shield of a connected cable) is directly connected to the internal system ground of the VX1135A Base Module.

  A green connector isolation indicates that the connector housing (and therefore the shield of a connected cable) is capacitively coupled to the internal system ground of the VX1135A Base Module. This variant has been discontinued.

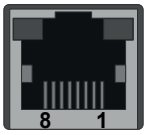
Note
The VX1135A supports only one POD at the same time. If two PODs are connected in parallel, just one will be found and used.

ETH2 • TARGET

The ETH2 Ethernet port acts as a built-in switch port. It supports Ethernet IEEE802.3 (100 Mbit/s, 1000 Mbit/s) as well as 100BASE-T1 physical layer. The different physical layers are automatically detected. The ETH2 port is intended to connect a VX1621 XPOD or a cascaded VX1000 Base Module to the Host PC through the ETH HOST port.

Standard Ethernet connector (RJ45) for 100BASE-TX, 1000BASE-T or 100BASE-T1.

Pin	Assignment for 100BASE-T1
1	MDI+
2	MDI-
3...8	Not connected



WARNING!
Please note that VX1000 Ethernet ports supporting both 1000BASE-T and 100BASE-T1 are only intended as interface to measurement and calibration hardware, like an XPOD. They are not designed to be connected to automotive Ethernet networks.

These ports are part of an Ethernet switch connecting the VX device itself, and any network node or network segment attached to ETH ports of the VX device. Such, the ETH ports may be exposed to unexpected Ethernet traffic, like broadcasts, which might not be tolerable in an automotive Ethernet network. Also, due to supporting both, standard Ethernet, and 100BASE-T1 on the same port, the electrical circuit slightly differs from what is common in automotive Ethernet networks, which also might not be tolerable in an automotive Ethernet network.

To communicate with automotive Ethernet networks, please use dedicated Vector Network Interface Hardware.

The ETH2 Ethernet connector contains two LEDs (BR and ETH). The description of both indicators is given in the following two tables.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off Y = yellow G = green

BR/AE
LED indicating the 100BASE-T1 status.

Sequence	Description
G G G G G G G G	100BASE-T1 link is up with VX1135 as Master
G □ G □ G □ G □	100BASE-T1 activity
Y Y Y Y Y Y Y Y	100BASE-T1 link is up with VX1135 as Slave
Y □ Y □ Y □ Y □	100BASE-T1 activity

ETH

LED indicating the Ethernet status.

Sequence	Description
G G G G G G G G	Ethernet link is up with 100 Mbit/s
G ☐ G ☐ G ☐ G ☐	Ethernet Activity
Y Y Y Y Y Y Y Y	Ethernet link is up with 1000 Mbit/s
Y ☐ Y ☐ Y ☐ Y ☐	Ethernet Activity

▶ **ETH1 • HOST**

The communication between a Measurement and Calibration tool like CANape and the VX1000 system is implemented by XCP over Ethernet. Therefore the VX1135 must be connected to a common PC network with a network cable (Cat 5 or better). As the Ethernet interface features Auto-MDI(X) a crossover cable or patch cable can be used equivalently.

The following protocols are supported for measurement and calibration:
- XCP on TCP
- XCP on UDP

WARNING!

Please note that VX1000 Ethernet ports supporting both 1000BASE-T and 100BASE-T1 are only intended as interface to measurement and calibration hardware, like an XPOD. They are not designed to be connected to automotive Ethernet networks.

These ports are part of an Ethernet switch connecting the VX device itself, and any network node or network segment attached to ETH ports of the VX device. Such, the ETH ports may be exposed to unexpected Ethernet traffic, like broadcasts, which might not be tolerable in an automotive Ethernet network. Also, due to supporting both, standard Ethernet, and 100BASE-T1 on the same port, the electrical circuit slightly differs from what is common in automotive Ethernet networks, which also might not be tolerable in an automotive Ethernet network.

To communicate with automotive Ethernet networks, please use dedicated Vector Network Interface Hardware.

Ensure that no Firewall on the network or on the PC blocks the traffic of the VX1000 system. The Ethernet connector contains two LEDs (100 and ACT). The description of both indicators is given in the following two tables. The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:

□ = off Y = yellow G = green

LNK

LED indicating the link speed.

Sequence	Description
□□□□□□□□	Ethernet link is down
G G G G G G G G	100 Mbit/s Ethernet link is up
Y Y Y Y Y Y Y Y	1000 Mbit/s Ethernet link is up

ACT

LED indicating the Ethernet activity status.

Sequence	Description
□□□□□□□□	No Ethernet activity
G □ G □ G □ G □	Ethernet activity

- **PWR**

Power is provided through a 2-pin ODU Mini-Snap male connector (labeled with PWR) on the back side of the VX1135A. The power supply is protected against reverse polarity and over-current. All power domains (input power supply, POD/ECU, FlexRay, CAN) are galvanically isolated to avoid stray electric currents. Vector offers different accessories for supplying power with a mating 2-pin ODU Mini-Snap female connector.

Pin	Assignment
2	GND
1	VBAT

- **SYNC**

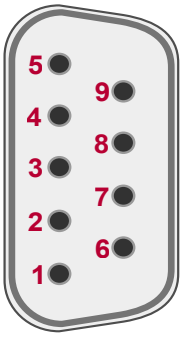
Hardware time synchronization allows the Measurement and Calibration tool to synchronize the timestamps of different devices to a common time base with a precision of 1 µs.For synchronization of VX devices a two-wire connection (signals: Sync, GND) between all devices is required. At each falling edge on the Sync line the VX device generates a timestamp that is provided to the Measurement and Calibration tool via XCP.

Reference

Power supply cables and sync cables can be found in section Accessories on page 580.

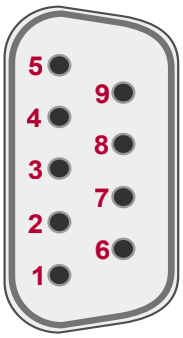
- **FLEXRAY/CAN5**
  The VX1135A offers a D-SUB9 connector on the back side of the device which combines an ISO CAN FD channel and a FlexRay channel A / B pair. The FlexRay channels use TJA1082 transceivers (capacitively decoupled). To use FlexRay, the VX1135 must be ordered with the "Full FlexRay Option". Depending on the CAN variant ordered, the CAN channels use high-speed TJA1051 transceivers (capacitively decoupled) or TJA1462BT transceivers (CAN SIC, capacitively decoupled).

Pin	Assignment	Direction
1	CAN Low	Input/output of VX1135A
2	FlexRay BM A	Input/output of VX1135A
3	FlexRay GND	-
4	FlexRay BM B	Input/output of VX1135A
5	Shield	-
6	CAN GND	-
7	FlexRay BP A	Input/output of VX1135A
8	FlexRay BP B	Input/output of VX1135A
9	CAN High	Input/output of VX1135A



- **CAN1/CAN2 and CAN3/CAN4**
  The VX1135A offers another 4 ISO CAN FD channels which are accessible through D-SUB9 connectors on the back side of the device. Depending on the CAN variant ordered, the CAN channels use high-speed TJA1051 transceivers (capacitively decoupled) or TJA1462BT transceivers (CAN SIC, capacitively decoupled).

Pin	Assignment	Direction
1	CAN Low (CH B)	Input/output of VX1135A
2	CAN Low (CH A)	Input/output of VX1135A
3	GND	-
4	reserved	-
5	Shield	-
6	GND	-
7	CAN High (CH A)	Input/output of VX1135A
8	CAN High (CH B)	Input/output of VX1135A
9	Not connected	-



Note
Please note the following hardware channel mapping:

VX1135	Vector Hardware Manager / XL API
FLEXRAY	Channel 1
CAN1	Channel 2
CAN2	Channel 3
CAN3	Channel 4
CAN4	Channel 5
CAN5	Channel 6

4.3.1.2 LEDs

Overview

On the front side, there are several LEDs that display the state of the Base Module, the attached POD and the target ECU.

The following tables list all available LED sequences and describe the condition at which they will be active.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off R = red G = green

PWR (Power)

Sequence	Description
R R R R R R R R	Device is powered but Firmware not running. The PWR LED will light up red at the beginning of the boot sequence of the Base Module. ► Please contact Vector VX support if the PWR LED stays red for more than 10 seconds. In this case a critical error has occurred.
G G G G G G G G	Firmware loaded successfully.
G □ G □ G □ G □	Response to ”Identify VX device command” (VXconfig)
R □ □ □ □ □ □ □	Standby Mode is active. A cyclical short red pulse on the PWR LED indicates that the Base Module is in Standby Mode. ► Please contact the VX support if the Standby Mode is not left even if the Ethernet Host link is up again for more than 10 seconds.

SYS (System)

Sequence	Description
□ □ □ □ □ □ □ □	Firmware not loaded.
R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Base Module, its system components are initialized. If a component fails to initialize properly, the SYS LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
G G G G G G G G	Device is in normal operating mode

POD

Sequence	Description
☐☐☐☐☐☐☐☐	Device is not in normal operating mode.
R R R R R R R R	POD not detected. After the Base Module boot sequence has been completed successfully, the system searches for an attached POD. As long as no POD is found, the POD LED will stay red. ► Please check if the cable between POD and Base Module is undamaged. ► Please also check if all connectors are correctly locked into their corresponding sockets.
G G G G G G G G	POD in normal operating mode
G G G G R R R R	POD detected but configuration incorrect The POD was detected by the Base Module, but the target interface configuration is incorrect. ► Please open VXconfig and adjust the POD settings according to the target ECU configuration.

ECU

Sequence	Description
☐☐☐☐☐☐☐☐	POD not in normal operating mode
R R R R R R R R	Target microcontroller not detected. The POD was detected by the VX1000 Base Module and is configured correctly, but the target ECU cannot be accessed. ► Please check if the POD is connected (via an appropriate cable) to the target microcontroller. ► Please check if the target system is powered correctly and operational. ► Please check if the target interface settings of the POD match the ECU configuration. For some targets, the target interface is disabled if the ECU parameter structure is not found. ► Please check if the ECU parameter structure (gVX1000) is present in the target RAM and that its address is correctly configured in VXconfig.
R R R R ☐☐☐☐	Inconsistent Measurement Data detected. The target microcontroller is accessible and was configured correctly to be accessed by the VX1000 system. The VX1000 detected a situation where the consistency of the measurement data cannot be guaranteed anymore. Therefore the measurement process was stopped. All data processed up to that event are valid and consistent. This problem can either be caused by: ► Overload Situations

ECU

Sequence	Description
	- Possible Solution: Increase target interface clock frequency of the POD, if possible. ► Technical issues of either ECU or VX1000 - See the VXconsole log or the messages sent via XCP to the XCP-Master (e.g. CANape) for details.
G G G G □ □ □ □	RAM synchronization in process
G G G G G G G G	Target microcontroller detected and configuration correct
G G G G G G G R	Target microcontroller detected but configuration incorrect The target microcontroller is accessible but was not configured correctly to be accessed by the VX1000 system. This means the settings made in VXconfig do not match the properties of the ECU Code Instrumentation. ► Please check if the ECU Parameter Structure address set in VXconfig matches the address in the Linker MAP file of the target ECU code. ► Please check if the byte order set in VXconfig matches the byte order of the target ECU.

ETH (Ethernet)

Sequence	Description
□ □ □ □ □ □ □ □	Ethernet link is down. The Ethernet link between the VX device and the next active network component (e.g. PC or switch) is down. ► Please verify that an appropriate Ethernet cable is properly connected to the Base Module. ► Please verify that all network components between the Base Module and the PC are operational. ► Please verify if the Ethernet cable is not damaged.
R R R R R R R R	Ethernet link up but no IP address assigned via DHCP
R □ R □ R □ R □	Ethernet link activity but no IP address assigned via DHCP . The Ethernet link is up and the IP address has not yet been assigned by a DHCP server. This might take a few seconds. As soon as a valid IP address has been assigned, the ETH LED color will change to green. The System is not accessible unless a valid IP address is configured. Depending on the network configuration there are different solutions: 1. Network with static IP address allocation ► Please set a valid IP address for the given network with VXconfig as described in section VX device IP Address Configuration on page 50. Please verify afterwards with VXconfig that the DHCP Mode is set to “Off”. 2. Network with dynamic IP address allocation (assigned via DHCP server)

ETH (Ethernet)

Sequence	Description
	► Please consult the IT systems administrator if the ETH LED color stays red.
G G G G G G G G	Ethernet link is up (network configured)
G ☐ G ☐ G ☐ G ☐	Ethernet link activity (network configured). The Base Module detected an active Ethernet connection and is configured with a valid IP address either via the DHCP server or the stored static IP address. If there is no network connection with the Base Module possible, the following items should be verified: ► Please check if the configured IP address of the System matches the subnet of the used network. ► Please check if the network traffic between the PC and the System is not filtered (blocked) by any Firewall present in the network.

XCP

Sequence	Description
☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	No XCP master connected to the device.
R R R R ☐ ☐ ☐ ☐	XCP connection to Config XCP slave (no ECU access). A XCP master is connected to the Config XCP port/slave of the Base Module. This port is reserved for the VXtools (VXconfig and VXupdate) and cannot be used by the Measurement tool as access to the target is suppressed. ► Please verify that the measurement port configuration of the Measurement tool (A2L) and the VX device are matching. The settings of the VX device can be checked and modified via VXconfig.
G G G G G G G G	XCP connection to Fully-Featured or Limited XCP slave
G G G G ☐ ☐ ☐ ☐	DAQ measurement running
G G G G R R R R	DAQ measurement running and DAQ overload detected The ECU generated too much measurement data in a short period of time which could not be processed by the System. In that case the measurement process will continue but there will be a data loss. All available data are valid and consistent. ► Please increase the target interface clock frequency of the POD, if possible. ► Please reduce the amount of generated measurement data by reducing the number of measurement signals in the measurement configuration. ► Please reduce the amount of generated measurement data by reducing the size of the data trace windows.

4.3.1.3 Technical Data

Dimensions (W/D/H)	172 x 147 x 55 mm
Weight	1050 g (VX1135 Base Module)
Power supply	5 VDC ... 34 VDC (operating voltage) 7 VDC ... 34 VDC (start-up voltage)
Power consumption	typ. 740 mA @ 12 V (without PODs) Standby: 1 mA @ 12 V
Temperature range	-40 °C ... +60 °C
PC interface	Ethernet 1 Gbit/s
Data rate	up to 100 MB/s overall for XCP DAQ data and radar raw data
Part number	► **VX1135A Base Module** Target interfaces: HSSL, Serial, CAN, FlexRay (optional) - 22399 VX1135 Base Module - 22420 IF Option A for VX1135 - CAN1051 - 22658 IF Option A for VX1135 - CAN1462BT

4.3.2 VX1135C

4.3.2.1 Connectors

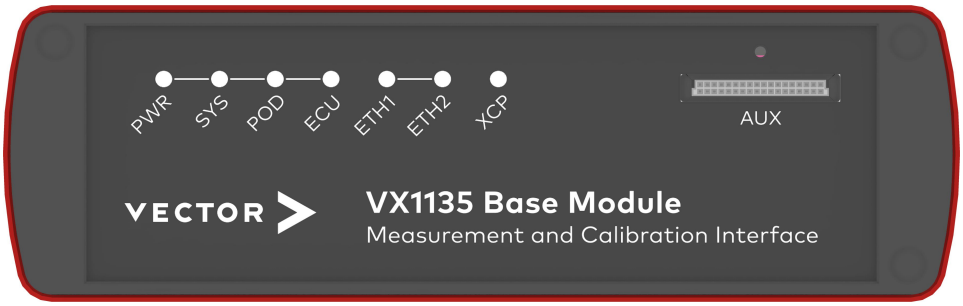


Figure 13: VX1135C Base Module front side

- **AUX**
  The AUX port is an interface to connect additional hardware to the VX1135. For now, no auxiliary hardware is available.

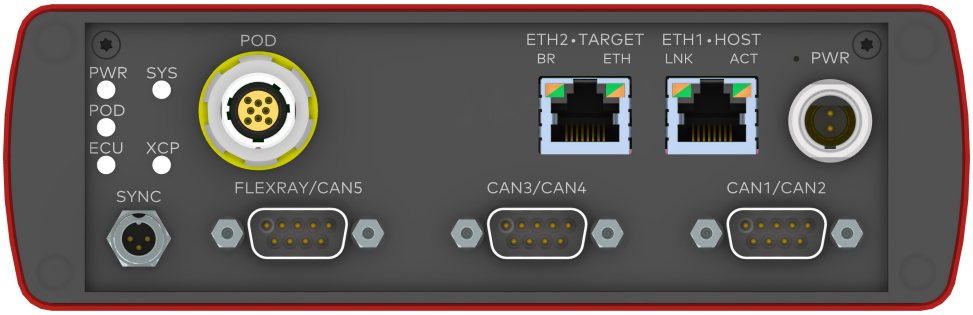


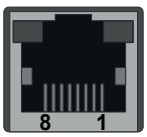
Figure 14: VX1135C Base Module back side

- **POD (yellow)**
  This connector is used to attach a VX14xx POD with With VX1381 HSSL2 Cable. The connector housing (and therefore the shield of a connected cable) is capacitively coupled to the internal system ground of the VX1135C Base Module.

- **ETH2 • TARGET**
  The ETH2 Ethernet port acts as a built-in switch port. It supports Ethernet IEEE802.3 (100 Mbit/s, 1000 Mbit/s) as well as 100BASE-T1 physical layer. The different physical layers are automatically detected. The ETH2 port is intended to connect a VX1621 XPOD or a cascaded VX1000 Base Module to the Host PC through the ETH HOST port.

  Standard Ethernet connector (RJ45) for 100BASE-TX, 1000BASE-T or 100BASE-T1.

Pin	Assignment for 100BASE-T1
1	MDI+
2	MDI-
3...8	Not connected



WARNING!
Please note that VX1000 Ethernet ports supporting both 1000BASE-T and 100BASE-T1 are only intended as interface to measurement and calibration hardware, like an XPOD. They are not designed to be connected to automotive Ethernet networks.

These ports are part of an Ethernet switch connecting the VX device itself, and any network node or network segment attached to ETH ports of the VX device. Such, the ETH ports may be exposed to unexpected Ethernet traffic, like broadcasts, which might not be tolerable in an automotive Ethernet network. Also, due to supporting both, standard Ethernet, and 100BASE-T1 on the same port, the electrical circuit slightly differs from what is common in automotive Ethernet networks, which also might not be tolerable in an automotive Ethernet network.

To communicate with automotive Ethernet networks, please use dedicated Vector Network Interface Hardware.

The ETH2 Ethernet connector contains two LEDs (BR and ETH). The description of both indicators is given in the following two tables.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off Y = yellow G = green

BR/AE
LED indicating the 100BASE-T1 status.

Sequence	Description
G G G G G G G G	100BASE-T1 link is up with VX1135 as Master
G □ G □ G □ G □	100BASE-T1 activity
Y Y Y Y Y Y Y Y	100BASE-T1 link is up with VX1135 as Slave
Y □ Y □ Y □ Y □	100BASE-T1 activity

ETH
LED indicating the Ethernet status.

Sequence	Description
G G G G G G G G	Ethernet link is up with 100 Mbit/s
G □ G □ G □ G □	Ethernet Activity
Y Y Y Y Y Y Y Y	Ethernet link is up with 1000 Mbit/s
Y □ Y □ Y □ Y □	Ethernet Activity

▶ **ETH1 • HOST**
The communication between a Measurement and Calibration tool like CANape and the VX1000 system is implemented by XCP over Ethernet. Therefore the VX1135 must be connected to a common PC network with a network cable (Cat 5 or better). As the Ethernet interface features Auto-MDI(X) a crossover cable or patch cable can be used equivalently.

The following protocols are supported for measurement and calibration:
- XCP on TCP
- XCP on UDP

WARNING!
Please note that VX1000 Ethernet ports supporting both 1000BASE-T and 100BASE-T1 are only intended as interface to measurement and calibration hardware, like an XPOD. They are not designed to be connected to automotive Ethernet networks.

These ports are part of an Ethernet switch connecting the VX device itself, and any network node or network segment attached to ETH ports of the VX device. Such, the ETH ports may be exposed to unexpected Ethernet traffic, like broadcasts, which might not be tolerable in an automotive Ethernet network. Also, due to supporting both, standard Ethernet, and 100BASE-T1 on the same port, the electrical circuit slightly differs from what is common in automotive Ethernet networks, which also might not be tolerable in an automotive Ethernet network.

To communicate with automotive Ethernet networks, please use dedicated Vector Network Interface Hardware.

Ensure that no Firewall on the network or on the PC blocks the traffic of the VX1000 system. The Ethernet connector contains two LEDs (100 and ACT). The description of both indicators is given in the following two tables. The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off Y = yellow G = green

LNK
LED indicating the link speed.

Sequence	Description
□□□□□□□□	Ethernet link is down
GGGGGGGG	100 Mbit/s Ethernet link is up
YYYYYYYY	1000 Mbit/s Ethernet link is up

ACT
LED indicating the Ethernet activity status.

Sequence	Description
□□□□□□□□	No Ethernet activity
G□G□G□G□	Ethernet activity

- **PWR**
  Power is provided through a 2-pin ODU Mini-Snap male connector (labeled with PWR) on the back side of the VX1135C. The power supply is protected against reverse polarity and over-current. All power domains (input power supply, POD/ECU, FlexRay, CAN) are galvanically isolated to avoid stray electric currents. Vector offers different accessories for supplying power with a mating 2-pin ODU Mini-Snap female connector.

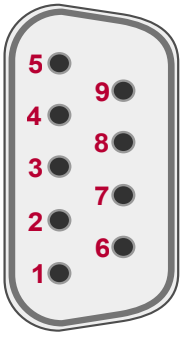
Pin	Assignment
2	GND
1	VBAT

- **SYNC**
  Hardware time synchronization allows the Measurement and Calibration tool to synchronize the timestamps of different devices to a common time base with a precision of 1 µs.For synchronization of VX devices a two-wire connection (signals: Sync, GND) between all devices is required. At each falling edge on the Sync line the VX device generates a timestamp that is provided to the Measurement and Calibration tool via XCP.

Reference
Power supply cables and sync cables can be found in section Accessories on page 580.

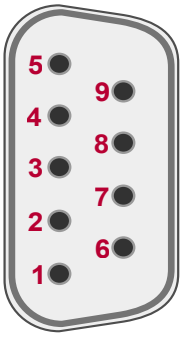
- **FLEXRAY/CAN5**
  The VX1135C offers a D-SUB9 connector on the back side of the device which combines an ISO CAN FD channel and a FlexRay channel A / B pair. The FlexRay channels use TJA1082 transceivers (capacitively decoupled). To use FlexRay, the VX1135 must be ordered with the "Full FlexRay Option". Depending on the CAN variant ordered, the CAN channels use high-speed TJA1051 transceivers (capacitively decoupled) or TJA1462BT transceivers (CAN SIC, capacitively decoupled).

Pin	Assignment	Direction
1	CAN Low	Input/output of VX1135C
2	FlexRay BM A	Input/output of VX1135C
3	FlexRay GND	-
4	FlexRay BM B	Input/output of VX1135C
5	Shield	-
6	CAN GND	-
7	FlexRay BP A	Input/output of VX1135C
8	FlexRay BP B	Input/output of VX1135C
9	CAN High	Input/output of VX1135C



- **CAN1/CAN2 and CAN3/CAN4**
  The VX1135C offers another 4 ISO CAN FD channels which are accessible through D-SUB9 connectors on the back side of the device. Depending on the CAN variant ordered, the CAN channels use high-speed TJA1051 transceivers (capacitively decoupled) or TJA1462BT transceivers (CAN SIC, capacitively decoupled).

Pin	Assignment	Direction
1	CAN Low (CH B)	Input/output of VX1135C
2	CAN Low (CH A)	Input/output of VX1135C
3	GND	-
4	reserved	-
5	Shield	-
6	GND	-
7	CAN High (CH A)	Input/output of VX1135C
8	CAN High (CH B)	Input/output of VX1135C
9	Not connected	-



Note
Please note the following hardware channel mapping:

VX1135	Vector Hardware Manager / XL API
FLEXRAY	Channel 1
CAN1	Channel 2
CAN2	Channel 3
CAN3	Channel 4
CAN4	Channel 5
CAN5	Channel 6

4.3.2.2 LEDs

Overview

On the front side, there are several LEDs that display the state of the Base Module, the attached POD and the target ECU.

The following tables list all available LED sequences and describe the condition at which they will be active.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off R = red G = green

PWR (Power)

Sequence	Description
R R R R R R R R	Device is powered but Firmware not running. The PWR LED will light up red at the beginning of the boot sequence of the Base Module. ► Please contact Vector VX support if the PWR LED stays red for more than 10 seconds. In this case a critical error has occurred.
G G G G G G G G	Firmware loaded successfully.
G □ G □ G □ G □	Response to ”Identify VX device command” (VXconfig)
R □ □ □ □ □ □ □	Standby Mode is active. A cyclical short red pulse on the PWR LED indicates that the Base Module is in Standby Mode. ► Please contact the VX support if the Standby Mode is not left even if the Ethernet Host link is up again for more than 10 seconds.

SYS (System)

Sequence	Description
□ □ □ □ □ □ □ □	Firmware not loaded.
R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Base Module, its system components are initialized. If a component fails to initialize properly, the SYS LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
G G G G G G G G	Device is in normal operating mode

POD

Sequence	Description
☐☐☐☐☐☐☐☐	Device is not in normal operating mode.
R R R R R R R R	POD not detected. After the Base Module boot sequence has been completed successfully, the system searches for an attached POD. As long as no POD is found, the POD LED will stay red. ► Please check if the cable between POD and Base Module is undamaged. ► Please also check if all connectors are correctly locked into their corresponding sockets.
G G G G G G G G	POD in normal operating mode
G G G G R R R R	POD detected but configuration incorrect The POD was detected by the Base Module, but the target interface configuration is incorrect. ► Please open VXconfig and adjust the POD settings according to the target ECU configuration.

ECU

Sequence	Description
☐☐☐☐☐☐☐☐	POD not in normal operating mode
R R R R R R R R	Target microcontroller not detected. The POD was detected by the VX1000 Base Module and is configured correctly, but the target ECU cannot be accessed. ► Please check if the POD is connected (via an appropriate cable) to the target microcontroller. ► Please check if the target system is powered correctly and operational. ► Please check if the target interface settings of the POD match the ECU configuration. For some targets, the target interface is disabled if the ECU parameter structure is not found. ► Please check if the ECU parameter structure (gVX1000) is present in the target RAM and that its address is correctly configured in VXconfig.
R R R R ☐☐☐☐	Inconsistent Measurement Data detected. The target microcontroller is accessible and was configured correctly to be accessed by the VX1000 system. The VX1000 detected a situation where the consistency of the measurement data cannot be guaranteed anymore. Therefore the measurement process was stopped. All data processed up to that event are valid and consistent. This problem can either be caused by: ► Overload Situations

ECU

Sequence	Description
	- Possible Solution: Increase target interface clock frequency of the POD, if possible. ► Technical issues of either ECU or VX1000 - See the VXconsole log or the messages sent via XCP to the XCP-Master (e.g. CANape) for details.
G G G G □ □ □ □	RAM synchronization in process
G G G G G G G G	Target microcontroller detected and configuration correct
G G G G G G G R	Target microcontroller detected but configuration incorrect The target microcontroller is accessible but was not configured correctly to be accessed by the VX1000 system. This means the settings made in VXconfig do not match the properties of the ECU Code Instrumentation. ► Please check if the ECU Parameter Structure address set in VXconfig matches the address in the Linker MAP file of the target ECU code. ► Please check if the byte order set in VXconfig matches the byte order of the target ECU.

ETH (Ethernet)

Sequence	Description
□ □ □ □ □ □ □ □	Ethernet link is down. The Ethernet link between the VX device and the next active network component (e.g. PC or switch) is down. ► Please verify that an appropriate Ethernet cable is properly connected to the Base Module. ► Please verify that all network components between the Base Module and the PC are operational. ► Please verify if the Ethernet cable is not damaged.
R R R R R R R R	Ethernet link up but no IP address assigned via DHCP
R □ R □ R □ R □	Ethernet link activity but no IP address assigned via DHCP . The Ethernet link is up and the IP address has not yet been assigned by a DHCP server. This might take a few seconds. As soon as a valid IP address has been assigned, the ETH LED color will change to green. The System is not accessible unless a valid IP address is configured. Depending on the network configuration there are different solutions: 1. Network with static IP address allocation ► Please set a valid IP address for the given network with VXconfig as described in section VX device IP Address Configuration on page 50. Please verify afterwards with VXconfig that the DHCP Mode is set to “Off”. 2. Network with dynamic IP address allocation (assigned via DHCP server)

ETH (Ethernet)

Sequence	Description
	► Please consult the IT systems administrator if the ETH LED color stays red.
G G G G G G G G	Ethernet link is up (network configured)
G ☐ G ☐ G ☐ G ☐	Ethernet link activity (network configured). The Base Module detected an active Ethernet connection and is configured with a valid IP address either via the DHCP server or the stored static IP address. If there is no network connection with the Base Module possible, the following items should be verified: ► Please check if the configured IP address of the System matches the subnet of the used network. ► Please check if the network traffic between the PC and the System is not filtered (blocked) by any Firewall present in the network.

XCP

Sequence	Description
☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	No XCP master connected to the device.
R R R R ☐ ☐ ☐ ☐	XCP connection to Config XCP slave (no ECU access). A XCP master is connected to the Config XCP port/slave of the Base Module. This port is reserved for the VXtools (VXconfig and VXupdate) and cannot be used by the Measurement tool as access to the target is suppressed. ► Please verify that the measurement port configuration of the Measurement tool (A2L) and the VX device are matching. The settings of the VX device can be checked and modified via VXconfig.
G G G G G G G G	XCP connection to Fully-Featured or Limited XCP slave
G G G G ☐ ☐ ☐ ☐	DAQ measurement running
G G G G R R R R	DAQ measurement running and DAQ overload detected The ECU generated too much measurement data in a short period of time which could not be processed by the System. In that case the measurement process will continue but there will be a data loss. All available data are valid and consistent. ► Please increase the target interface clock frequency of the POD, if possible. ► Please reduce the amount of generated measurement data by reducing the number of measurement signals in the measurement configuration. ► Please reduce the amount of generated measurement data by reducing the size of the data trace windows.

4.3.2.3 Technical Data

Dimensions (W/D/H)	172 x 147 x 55 mm
Weight	1050 g (VX1135 Base Module)
Power supply	5 VDC ... 34 VDC (operating voltage) 7 VDC ... 34 VDC (start-up voltage)
Power consumption	typ. 740 mA @ 12 V (without PODs) Standby: 1 mA @ 12 V
Temperature range	-40 °C ... +60 °C
PC interface	Ethernet 1 Gbit/s
Data rate	up to 100 MB/s overall for XCP DAQ data and radar raw data
Part number	► **VX1135C Base Module** Target interfaces: HSSL2, CAN, FlexRay (optional) - 22399 VX1135 Base Module - 22422 IF Option C for VX1135 - CAN1051

4.3.3 VX1135D

4.3.3.1 Connectors

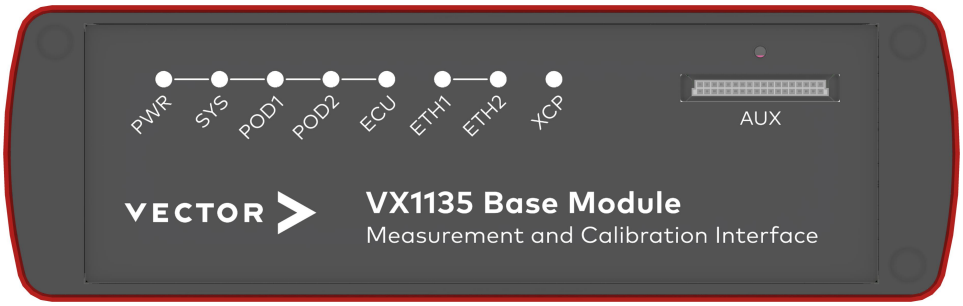


Figure 15: VX1135D Base Module front side

- **AUX**
  The AUX port is an interface to connect additional hardware to the VX1135. For now, no auxiliary hardware is available.

Figure 16: VX1135D Base Module back side

- **POD1 (red or green)**
  This connector is used to attach a VX14xx HSSL POD with VX137x HSSL Cable.

  A red connector isolation indicates that the connector housing (and therefore the shield of a connected cable) is directly connected to the internal system ground of the VX1135D Base Module.

  A green connector isolation indicates that the connector housing (and therefore the shield of a connected cable) is capacitively coupled to the internal system ground of the VX1135D Base Module. This variant has been discontinued.

- **POD2 (yellow)**
  This connector is used to attach a VX14xx HSSL POD with VX1381 HSSL2 Cable. The connector housing (and therefore the shield of a connected cable) is capacitively coupled to the internal system ground of the VX1135D Base Module.

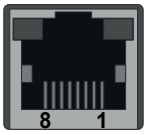
Note
The VX1135D supports only one POD at the same time. If two PODs are connected in parallel, just one will be found and used.

ETH2 • TARGET

The ETH2 Ethernet port acts as a built-in switch port. It supports Ethernet IEEE802.3 (100 Mbit/s, 1000 Mbit/s) as well as 100BASE-T1 physical layer. The different physical layers are automatically detected. The ETH2 port is intended to connect a VX1621 XPOD or a cascaded VX1000 Base Module to the Host PC through the ETH HOST port.

Standard Ethernet connector (RJ45) for 100BASE-TX, 1000BASE-T or 100BASE-T1.

Pin	Assignment for 100BASE-T1
1	MDI+
2	MDI-
3...8	Not connected



WARNING!
Please note that VX1000 Ethernet ports supporting both 1000BASE-T and 100BASE-T1 are only intended as interface to measurement and calibration hardware, like an XPOD. They are not designed to be connected to automotive Ethernet networks.

These ports are part of an Ethernet switch connecting the VX device itself, and any network node or network segment attached to ETH ports of the VX device. Such, the ETH ports may be exposed to unexpected Ethernet traffic, like broadcasts, which might not be tolerable in an automotive Ethernet network. Also, due to supporting both, standard Ethernet, and 100BASE-T1 on the same port, the electrical circuit slightly differs from what is common in automotive Ethernet networks, which also might not be tolerable in an automotive Ethernet network.

To communicate with automotive Ethernet networks, please use dedicated Vector Network Interface Hardware.

The ETH2 Ethernet connector contains two LEDs (BR and ETH). The description of both indicators is given in the following two tables.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off Y = yellow G = green

BR/AE
LED indicating the 100BASE-T1 status.

Sequence	Description
G G G G G G G G	100BASE-T1 link is up with VX1135 as Master
G □ G □ G □ G □	100BASE-T1 activity
Y Y Y Y Y Y Y Y	100BASE-T1 link is up with VX1135 as Slave
Y □ Y □ Y □ Y □	100BASE-T1 activity

ETH

LED indicating the Ethernet status.

Sequence	Description
G G G G G G G G	Ethernet link is up with 100 Mbit/s
G ☐ G ☐ G ☐ G ☐	Ethernet Activity
Y Y Y Y Y Y Y Y	Ethernet link is up with 1000 Mbit/s
Y ☐ Y ☐ Y ☐ Y ☐	Ethernet Activity

▶ **ETH1 • HOST**

The communication between a Measurement and Calibration tool like CANape and the VX1000 system is implemented by XCP over Ethernet. Therefore the VX1135 must be connected to a common PC network with a network cable (Cat 5 or better). As the Ethernet interface features Auto-MDI(X) a crossover cable or patch cable can be used equivalently.

The following protocols are supported for measurement and calibration:
- XCP on TCP
- XCP on UDP

Ensure that no Firewall on the network or on the PC blocks the traffic of the VX1000 system. The Ethernet connector contains two LEDs (100 and ACT). The description of both indicators is given in the following two tables. The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
☐ = off Y = yellow G = green

LNK

LED indicating the link speed.

Sequence	Description
☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Ethernet link is down
G G G G G G G G	100 Mbit/s Ethernet link is up
Y Y Y Y Y Y Y Y	1000 Mbit/s Ethernet link is up

ACT

LED indicating the Ethernet activity status.

Sequence	Description
☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	No Ethernet activity
G ☐ G ☐ G ☐ G ☐	Ethernet activity

▶ **PWR**

Power is provided through a 2-pin ODU Mini-Snap male connector (labeled with PWR) on the back side of the VX1135D. The power supply is protected against reverse polarity and over-current. All power domains (input power supply, POD/ECU, FlexRay, CAN) are galvanically isolated to avoid stray electric currents. Vector offers different accessories for supplying power with a mating 2-pin ODU Mini-Snap female connector.

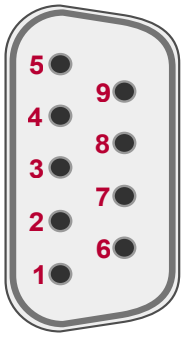
Pin	Assignment
2	GND
1	VBAT

- **SYNC**
  Hardware time synchronization allows the Measurement and Calibration tool to synchronize the timestamps of different devices to a common time base with a precision of 1 µs.For synchronization of VX devices a two-wire connection (signals: Sync, GND) between all devices is required. At each falling edge on the Sync line the VX device generates a timestamp that is provided to the Measurement and Calibration tool via XCP.

Reference
Power supply cables and sync cables can be found in section Accessories on page 580.

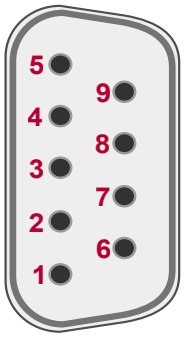
- **FLEXRAY/CAN5**
  The VX1135D offers a D-SUB9 connector on the back side of the device which combines an ISO CAN FD channel and a FlexRay channel A / B pair. The FlexRay channels use TJA1082 transceivers (capacitively decoupled). To use FlexRay, the VX1135 must be ordered with the "Full FlexRay Option". Depending on the CAN variant ordered, the CAN channels use high-speed TJA1051 transceivers (capacitively decoupled) or TJA1462BT transceivers (CAN SIC, capacitively decoupled).

Pin	Assignment	Direction
1	CAN Low	Input/output of VX1135D
2	FlexRay BM A	Input/output of VX1135D
3	FlexRay GND	-
4	FlexRay BM B	Input/output of VX1135D
5	Shield	-
6	CAN GND	-
7	FlexRay BP A	Input/output of VX1135D
8	FlexRay BP B	Input/output of VX1135D
9	CAN High	Input/output of VX1135D



- **CAN1/CAN2 and CAN3/CAN4**
  The VX1135D offers another 4 ISO CAN FD channels which are accessible through D-SUB9 connectors on the back side of the device. Depending on the CAN variant ordered, the CAN channels use high-speed TJA1051 transceivers (capacitively decoupled) or TJA1462BT transceivers (CAN SIC, capacitively decoupled).

Pin	Assignment	Direction
1	CAN Low (CH B)	Input/output of VX1135D
2	CAN Low (CH A)	Input/output of VX1135D
3	GND	-
4	reserved	-
5	Shield	-
6	GND	-
7	CAN High (CH A)	Input/output of VX1135D
8	CAN High (CH B)	Input/output of VX1135D
9	Not connected	-



Note

Please note the following hardware channel mapping:

VX1135	Vector Hardware Manager / XL API
FLEXRAY	Channel 1
CAN1	Channel 2
CAN2	Channel 3
CAN3	Channel 4
CAN4	Channel 5
CAN5	Channel 6

4.3.3.2 LEDs

Overview

On the front side, there are several LEDs that display the state of the Base Module, the attached POD and the target ECU.

The following tables list all available LED sequences and describe the condition at which they will be active.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off R = red G = green

PWR (Power)

Sequence	Description
R R R R R R R R	Device is powered but Firmware not running. The PWR LED will light up red at the beginning of the boot sequence of the Base Module. ► Please contact Vector VX support if the PWR LED stays red for more than 10 seconds. In this case a critical error has occurred.
G G G G G G G G	Firmware loaded successfully.
G □ G □ G □ G □	Response to ”Identify VX device command” (VXconfig)
R □ □ □ □ □ □ □	Standby Mode is active. A cyclical short red pulse on the PWR LED indicates that the Base Module is in Standby Mode. ► Please contact the VX support if the Standby Mode is not left even if the Ethernet Host link is up again for more than 10 seconds.

SYS (System)

Sequence	Description
□ □ □ □ □ □ □ □	Firmware not loaded.
R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Base Module, its system components are initialized. If a component fails to initialize properly, the SYS LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
G G G G G G G G	Device is in normal operating mode

POD

Sequence	Description
□□□□□□□□	Device is not in normal operating mode.
R R R R R R R R	POD not detected. After the Base Module boot sequence has been completed successfully, the system searches for an attached POD. As long as no POD is found, the POD LED will stay red. ► Please check if the cable between POD and Base Module is undamaged. ► Please also check if all connectors are correctly locked into their corresponding sockets.
G G G G G G G G	POD in normal operating mode
G G G G R R R R	POD detected but configuration incorrect The POD was detected by the Base Module, but the target interface configuration is incorrect. ► Please open VXconfig and adjust the POD settings according to the target ECU configuration.

ECU

Sequence	Description
□□□□□□□□	POD not in normal operating mode
R R R R R R R R	Target microcontroller not detected. The POD was detected by the VX1000 Base Module and is configured correctly, but the target ECU cannot be accessed. ► Please check if the POD is connected (via an appropriate cable) to the target microcontroller. ► Please check if the target system is powered correctly and operational. ► Please check if the target interface settings of the POD match the ECU configuration. For some targets, the target interface is disabled if the ECU parameter structure is not found. ► Please check if the ECU parameter structure (gVX1000) is present in the target RAM and that its address is correctly configured in VXconfig.
R R R R □ □ □ □	Inconsistent Measurement Data detected. The target microcontroller is accessible and was configured correctly to be accessed by the VX1000 system. The VX1000 detected a situation where the consistency of the measurement data cannot be guaranteed anymore. Therefore the measurement process was stopped. All data processed up to that event are valid and consistent. This problem can either be caused by: ► Overload Situations

ECU

Sequence	Description
	- Possible Solution: Increase target interface clock frequency of the POD, if possible. ► Technical issues of either ECU or VX1000 - See the VXconsole log or the messages sent via XCP to the XCP-Master (e.g. CANape) for details.
G G G G □ □ □ □	RAM synchronization in process
G G G G G G G G	Target microcontroller detected and configuration correct
G G G G G G G R	Target microcontroller detected but configuration incorrect The target microcontroller is accessible but was not configured correctly to be accessed by the VX1000 system. This means the settings made in VXconfig do not match the properties of the ECU Code Instrumentation. ► Please check if the ECU Parameter Structure address set in VXconfig matches the address in the Linker MAP file of the target ECU code. ► Please check if the byte order set in VXconfig matches the byte order of the target ECU.

ETH (Ethernet)

Sequence	Description
□ □ □ □ □ □ □ □	Ethernet link is down. The Ethernet link between the VX device and the next active network component (e.g. PC or switch) is down. ► Please verify that an appropriate Ethernet cable is properly connected to the Base Module. ► Please verify that all network components between the Base Module and the PC are operational. ► Please verify if the Ethernet cable is not damaged.
R R R R R R R R	Ethernet link up but no IP address assigned via DHCP
R □ R □ R □ R □	Ethernet link activity but no IP address assigned via DHCP . The Ethernet link is up and the IP address has not yet been assigned by a DHCP server. This might take a few seconds. As soon as a valid IP address has been assigned, the ETH LED color will change to green. The System is not accessible unless a valid IP address is configured. Depending on the network configuration there are different solutions: 1. Network with static IP address allocation ► Please set a valid IP address for the given network with VXconfig as described in section VX device IP Address Configuration on page 50. Please verify afterwards with VXconfig that the DHCP Mode is set to “Off”. 2. Network with dynamic IP address allocation (assigned via DHCP server)

ETH (Ethernet)

Sequence	Description
	► Please consult the IT systems administrator if the ETH LED color stays red.
G G G G G G G G	Ethernet link is up (network configured)
G ☐ G ☐ G ☐ G ☐	Ethernet link activity (network configured). The Base Module detected an active Ethernet connection and is configured with a valid IP address either via the DHCP server or the stored static IP address. If there is no network connection with the Base Module possible, the following items should be verified: ► Please check if the configured IP address of the System matches the subnet of the used network. ► Please check if the network traffic between the PC and the System is not filtered (blocked) by any Firewall present in the network.

XCP

Sequence	Description
☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	No XCP master connected to the device.
R R R R ☐ ☐ ☐ ☐	XCP connection to Config XCP slave (no ECU access). A XCP master is connected to the Config XCP port/slave of the Base Module. This port is reserved for the VXtools (VXconfig and VXupdate) and cannot be used by the Measurement tool as access to the target is suppressed. ► Please verify that the measurement port configuration of the Measurement tool (A2L) and the VX device are matching. The settings of the VX device can be checked and modified via VXconfig.
G G G G G G G G	XCP connection to Fully-Featured or Limited XCP slave
G G G G ☐ ☐ ☐ ☐	DAQ measurement running
G G G G R R R R	DAQ measurement running and DAQ overload detected The ECU generated too much measurement data in a short period of time which could not be processed by the System. In that case the measurement process will continue but there will be a data loss. All available data are valid and consistent. ► Please increase the target interface clock frequency of the POD, if possible. ► Please reduce the amount of generated measurement data by reducing the number of measurement signals in the measurement configuration. ► Please reduce the amount of generated measurement data by reducing the size of the data trace windows.

4.3.3.3 Technical Data

Dimensions (W/D/H)	172 x 147 x 55 mm
Weight	1050 g (VX1135 Base Module)
Power supply	5 VDC ... 34 VDC (operating voltage) 7 VDC ... 34 VDC (start-up voltage)
Power consumption	typ. 740 mA @ 12 V (without PODs) Standby: 1 mA @ 12 V
Temperature range	-40 °C ... +60 °C
PC interface	Ethernet 1 Gbit/s
Data rate	up to 100 MB/s overall for XCP DAQ data and radar raw data
Part number	► **VX1135D Base Module** Target interfaces: HSSL, HSSL2, CAN, FlexRay (optional) - 22399 VX1135 Base Module - 22423 IF Option D for VX1135 - CAN1051 - 22660 IF Option D for VX1135 - CAN1462BT

4.3.4 VX1135E

4.3.4.1 Connectors

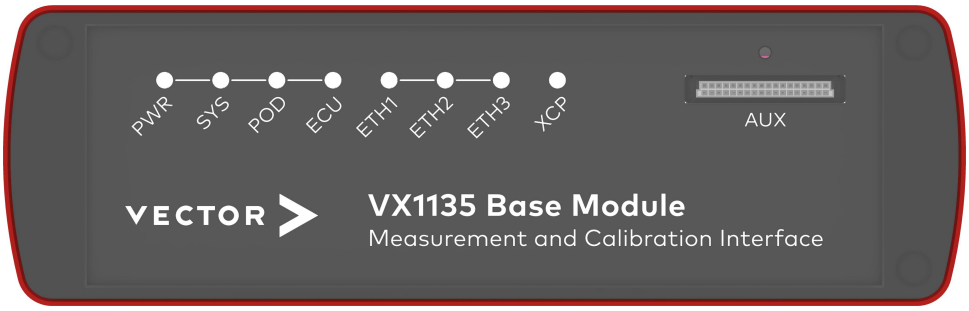


Figure 17: VX1135E Base Module front side

- **AUX**
  The AUX port is an interface to connect additional hardware to the VX1135. For now, no auxiliary hardware is available.

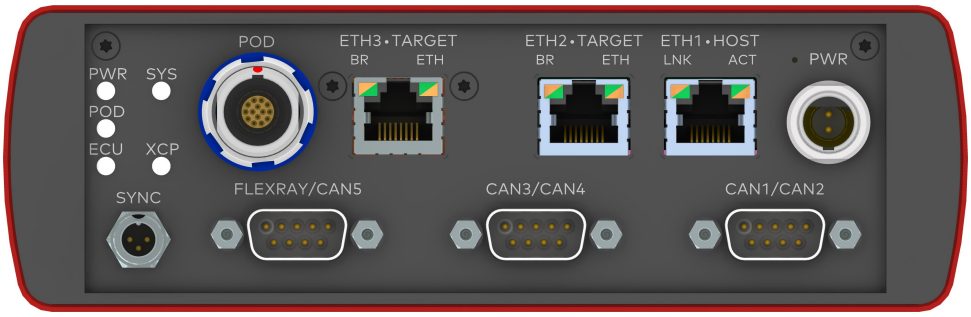


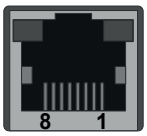
Figure 18: VX1135E Base Module back side

- **POD (blue)**
  This connector is used to attach a VX154x Serial POD. The connector housing (and therefore the shield of a connected cable) is capacitively coupled to the internal system ground of the VX1135E Base Module.

- **ETH2/ETH3 • TARGET**
  The ETH2/ETH3 Ethernet port acts as a built-in switch port. It supports Ethernet IEEE802.3 (100 Mbit/s, 1000 Mbit/s) as well as 100BASE-T1 physical layer. The different physical layers are automatically detected.

  Standard Ethernet connector (RJ45) for 100BASE-TX, 1000BASE-T or 100BASE-T1.

Pin	Assignment for 100BASE-T1
1	MDI+
2	MDI-
3...8	Not connected



WARNING!
Please note that VX1000 Ethernet ports supporting both 1000BASE-T and 100BASE-T1 are only intended as interface to measurement and calibration hardware, like an XPOD. They are not designed to be connected to automotive Ethernet networks.

These ports are part of an Ethernet switch connecting the VX device itself, and any network node or network segment attached to ETH ports of the VX device. Such, the ETH ports may be exposed to unexpected Ethernet traffic, like broadcasts, which might not be tolerable in an automotive Ethernet network. Also, due to supporting both, standard Ethernet, and 100BASE-T1 on the same port, the electrical circuit slightly differs from what is common in automotive Ethernet networks, which also might not be tolerable in an automotive Ethernet network.

To communicate with automotive Ethernet networks, please use dedicated Vector Network Interface Hardware.

The ETH2/ETH3 port is intended to connect a VX1621 XPOD or a cascaded VX1000 Base Module to the Host PC through the ETH HOST port. The ETH2/ETH3 Ethernet connector contains two LEDs (BR and ETH). The description of both indicators is given in the following two tables.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off Y = yellow G = green

BR/AE
LED indicating the 100BASE-T1 status.

Sequence	Description
G G G G G G G G	100BASE-T1 link is up with VX1135 as Master
G □ G □ G □ G □	100BASE-T1 activity
Y Y Y Y Y Y Y Y	100BASE-T1 link is up with VX1135 as Slave
Y □ Y □ Y □ Y □	100BASE-T1 activity

ETH
LED indicating the Ethernet status.

Sequence	Description
G G G G G G G G	Ethernet link is up with 100 Mbit/s
G □ G □ G □ G □	Ethernet Activity
Y Y Y Y Y Y Y Y	Ethernet link is up with 1000 Mbit/s
Y □ Y □ Y □ Y □	Ethernet Activity

Note
Communication between the Ethernet target ports ETH2 and ETH3 is not possible.

ETH1 • HOST

The communication between a Measurement and Calibration tool like CANape and the VX1000 system is implemented by XCP over Ethernet. Therefore the VX1135E must be connected to a common PC network with a network cable (Cat 5 or better). As the Ethernet interface features Auto-MDI(X) a crossover cable or patch cable can be used equivalently.

The following protocols are supported for measurement and calibration:
- XCP on TCP
- XCP on UDP

Ensure that no Firewall on the network or on the PC blocks the traffic of the VX1000 system. The Ethernet connector contains two LEDs (100 and ACT). The description of both indicators is given in the following two tables. The LED indicators show a recurring sequence of colors and blink codes.

The following block icons are used to describe this sequence:
□ = off Y = yellow G = green

LNK

LED indicating the link speed.

Sequence	Description
□□□□□□□□	Ethernet link is down
GGGGGGGG	100 Mbit/s Ethernet link is up
YYYYYYYY	1000 Mbit/s Ethernet link is up

ACT

LED indicating the Ethernet activity status.

Sequence	Description
□□□□□□□□	No Ethernet activity
G□G□G□G□	Ethernet activity

PWR

Power is provided through a 2-pin ODU Mini-Snap male connector (labeled with PWR) on the back side of the VX1135E. The power supply is protected against reverse polarity and over-current. All power domains (input power supply, POD/ECU, FlexRay, CAN) are galvanically isolated to avoid stray electric currents. Vector offers different accessories for supplying power with a mating 2-pin ODU Mini-Snap female connector.

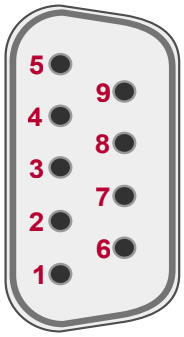
Pin	Assignment
2	GND
1	VBAT

- **SYNC**
  Hardware time synchronization allows the Measurement and Calibration tool to synchronize the timestamps of different devices to a common time base with a precision of 1 µs.For synchronization of VX devices a two-wire connection (signals: Sync, GND) between all devices is required. At each falling edge on the Sync line the VX device generates a timestamp that is provided to the Measurement and Calibration tool via XCP.

Reference
Power supply cables and sync cables can be found in section Accessories on page 580.

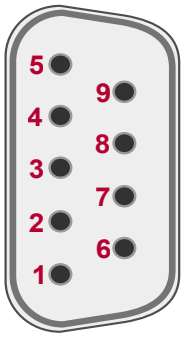
- **FLEXRAY/CAN5**
  The VX1135E offers a D-SUB9 connector on the back side of the device which combines an ISO CAN FD channel and a FlexRay channel A / B pair. The FlexRay channels use TJA1082 transceivers (capacitively decoupled). To use FlexRay, the VX1135 must be ordered with the "Full FlexRay Option". Depending on the CAN variant ordered, the CAN channels use high-speed TJA1051 transceivers (capacitively decoupled) or TJA1462BT transceivers (CAN SIC, capacitively decoupled).

Pin	Assignment	Direction
1	CAN Low	Input/output of VX1135E
2	FlexRay BM A	Input/output of VX1135E
3	FlexRay GND	-
4	FlexRay BM B	Input/output of VX1135E
5	Shield	-
6	CAN GND	-
7	FlexRay BP A	Input/output of VX1135E
8	FlexRay BP B	Input/output of VX1135E
9	CAN High	Input/output of VX1135E



- **CAN1/CAN2 and CAN3/CAN4**
  The VX1135E offers another 4 ISO CAN FD channels which are accessible through D-SUB9 connectors on the back side of the device. Depending on the CAN variant ordered, the CAN channels use high-speed TJA1051 transceivers (capacitively decoupled) or TJA1462BT transceivers (CAN SIC, capacitively decoupled).

Pin	Assignment	Direction
1	CAN Low (CH B)	Input/output of VX1135E
2	CAN Low (CH A)	Input/output of VX1135E
3	GND	-
4	reserved	-
5	Shield	-
6	GND	-
7	CAN High (CH A)	Input/output of VX1135E
8	CAN High (CH B)	Input/output of VX1135E
9	Not connected	-



Note

Please note the following hardware channel mapping:

VX1135	Vector Hardware Manager / XL API
FLEXRAY	Channel 1
CAN1	Channel 2
CAN2	Channel 3
CAN3	Channel 4
CAN4	Channel 5
CAN5	Channel 6

4.3.4.2 LEDs

Overview

On the front side, there are several LEDs that display the state of the Base Module, the attached POD and the target ECU.

The following tables list all available LED sequences and describe the condition at which they will be active.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off R = red G = green

PWR (Power)

Sequence	Description
R R R R R R R R	Device is powered but Firmware not running. The PWR LED will light up red at the beginning of the boot sequence of the Base Module. ► Please contact Vector VX support if the PWR LED stays red for more than 10 seconds. In this case a critical error has occurred.
G G G G G G G G	Firmware loaded successfully.
G □ G □ G □ G □	Response to ”Identify VX device command” (VXconfig)
R □ □ □ □ □ □ □	Standby Mode is active. A cyclical short red pulse on the PWR LED indicates that the Base Module is in Standby Mode. ► Please contact the VX support if the Standby Mode is not left even if the Ethernet Host link is up again for more than 10 seconds.

SYS (System)

Sequence	Description
□ □ □ □ □ □ □ □	Firmware not loaded.
R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Base Module, its system components are initialized. If a component fails to initialize properly, the SYS LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
G G G G G G G G	Device is in normal operating mode

POD

Sequence	Description
☐☐☐☐☐☐☐☐	Device is not in normal operating mode.
R R R R R R R R	POD not detected. After the Base Module boot sequence has been completed successfully, the system searches for an attached POD. As long as no POD is found, the POD LED will stay red. ► Please check if the cable between POD and Base Module is undamaged. ► Please also check if all connectors are correctly locked into their corresponding sockets.
G G G G G G G G	POD in normal operating mode
G G G G R R R R	POD detected but configuration incorrect The POD was detected by the Base Module, but the target interface configuration is incorrect. ► Please open VXconfig and adjust the POD settings according to the target ECU configuration.

ECU

Sequence	Description
☐☐☐☐☐☐☐☐	POD not in normal operating mode
R R R R R R R R	Target microcontroller not detected. The POD was detected by the VX1000 Base Module and is configured correctly, but the target ECU cannot be accessed. ► Please check if the POD is connected (via an appropriate cable) to the target microcontroller. ► Please check if the target system is powered correctly and operational. ► Please check if the target interface settings of the POD match the ECU configuration. For some targets, the target interface is disabled if the ECU parameter structure is not found. ► Please check if the ECU parameter structure (gVX1000) is present in the target RAM and that its address is correctly configured in VXconfig.
R R R R ☐☐☐☐	Inconsistent Measurement Data detected. The target microcontroller is accessible and was configured correctly to be accessed by the VX1000 system. The VX1000 detected a situation where the consistency of the measurement data cannot be guaranteed anymore. Therefore the measurement process was stopped. All data processed up to that event are valid and consistent. This problem can either be caused by: ► Overload Situations

ECU

Sequence	Description
	- Possible Solution: Increase target interface clock frequency of the POD, if possible. ► Technical issues of either ECU or VX1000 - See the VXconsole log or the messages sent via XCP to the XCP-Master (e.g. CANape) for details.
G G G G □ □ □ □	RAM synchronization in process
G G G G G G G G	Target microcontroller detected and configuration correct
G G G G G G G R	Target microcontroller detected but configuration incorrect The target microcontroller is accessible but was not configured correctly to be accessed by the VX1000 system. This means the settings made in VXconfig do not match the properties of the ECU Code Instrumentation. ► Please check if the ECU Parameter Structure address set in VXconfig matches the address in the Linker MAP file of the target ECU code. ► Please check if the byte order set in VXconfig matches the byte order of the target ECU.

ETH (Ethernet)

Sequence	Description
□ □ □ □ □ □ □ □	Ethernet link is down. The Ethernet link between the VX device and the next active network component (e.g. PC or switch) is down. ► Please verify that an appropriate Ethernet cable is properly connected to the Base Module. ► Please verify that all network components between the Base Module and the PC are operational. ► Please verify if the Ethernet cable is not damaged.
R R R R R R R R	Ethernet link up but no IP address assigned via DHCP
R □ R □ R □ R □	Ethernet link activity but no IP address assigned via DHCP . The Ethernet link is up and the IP address has not yet been assigned by a DHCP server. This might take a few seconds. As soon as a valid IP address has been assigned, the ETH LED color will change to green. The System is not accessible unless a valid IP address is configured. Depending on the network configuration there are different solutions: 1. Network with static IP address allocation ► Please set a valid IP address for the given network with VXconfig as described in section VX device IP Address Configuration on page 50. Please verify afterwards with VXconfig that the DHCP Mode is set to “Off”. 2. Network with dynamic IP address allocation (assigned via DHCP server)

ETH (Ethernet)

Sequence	Description
	► Please consult the IT systems administrator if the ETH LED color stays red.
G G G G G G G G	Ethernet link is up (network configured)
G ☐ G ☐ G ☐ G ☐	Ethernet link activity (network configured). The Base Module detected an active Ethernet connection and is configured with a valid IP address either via the DHCP server or the stored static IP address. If there is no network connection with the Base Module possible, the following items should be verified: ► Please check if the configured IP address of the System matches the subnet of the used network. ► Please check if the network traffic between the PC and the System is not filtered (blocked) by any Firewall present in the network.

XCP

Sequence	Description
☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	No XCP master connected to the device.
R R R R ☐ ☐ ☐ ☐	XCP connection to Config XCP slave (no ECU access). A XCP master is connected to the Config XCP port/slave of the Base Module. This port is reserved for the VXtools (VXconfig and VXupdate) and cannot be used by the Measurement tool as access to the target is suppressed. ► Please verify that the measurement port configuration of the Measurement tool (A2L) and the VX device are matching. The settings of the VX device can be checked and modified via VXconfig.
G G G G G G G G	XCP connection to Fully-Featured or Limited XCP slave
G G G G ☐ ☐ ☐ ☐	DAQ measurement running
G G G G R R R R	DAQ measurement running and DAQ overload detected The ECU generated too much measurement data in a short period of time which could not be processed by the System. In that case the measurement process will continue but there will be a data loss. All available data are valid and consistent. ► Please increase the target interface clock frequency of the POD, if possible. ► Please reduce the amount of generated measurement data by reducing the number of measurement signals in the measurement configuration. ► Please reduce the amount of generated measurement data by reducing the size of the data trace windows.

4.3.4.3 Technical Data

Dimensions (W/D/H)	172 x 147 x 55 mm
Weight	1050 g (VX1135A Base Module)
Power supply	5 VDC ... 34 VDC (operating voltage) 7 VDC ... 34 VDC (start-up voltage)
Power consumption	typ. 740 mA @ 12 V (without PODs) Standby: 1 mA @ 12 V
Temperature range	-40 °C ... +60 °C
PC interface	Ethernet 1 Gbit/s
Data rate	up to 20 MB/s XCP DAQ data
Part number	► **VX1135E Base Module** Target interfaces: Serial, 2x 100BASE-T1/ETH, CAN, FlexRay (optional) - 22399 VX1135 Base Module - 22424 IF Option E for VX1135 - CAN1051 - 22661 IF Option E for VX1135 - CAN1462BT

4.3.5 VX1135F

4.3.5.1 Connectors

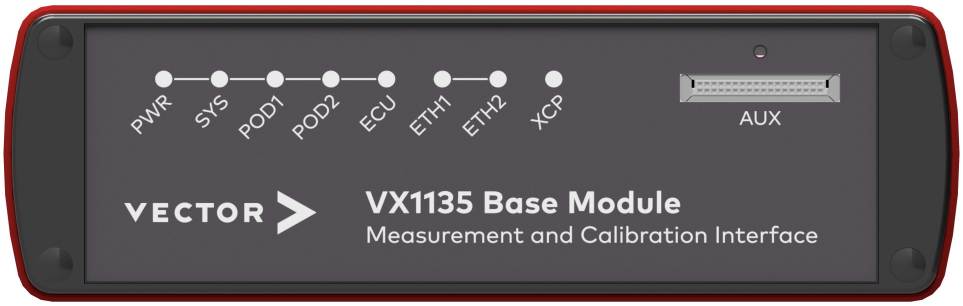


Figure 19: VX1135F Base Module front side

- **AUX**
  The AUX port is an interface to connect additional hardware to the VX1135. For now, no auxiliary hardware is available.

Figure 20: VX1135F Base Module back side

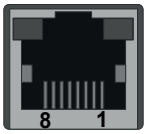
- **POD1 (blue)**
  This connector is used to attach a VX154x Serial POD. The connector housing (and therefore the shield of a connected cable) is capacitively coupled to the internal system ground of the VX1135F Base Module.
- **POD2 (yellow)**
  This connector is used to attach a VX14xx HSSL POD with VX1381 HSSL2 Cable. The connector housing (and therefore the shield of a connected cable) is capacitively coupled to the internal system ground of the VX1135F Base Module.

> **Note**
> The VX1135F supports only one POD at the same time. If two PODs are connected in parallel, just one will be found and used.

- **ETH2 • TARGET**
  The ETH2 Ethernet port acts as a built-in switch port. It supports Ethernet IEEE802.3 (100 Mbit/s, 1000 Mbit/s) as well as 100BASE-T1 physical layer. The different physical layers are automatically detected. The ETH2 port is intended to connect a VX1621 XPOD or a cascaded VX1000 Base Module to the Host PC through the ETH HOST port.

Standard Ethernet connector (RJ45) for 100BASE-TX, 1000BASE-T or 100BASE-T1.

Pin	Assignment for 100BASE-T1
1	MDI+
2	MDI-
3...8	Not connected



WARNING!
Please note that VX1000 Ethernet ports supporting both 1000BASE-T and 100BASE-T1 are only intended as interface to measurement and calibration hardware, like an XPOD. They are not designed to be connected to automotive Ethernet networks.

These ports are part of an Ethernet switch connecting the VX device itself, and any network node or network segment attached to ETH ports of the VX device. Such, the ETH ports may be exposed to unexpected Ethernet traffic, like broadcasts, which might not be tolerable in an automotive Ethernet network. Also, due to supporting both, standard Ethernet, and 100BASE-T1 on the same port, the electrical circuit slightly differs from what is common in automotive Ethernet networks, which also might not be tolerable in an automotive Ethernet network.

To communicate with automotive Ethernet networks, please use dedicated Vector Network Interface Hardware.

The ETH2 Ethernet connector contains two LEDs (BR and ETH). The description of both indicators is given in the following two tables.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off Y = yellow G = green

BR/AE
LED indicating the 100BASE-T1 status.

Sequence	Description
G G G G G G G G	100BASE-T1 link is up with VX1135 as Master
G □ G □ G □ G □	100BASE-T1 activity
Y Y Y Y Y Y Y Y	100BASE-T1 link is up with VX1135 as Slave
Y □ Y □ Y □ Y □	100BASE-T1 activity

ETH
LED indicating the Ethernet status.

Sequence	Description
G G G G G G G G	Ethernet link is up with 100 Mbit/s
G □ G □ G □ G □	Ethernet Activity
Y Y Y Y Y Y Y Y	Ethernet link is up with 1000 Mbit/s
Y □ Y □ Y □ Y □	Ethernet Activity

ETH1 • HOST

The communication between a Measurement and Calibration tool like CANape and the VX1000 system is implemented by XCP over Ethernet. Therefore the VX1135 must be connected to a common PC network with a network cable (Cat 5 or better). As the Ethernet interface features Auto-MDI(X) a crossover cable or patch cable can be used equivalently.

The following protocols are supported for measurement and calibration:
- XCP on TCP
- XCP on UDP

Ensure that no Firewall on the network or on the PC blocks the traffic of the VX1000 system. The Ethernet connector contains two LEDs (100 and ACT). The description of both indicators is given in the following two tables. The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off Y = yellow G = green

LNK

LED indicating the link speed.

Sequence	Description
□□□□□□□□	Ethernet link is down
G G G G G G G G	100 Mbit/s Ethernet link is up
Y Y Y Y Y Y Y Y	1000 Mbit/s Ethernet link is up

ACT

LED indicating the Ethernet activity status.

Sequence	Description
□□□□□□□□	No Ethernet activity
G □ G □ G □ G □	Ethernet activity

PWR

Power is provided through a 2-pin ODU Mini-Snap male connector (labeled with PWR) on the back side of the VX1135F. The power supply is protected against reverse polarity and over-current. All power domains (input power supply, POD/ECU, FlexRay, CAN) are galvanically isolated to avoid stray electric currents. Vector offers different accessories for supplying power with a mating 2-pin ODU Mini-Snap female connector.

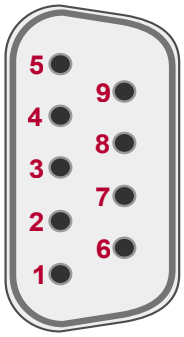
Pin	Assignment
2	GND
1	VBAT

- **SYNC**
  Hardware time synchronization allows the Measurement and Calibration tool to synchronize the timestamps of different devices to a common time base with a precision of 1 µs.For synchronization of VX devices a two-wire connection (signals: Sync, GND) between all devices is required. At each falling edge on the Sync line the VX device generates a timestamp that is provided to the Measurement and Calibration tool via XCP.

Reference
Power supply cables and sync cables can be found in section Accessories on page 580.

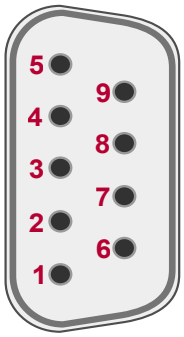
- **FLEXRAY/CAN5**
  The VX1135F offers a D-SUB9 connector on the back side of the device which combines an ISO CAN FD channel and a FlexRay channel A / B pair. The FlexRay channels use TJA1082 transceivers (capacitively decoupled). To use FlexRay, the VX1135 must be ordered with the "Full FlexRay Option". Depending on the CAN variant ordered, the CAN channels use high-speed TJA1051 transceivers (capacitively decoupled) or TJA1462BT transceivers (CAN SIC, capacitively decoupled).

Pin	Assignment	Direction
1	CAN Low	Input/output of VX1135F
2	FlexRay BM A	Input/output of VX1135F
3	FlexRay GND	-
4	FlexRay BM B	Input/output of VX1135F
5	Shield	-
6	CAN GND	-
7	FlexRay BP A	Input/output of VX1135F
8	FlexRay BP B	Input/output of VX1135F
9	CAN High	Input/output of VX1135F



- **CAN1/CAN2 and CAN3/CAN4**
  The VX1135F offers another 4 ISO CAN FD channels which are accessible through D-SUB9 connectors on the back side of the device. Depending on the CAN variant ordered, the CAN channels use high-speed TJA1051 transceivers (capacitively decoupled) or TJA1462BT transceivers (CAN SIC, capacitively decoupled).

Pin	Assignment	Direction
1	CAN Low (CH B)	Input/output of VX1135F
2	CAN Low (CH A)	Input/output of VX1135F
3	GND	-
4	reserved	-
5	Shield	-
6	GND	-
7	CAN High (CH A)	Input/output of VX1135F
8	CAN High (CH B)	Input/output of VX1135F
9	Not connected	-



Note

Please note the following hardware channel mapping:

VX1135	Vector Hardware Manager / XL API
FLEXRAY	Channel 1
CAN1	Channel 2
CAN2	Channel 3
CAN3	Channel 4
CAN4	Channel 5
CAN5	Channel 6

4.3.5.2 LEDs

Overview

On the front side, there are several LEDs that display the state of the Base Module, the attached POD and the target ECU.

The following tables list all available LED sequences and describe the condition at which they will be active.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off R = red G = green

PWR (Power)

Sequence	Description
R R R R R R R R	Device is powered but Firmware not running. The PWR LED will light up red at the beginning of the boot sequence of the Base Module. ► Please contact Vector VX support if the PWR LED stays red for more than 10 seconds. In this case a critical error has occurred.
G G G G G G G G	Firmware loaded successfully.
G □ G □ G □ G □	Response to ”Identify VX device command” (VXconfig)
R □ □ □ □ □ □ □	Standby Mode is active. A cyclical short red pulse on the PWR LED indicates that the Base Module is in Standby Mode. ► Please contact the VX support if the Standby Mode is not left even if the Ethernet Host link is up again for more than 10 seconds.

SYS (System)

Sequence	Description
□ □ □ □ □ □ □ □	Firmware not loaded.
R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Base Module, its system components are initialized. If a component fails to initialize properly, the SYS LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
G G G G G G G G	Device is in normal operating mode

POD

Sequence	Description
☐☐☐☐☐☐☐☐	Device is not in normal operating mode.
R R R R R R R R	POD not detected. After the Base Module boot sequence has been completed successfully, the system searches for an attached POD. As long as no POD is found, the POD LED will stay red. ► Please check if the cable between POD and Base Module is undamaged. ► Please also check if all connectors are correctly locked into their corresponding sockets.
G G G G G G G G	POD in normal operating mode
G G G G R R R R	POD detected but configuration incorrect The POD was detected by the Base Module, but the target interface configuration is incorrect. ► Please open VXconfig and adjust the POD settings according to the target ECU configuration.

ECU

Sequence	Description
☐☐☐☐☐☐☐☐	POD not in normal operating mode
R R R R R R R R	Target microcontroller not detected. The POD was detected by the VX1000 Base Module and is configured correctly, but the target ECU cannot be accessed. ► Please check if the POD is connected (via an appropriate cable) to the target microcontroller. ► Please check if the target system is powered correctly and operational. ► Please check if the target interface settings of the POD match the ECU configuration. For some targets, the target interface is disabled if the ECU parameter structure is not found. ► Please check if the ECU parameter structure (gVX1000) is present in the target RAM and that its address is correctly configured in VXconfig.
R R R R ☐☐☐☐	Inconsistent Measurement Data detected. The target microcontroller is accessible and was configured correctly to be accessed by the VX1000 system. The VX1000 detected a situation where the consistency of the measurement data cannot be guaranteed anymore. Therefore the measurement process was stopped. All data processed up to that event are valid and consistent. This problem can either be caused by: ► Overload Situations

ECU

Sequence	Description
	- Possible Solution: Increase target interface clock frequency of the POD, if possible. ► Technical issues of either ECU or VX1000 - See the VXconsole log or the messages sent via XCP to the XCP-Master (e.g. CANape) for details.
G G G G □ □ □ □	RAM synchronization in process
G G G G G G G G	Target microcontroller detected and configuration correct
G G G G G G G R	Target microcontroller detected but configuration incorrect The target microcontroller is accessible but was not configured correctly to be accessed by the VX1000 system. This means the settings made in VXconfig do not match the properties of the ECU Code Instrumentation. ► Please check if the ECU Parameter Structure address set in VXconfig matches the address in the Linker MAP file of the target ECU code. ► Please check if the byte order set in VXconfig matches the byte order of the target ECU.

ETH (Ethernet)

Sequence	Description
□ □ □ □ □ □ □ □	Ethernet link is down. The Ethernet link between the VX device and the next active network component (e.g. PC or switch) is down. ► Please verify that an appropriate Ethernet cable is properly connected to the Base Module. ► Please verify that all network components between the Base Module and the PC are operational. ► Please verify if the Ethernet cable is not damaged.
R R R R R R R R	Ethernet link up but no IP address assigned via DHCP
R □ R □ R □ R □	Ethernet link activity but no IP address assigned via DHCP . The Ethernet link is up and the IP address has not yet been assigned by a DHCP server. This might take a few seconds. As soon as a valid IP address has been assigned, the ETH LED color will change to green. The System is not accessible unless a valid IP address is configured. Depending on the network configuration there are different solutions: 1. Network with static IP address allocation ► Please set a valid IP address for the given network with VXconfig as described in section VX device IP Address Configuration on page 50. Please verify afterwards with VXconfig that the DHCP Mode is set to “Off”. 2. Network with dynamic IP address allocation (assigned via DHCP server)

ETH (Ethernet)

Sequence	Description
	► Please consult the IT systems administrator if the ETH LED color stays red.
G G G G G G G G	Ethernet link is up (network configured)
G ☐ G ☐ G ☐ G ☐	Ethernet link activity (network configured). The Base Module detected an active Ethernet connection and is configured with a valid IP address either via the DHCP server or the stored static IP address. If there is no network connection with the Base Module possible, the following items should be verified: ► Please check if the configured IP address of the System matches the subnet of the used network. ► Please check if the network traffic between the PC and the System is not filtered (blocked) by any Firewall present in the network.

XCP

Sequence	Description
☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	No XCP master connected to the device.
R R R R ☐ ☐ ☐ ☐	XCP connection to Config XCP slave (no ECU access). A XCP master is connected to the Config XCP port/slave of the Base Module. This port is reserved for the VXtools (VXconfig and VXupdate) and cannot be used by the Measurement tool as access to the target is suppressed. ► Please verify that the measurement port configuration of the Measurement tool (A2L) and the VX device are matching. The settings of the VX device can be checked and modified via VXconfig.
G G G G G G G G	XCP connection to Fully-Featured or Limited XCP slave
G G G G ☐ ☐ ☐ ☐	DAQ measurement running
G G G G R R R R	DAQ measurement running and DAQ overload detected The ECU generated too much measurement data in a short period of time which could not be processed by the System. In that case the measurement process will continue but there will be a data loss. All available data are valid and consistent. ► Please increase the target interface clock frequency of the POD, if possible. ► Please reduce the amount of generated measurement data by reducing the number of measurement signals in the measurement configuration. ► Please reduce the amount of generated measurement data by reducing the size of the data trace windows.

4.3.5.3 Technical Data

Dimensions (W/D/H)	172 x 147 x 55 mm
Weight	1050 g (VX1135 Base Module)
Power supply	5 VDC ... 34 VDC (operating voltage) 7 VDC ... 34 VDC (start-up voltage)
Power consumption	typ. 740 mA @ 12 V (without PODs) Standby: 1 mA @ 12 V
Temperature range	-40 °C ... +60 °C
PC interface	Ethernet 1 Gbit/s
Data rate	up to 100 MB/s overall for XCP DAQ data and radar raw data
Part number	► **VX1135F Base Module** Target interfaces: HSSL2, Serial, CAN, FlexRay (optional) - 22399 VX1135 Base Module - 22425 IF Option F for VX1135 - CAN1051 - 22662 IF Option F for VX1135 - CAN1462BT

4.3.6 Accessories

- Vector Power Supply ODU Mini-Snap (page 646)
- Cable Vehicle Input <> ODU Mini-Snap (page 650)
- Cable Banana Plug <> ODU Mini-Snap (page 653)
- Vector SYNCcableXL (page 656)
- CANcable 2Y (page 635)
- FRcable AB (page 673)
- FR/CANcable 2Y (page 674)
- Fix Kit Large Device (page 676)

4.4 VX1152 Video Interface

4.4.1 General Information

The VX1152 is a video interface designed for measuring Single-Cam ADAS ECUs during test drives, in the lab, or in the office. In contrast, the VX1161 rack system, its big brother, has its advantages in multi-cam projects and when measuring many different sensors.

Figure 21: VX1152

The ECU's video channel is measured via the Video TAP (Test Access Port). Its receiver connects to the sensor head, and the transmitter connects to the ECU. The Video TAP receives the video data, duplicates it, and forwards one copy to the ECU. The second copy is timestamped and forwarded to a recording device. The integrated video binary lossless compressor reduces the required uplink bandwidth and storage space by 40 % ... 60 % (see section Lossless Compression on page 139). Depending on the transceivers used, up to 10 Gbps of video can be processed.

Additionally, the VX1152 can carry an optional VNmodule60 4AE1G 88Q2221M (see section Automotive Ethernet on VNmodule60 (optional) on page 165).

The VX1152 provides two 10GBASE-T ports and two 100/1000BASE-T ports for host configuration, recorder connection, cascading, and PTP time synchronization. The integrated 45 Gbps Ethernet switch routes all traffic in a PTP-transparent manner.

Having the video and AEth transceivers on replaceable modules makes upgrading to future cable technologies simple and cost-efficient.

The following modules are currently supported by the VX1152:

- Video
  - VXmodule GMSL3 TAP MX96792A/MX96793
- Automotive Ethernet
  - VNmodule60 4AE1G 88Q2221M

For more modules and protocols, please contact Vector.

Note
The VXmodule60 and its video features are only supported by the VX1152, not by Vector's VN devices.

Video transceivers can operate in either TAP or RX mode. In single-camera projects, TAP mode is the standard operating mode. RX mode is useful for sensor evaluation and setting up the sensor-side transceivers before adding the ECU-side transceivers in TAP mode.

The VXconfig tool is used for configuration and the VXupdate tool is used for updates. The video transceivers and remote sensor can be initialized via CASL language scripts within VXconfig (see section Scripting Language (CASL) on page 564). The VXconsole tool can be used to read important interface information. Printouts of CASL scripts are also shown, which can be useful for system debugging.

Note
VX1152 is supported by VXtools version 5.0 and higher.

4.4.2 Lossless Compression

The video interface supports the hardware-based binary lossless compression of video streams. This compression method typically reduces the required bandwidth for video transmission, logging, and storage by 40 % ... 60 %, depending on the video content. Each channel of the interface can be compressed. The compression pipelines support the following pixel formats: RAW8, RAW10, RAW12, RAW16, and YUV422 8-bit. For other pixel formats, please contact Vector.

Four different compression modes are supported:

- **MobilEye on PC**
  This mode compresses the recording on the computer and requires a MobilEye license.

- **MobilEye on VX**
  This mode enables lossless compression on VX hardware and requires a MobilEye license. It reduces the bandwidth from the VX to the recording computer and offloads the necessary computing power.

- **Vector on VX**
  This mode enables lossless compression on the VX hardware and does not require a license. Only the payload of all received CSI long packets (all data types) is fed into the compression pipelines. The CSI headers and footers are stripped off beforehand. The embedded line payload is also fed into the compressor and reconstructed when decompressed. In this mode, all lines (embedded data, visible, and statistic lines) must be the same length as set in VXconfig. Enabling line length padding in the sensor may be necessary. To ensure good compression efficiency and enable later decoding, it is essential that all lines be the same length as the length set in VXconfig. The sensor may need to pad the embedded lines to the full length. Starting with VXtools V5.1, the interface enforces this.

- **Vector on VX V2**
  This mode has been supported since VXtools V5.1 and enables lossless compression on the VX hardware and does not require a license.

  If selected:

  - The CSI packets specified via CSI leading embedded packets and CSI trailing embedded packets of a frame are stored uncompressed.
  - The CSI headers and footers of all packets are also stored uncompressed.
  - The pixels are losslessly compressed on the interface hardware.
  - All three stored parts are secured via individual CRCs to detect data changes during transport or storage.

Each of the two compression pipelines is capable of compression:

- one stream with up to 350 MPx/s RAW8 ... RAW16
- one stream with up to 170 MPx/s YUV422

Vector provides decompression libraries and tools for Windows and Linux.

4.4.3 CRC Algorithm

```
std::vector<uint32_t> VCrc32::sCrcTable;

uint32_t VCrc32::Calculate(const uint8_t* rawData, const uint64_t rawDataS-
ize, const uint8_t bitsPerSample)
{
  uint32_t crc = 0xFFFFFFFF;
  const uint32_t specialWidth = 32 - bitsPerSample;
  const int andMask = (1 << bitsPerSample) - 1;
  uint32_t usedBytes = 0;
  const uint16_t* pData = reinterpret_cast<const uint16_t*>(rawData);

  while (usedBytes < rawDataSize)
  {
    /* XOR-in next input byte into MSB of crc and get this MSB, that's our
new intermediate dividend */
const uint16_t pos = static_cast<uint16_t>(((crc ^ (*pData << spe-
cialWidth)) >> specialWidth) & andMask);
    /* Shift out the MSB used for division per look up table and XOR with
the remainder */

    crc = (crc << bitsPerSample) ^ sCrcTable[pos];
    usedBytes += 2;
    pData++;

    if (rawDataSize % 2 != 0 && usedBytes == rawDataSize - 1)
    {
      uint16_t paddedValue = *pData;
      paddedValue = paddedValue & 0xFF;
      pData = &paddedValue;
    }
  }
  return crc;
}



void VCrc32::GenerateCrc32Table(const uint8_t bitsPerSample)
{
  if (!sCrcTable.empty())
  {
    return ;
  }

  const uint32_t specialWidth = 32 - bitsPerSample;
  const uint32_t elementCount = 1 << bitsPerSample;
  sCrcTable.resize(elementCount);

  for (uint32_t dividend = 0; dividend < elementCount; dividend++)
  {
    uint32_t currentByte = dividend << specialWidth;
    for (uint8_t bit = 0; bit < bitsPerSample; bit++)
    {
       if ((currentByte & 0x80000000) != 0)
      {
        constexpr uint32_t polynomial = 0x04C11DB7;
        currentByte <<= 1;
        currentByte ^= polynomial;
       }
       else
      {
         currentByte <<= 1;
       }
     }
    sCrcTable[dividend] = currentByte;
  }
}
```

4.4.4 Connectors

4.4.4.1 Front Side

Device connectors

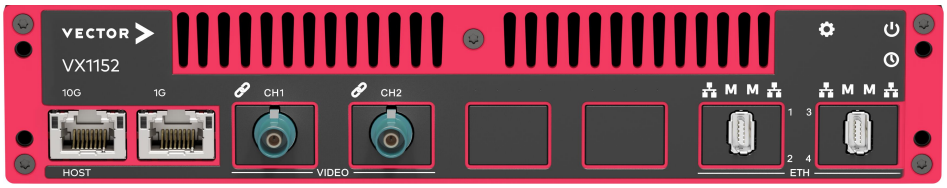


Figure 22: Connectors on the Ethernet side

- **1G/10G (RJ45 connectors)**
  Infrastructure port with different possible configurations:
  - Uplink:
    - During the configuration process, connect your computer via an 1G or 10G port with VXtools and the VX1152 via an RJ45.
    - For maximum performance, connect your logger to a 10G port during the measurement process.
  - Cascading: Connect additional interfaces and aggregate their data to a single uplink port.
  - PTP: Connect other devices to participate the same PTP timeTransmitter. The VX1152 can be configured as the PTP timesTransmitter.

Note
Loops between all Ethernet ports (10G, 1G and AEth) must be avoided under any circumstances.

Since all Ethernet ports are connected to the same internal Ethernet switch, creating a logical loop, also via external switches, causes data packets to loop at maximum speed, which makes the system unstable or even unresponsive.

To resolve this issue, remove the loop and restart the device.

Note
10 Gbps Ethernet connections are more sensitive to electromagnetic interference (EMI). Data integrity is guaranteed through the use of protocols, which may reduce data throughput.

- **Video CH1 and CH2**
  FAKRA coaxial Type Z connectors are used to connect the data source (sensor head) to CH1 and the data sink (ECU) to CH2. Type Z connectors are compatible with all other coding types of single-signal FAKRA connectors. The connector properties can be found in section Technical Data on page 169.

- **Ethernet ETH1...4 (ix Industrial)**
  ix Industrial connectors for 100BASE-T1/1000BASE-T1 (e. g. Harting ix Industrial type 10A-1). The channel characteristics, number of available channels and the pin assignments depends on the inserted VNmodule60.

- ETH1 and ETH2 are internally connected for an AEth TAP. Their configuration can be changed from master to slave or auto-detection with VXconfig.
- ETH3 and ETH4 are independent AEth ports. They are connected to the internal Ethernet switch and function as media converters for all RJ45 1G/10G BASE-T ports. Both ports operate in auto-detection mode.

4.4.4.2 Back Side

Device connectors

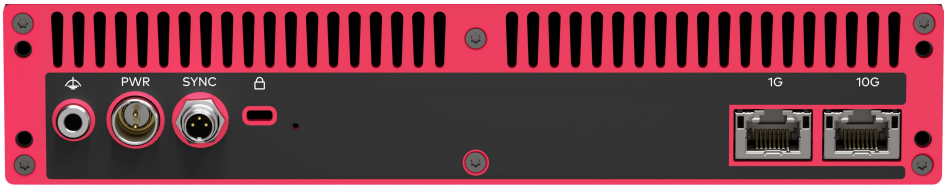


Figure 23: Connectors on the back side

- **Functional Earth (FE)**
  Optional connection of housing ground to system ground (see section Vehicle Battery Only (Closed System) on page 144).

- **Power (ODU)**
  For power supply, the VX1152 has a two-pin ODU connector (MINI-SNAP size 1, type GF1L0C-P02RP00-0000). Attach the enclosed power cable to power up the unit (matching ODU connector type S11L0C-P02NPL0-6200).

Pin	Assignment
1	Power supply (8 V ... 32 V)
2	Ground

- **Sync (Binder)**
  The VX1152 has a sync connector (Binder type 711) which can be used for time synchronization of different Vector devices. In addition, a pin is available for standby operation (see section Standby Operation on page 168).

Pin	Assignment
1	Terminal 15 (12 V ... 32 V) for standby operation
2	Synchronization line
3	Ground Sync

- **Kensington Lock**
  Mounting of Nano Kensington (NanoSaver).

- **1G/10G (RJ45 connectors)**
  Infrastructure port with different possible use cases:
  - Uplink:
    - During the configuration process, connect your computer via an 1G or 10G port with VXtools and the VX1152 via an RJ45.
    - For maximum performance, connect your logger to a 10G port during the measurement process.
  - Cascading: Connect additional interfaces and aggregate their data to a single uplink port.
  - PTP: Connect other devices to participate the same PTP timeTransmitter. The VX1152 can be configured as the PTP timeTransmitter.

4.4.5 Vehicle Battery Only (Closed System)

In this scenario, the individual Vector devices are powered directly from the vehicle battery. A third device is powered via USB (e. g. from a notebook in battery mode or from the vehicle battery) or via PCIe (card mounted in a computer and powered from the vehicle battery). The system is therefore closed.

The common reference ground is the chassis, which is connected to the negative terminal of the vehicle battery. Vector devices that have a functional earth connection (FE) must also be connected to the reference ground.

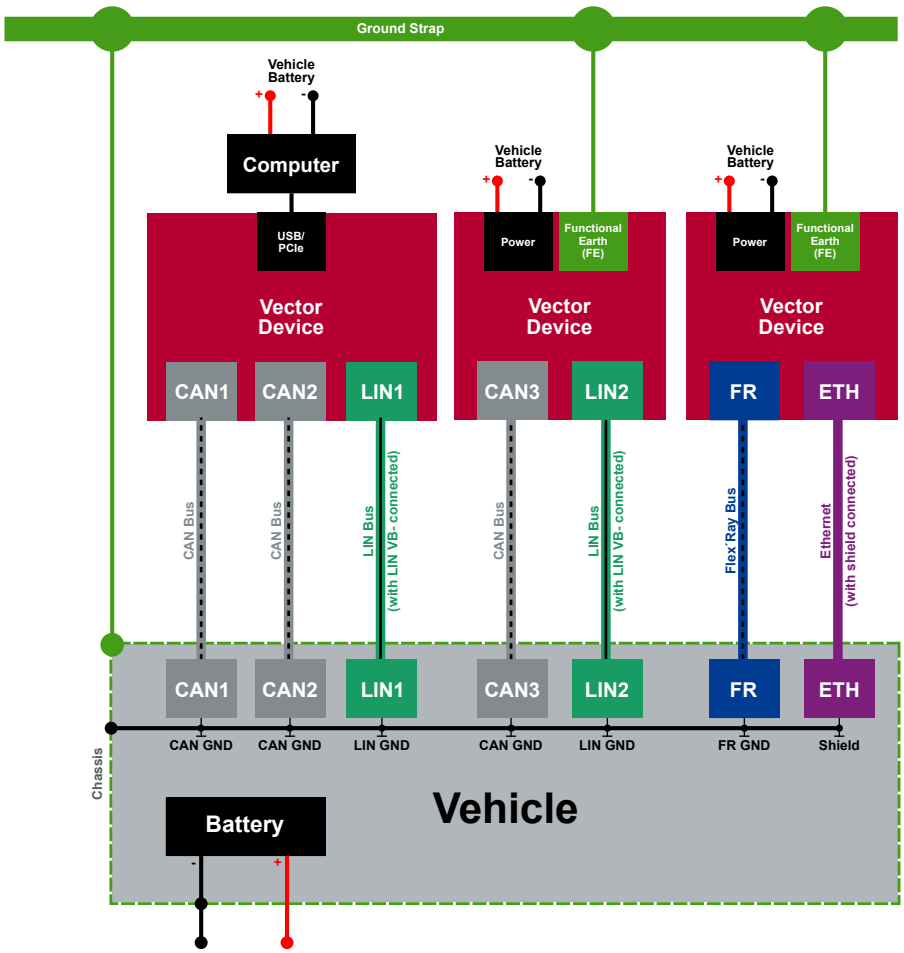


Figure 24: Vector devices powered by vehicle battery

The individual GNDs of the ECUs (in this example CAN, LIN, FlexRay) are connected to the chassis on the vehicle side. The shielding of the Ethernet connections is also connected to the chassis.

In this scenario, LIN VB- must be connected in the LIN signal path (LIN always requires VB- to be connected). For CAN and FlexRay it is recommended to connect all signal GNDs via their connectors (CAN GND, FlexRay VB-) to have the same reference ground from the vehicle battery.

4.4.6 LEDs

The front and back sides have different LEDs, which are described below.

4.4.6.1 Front Side

Power

Symbol	Sequence	Description
	R R R R R R R R	Device is powered but Firmware not running. The Power LED will light up red at the beginning of the boot sequence of the Interface Card. ► Please contact Vector VX support if the Power LED stays red for more than 10 seconds. In this case a critical error has occurred.
	G G G G G G G G	Firmware loaded successfully.
	G ☐ G ☐ G ☐ G ☐	Response to ”Identify VX device command” (VXconfig).

System

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Firmware not loaded.
	R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Interface Card, its system components are initialized. If a component fails to initialize properly, the System LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
	G G G G G G G G	Device is in normal operating mode.
	G ☐ G ☐ G ☐ G ☐	Firmware update in progress.

Time Sync

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	No modules to be synchronized.
	R ☐ R ☐ R ☐ R ☐	Problems during synchronization.
	R R R R R R R R	Synchronization error. At least one module not synchronized.
	G G G G G G G G	All modules synchronized.
	G ☐ G ☐ G ☐ G ☐	Synchronization in progress.

Master

Symbol	Sequence	Description
M	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Channel is in Slave mode
	R R R R R R R R	Channel is in Master mode 100Mbps
	G G G G G G G G	Channel is in Master mode 1000Mbps

Traffic

Symbol	Sequence	Description
	☐☐☐☐☐☐☐☐	Channel is not linked
	Y Y Y Y Y Y Y Y	Channel is linked at 100 Mbps; no traffic
	Y ☐ Y ☐ Y ☐ Y ☐	Channel is linked at 100 Mbps; data traffic
	G G G G G G G G	Channel is linked at 1000 Mbps; no traffic
	G ☐ G ☐ G ☐ G ☐	Channel is linked at 1000 Mbps; data traffic

Channel

Symbol	Sequence	Description
	☐☐☐☐☐☐☐☐	Channel disabled.
	R R R R ☐☐☐☐	Channel enabled, invalid configuration. See VXconsole for details.
	R R R R R R R R	Channel enabled, no link established.
	G G G G G G G G	Channel enabled, link established, no data traffic.
	G ☐ G ☐ G ☐ G ☐	Channel enabled, link established, data traffic.

10G Ports (RJ45)

LEDs illuminate if there is an Ethernet link or blink if there is Ethernet activity.

Color (Left)	Color (Right)	Description
Orange	-	100 Mbit/s
Green	-	1 Gbit/s
-	Orange	2.5 Gbit/s
-	Green	5 Gbit/s
Orange	Orange	10 Gbit/s

1G Ports (RJ45)

LEDs illuminate if there is an Ethernet link or blink if there is Ethernet activity.

Color (Left)	Color (Right)	Description
Green	-	Activity
-	Off	10 Mbit/s
-	Orange	100 Mbit/s
-	Green	1000 Mbit/s
-	Off	10 Mbit/s

4.4.6.2 Back Side

10G Ports (RJ45)

LEDs illuminate if there is an Ethernet link or blink if there is Ethernet activity.

Color (Left)	Color (Right)	Description
Orange	-	100 Mbit/s
Green	-	1 Gbit/s
-	Orange	2.5 Gbit/s
-	Green	5 Gbit/s
Orange	Orange	10 Gbit/s

1G Ports (RJ45)

LEDs illuminate if there is an Ethernet link or blink if there is Ethernet activity.

Color (Left)	Color (Right)	Description
Green	-	Activity
-	Off	10 Mbit/s
-	Orange	100 Mbit/s
-	Green	1000 Mbit/s
-	Off	10 Mbit/s

4.4.7 Video on VXmodule60

4.4.7.1 Channel in TAP Mode

The TAP is designed to be as transparent as possible to the ECU.

If a channel is configured in TAP mode, the incoming GMSL data stream is received by the MAX96792A TAP deserializer. The data payload (e. g. video or radar frames) is then duplicated. One copy is forwarded to the MAX96793 TAP serializer and one copy is forwarded to the recording data processor.

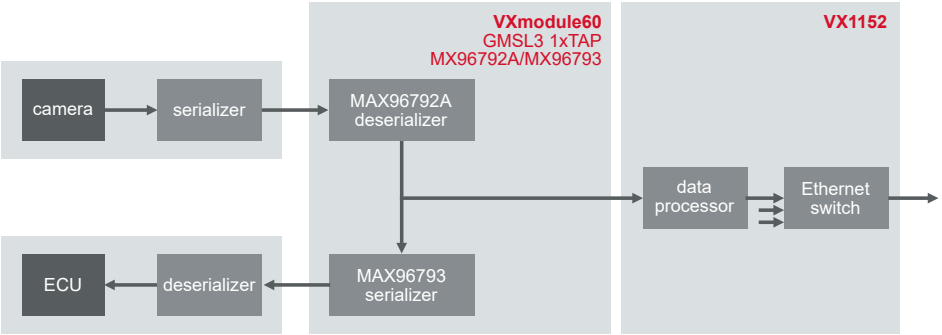


Figure 25: Channel in TAP mode; payload duplication

Control accesses from the ECU to the sensor via the I2C control channel or GPIOs are forwarded from the TAP serializer to the TAP deserializer.

TAP Initialization

The initialization sequence of the sensor and the ECU links follows this chart:

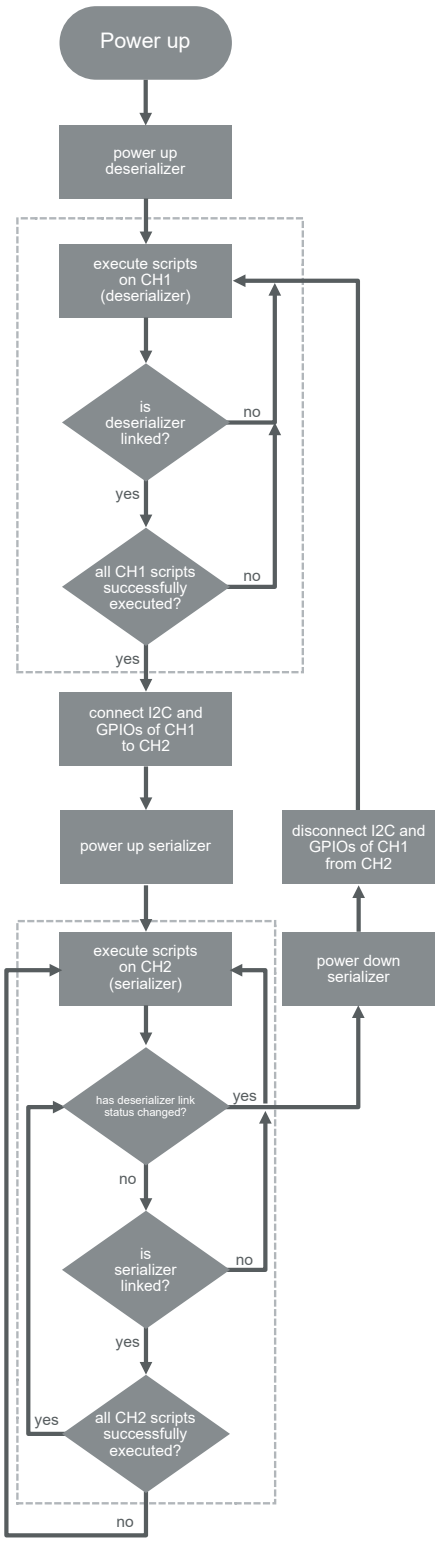


Figure 26: TAP initialization sequence

After power-up, the POC voltage is applied as configured and the deserializer is activated.

Next, the enabled scripts for the TAP RX channel (CH1/CH3) are immediately executed in an initial run (script 1, 2, 3) to configure the TAP deserializer, the sensor serializer and the sensor imager as early as possible.

After the initial script run on the TAP RX channel, the script execution engine waits for a TAP deserializer lock.

When the TAP deserializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the following conditions apply:

1. Script is not configured as "run once" (CASL SetConfig(7,1))
2. OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
3. OR script terminated with a return value ≠ 0 in previous run

If the TAP deserializer link unlocks, the script sequence repeats from the initial TAP deserializer run on, the I2C and GPIO bridges will be opened and the TAP serializer will be deactivated.

When the TAP deserializer locks, the I2C bus and GPIOs of the TAP serializer and TAP deserializer are connected. This allows the ECU to configure and control the sensor (see section GPIO Interconnect).

When the TAP deserializer locks, the TAP serializer is activated and the script execution on the TAP TX channel (CH2/CH4) are executed in an initial run (script 1, 2, 3).

After the initial script run on the TAP TX channel, the script execution engine waits for a TAP serializer lock.

When the TAP serializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the following conditions apply:

1. Script is not configured as "run once" (CASL SetConfig(7,1))
2. OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
3. OR script terminated with a return value ≠ 0 in previous run

If the TAP serializer link unlocks, the sequence repeats from the initial TAP serializer run on.

If the TAP deserializer link unlocks now, the I2C and GPIO bridges will be opened, the TAP serializer will be deactivated and the script sequence repeats from the initial TAP deserializer run on.

I2C Architecture

Without the TAP, the ECU is directly connected to the sensor head. Via the ECU deserializer and the sensor serializer it can configure and control all devices in that path. By inserting the TAP, the ECU deserializer is now connected to the TAP serializer and the sensor serializer is now connected to the TAP deserializer.

To allow the ECU controller to configure and control the sensor serializer and the sensor also in TAP mode, it might be necessary to configure the TAP serializer and the TAP deserializer via CASL scripts.

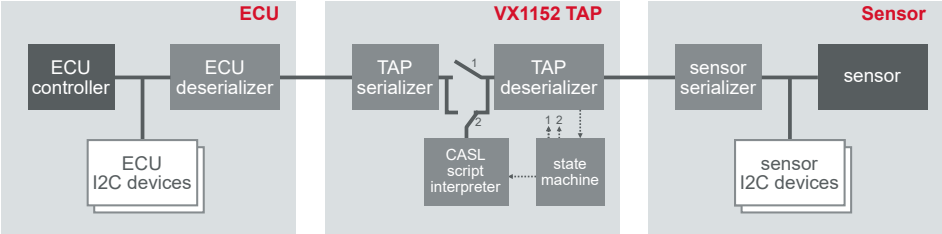


Figure 27: TAP I2C architecture; ECU configures ECU deserializer + TAP serializer + sensor

The I2C address of the TAP deserializer and the TAP serializer can be configured in VXconfig. Both chips start with their hardware strapped address and are then overwritten by the VX1152 firmware with the address configured in VXconfig. Additionally, the SetConfig(8,x) CASL command can be used to change the I2C address dynamically.

The sensor serializer I2C address is defined by its hardware straps. It can be changed via CASL scripts and must not be the same as the TAP serializer address.

The sensor I2C address is defined by the sensor itself. Accesses by the ECU will transparently be forwarded through the sensor link.

The additional TAP link will cause additional transmission delays (via clock stretching) to any sensor register access. These delays will reduce the effective control channel bandwidth and could affect system timings.

Detection of TAP by ECU Firmware for Measurement Mode

In cases where the ECU needs to know whether a TAP is present, CASL scripts can be used to manipulate unused spare default register values in the CAM serializer. The ECU can then read the register and check whether it contains the default value (no TAP) or the CASL script value (TAP node).

GPIO Interconnect

For forwarding GPIOs (general purpose input output pins) from the ECU to the sensor the TAP deserializer and the sensor serializer registers must be configured properly. This can be done via CASL scripts. This also allows static GPIO levels to be configured inside the deserializer.

Additionally, the VX1152 has a Sensor Trigger Generator which can generate synchronized trigger pulses on one GPIO pin per channel (see section Sensor Trigger Generator on page 357).

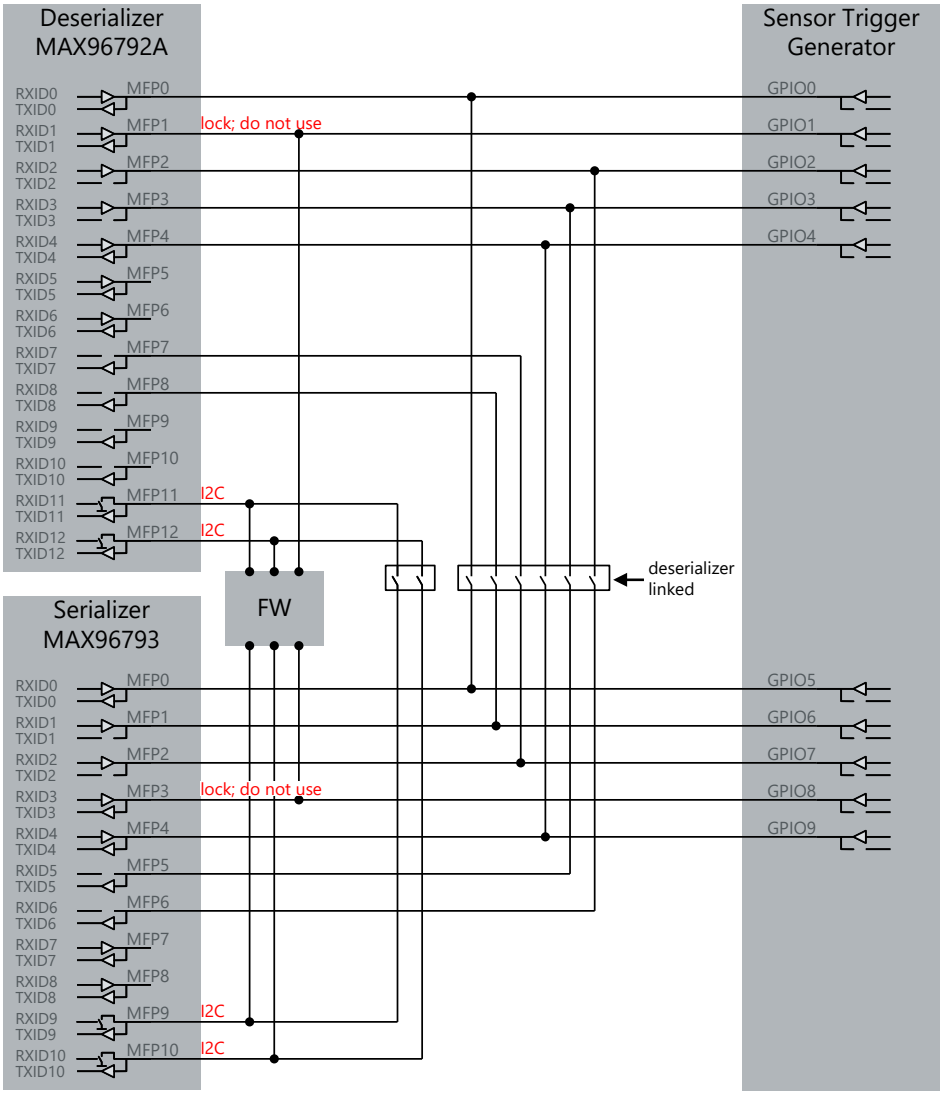


Figure 28: TAP MFP architecture; MFP connections between TAP deserializer, TAP serializer and Sensor Trigger Generator; switches are closed when TAP deserializer link is stable

Note
Care must be taken to avoid signal collisions between GPIO outputs and Sensor Trigger Generator outputs.

Note
MFP1 of the deserializer and MFP3 of serializer must stay configured as LOCK outputs. This function will be configured by the firmware at boot time. If the function of these pins is changed, the TAP will no longer be able to detect unlocks and script execution might be disturbed.

Forwarding of analog GPIO features is not supported.

The GPIOs of the TAP serializer are bidirectionally connected to the GPIOs of the deserializer and the Sensor Trigger Generator GPIOs as follows:

Signal of Serializer		Connected to Signal of Deserializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**	**Direction**	**Name**
CH2 - MFP0	IO	CH1 - MFP0	IO	CH2 - GPIO_5 = CH1 - GPIO_0
CH2 - MFP1	O	CH1 - MFP8	I	CH2 - GPIO_6
CH2 - MFP2	O	CH1 - MFP7	I	CH2 - GPIO_7
CH2 - MFP3	IO; LOCK			(CH2 - GPIO_8 do not use)
CH2 - MFP4	IO	CH1 - MFP4	IO	CH2 - GPIO_9 = CH1 - GPIO_4
CH2 - MFP5	IO	CH1 - MFP3	O	CH1 - GPIO_3
CH2 - MFP6	IO	CH1 - MFP2	O	CH1 - GPIO_2
CH2 - MFP7	IO			
CH2 - MFP8	IO			
CH2 - MFP9	I2C	CH1 - MFP11	I2C	do not use
CH2 - MFP10	I2C	CH1 - MFP12	I2C	do not use
		CH1 - MFP1	O; LOCK	(CH1 - GPIO_1 do not use)
		CH1 - MFP5	IO	
		CH1 - MFP6	IO	
		CH1 - MFP9	I	
		CH1 - MFP10	I	

Signal mapping of TAP deserializer GPIOs to trigger generator GPIOs; directions with view of deserializer (IO_OD: Input+open drain output); LOCK: Do not use. Signals required internally.

4.4.7.2 Channel in RX Mode

If a channel is configured in RX mode, the incoming GMSL2 data stream is received by the MAX96792A deserializer and forwarded to the recording data processing.

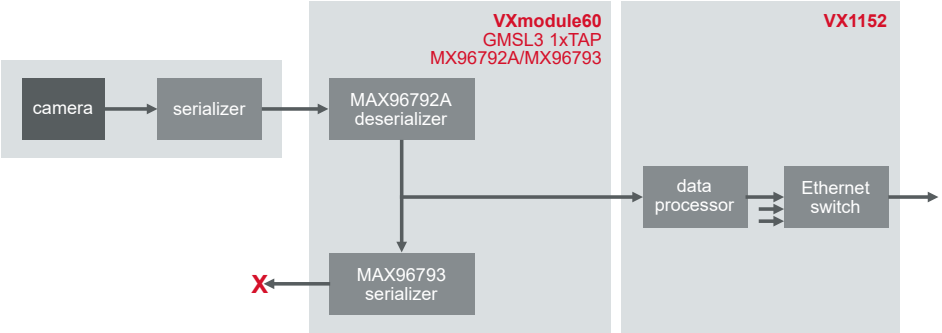


Figure 29: TAP channel in RX mode; payload reception

Initialization

The initialization sequence of the sensor link follows this chart:

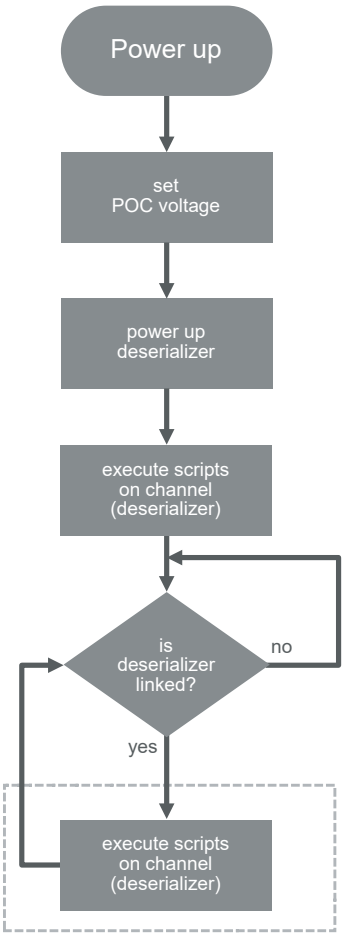


Figure 30: RX initialization sequence

After power-up, the POC voltage is applied as configured and the deserializer is activated.

Next, the enabled scripts for the RX channel are executed immediately (script 1, 2, 3) to configure the deserializer, sensor serializer and sensor as early as possible.

After the initial run, the script execution engine waits for a deserializer lock.

When the deserializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the follwing conditions apply:

- Script is not configured as "run once" (CASL SetConfig(7,1))
- OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
- OR script terminated with a return value ≠ 0 in previous run

If the deserializer link unlocks, the script sequence repeates from the inital deserializer run on.

I2C Architecture

In RX mode, the TAP deserializer, sensor serializer and sensor need to be configured via CASL scripts. A dynamic sensor reconfiguration is currently not supported.

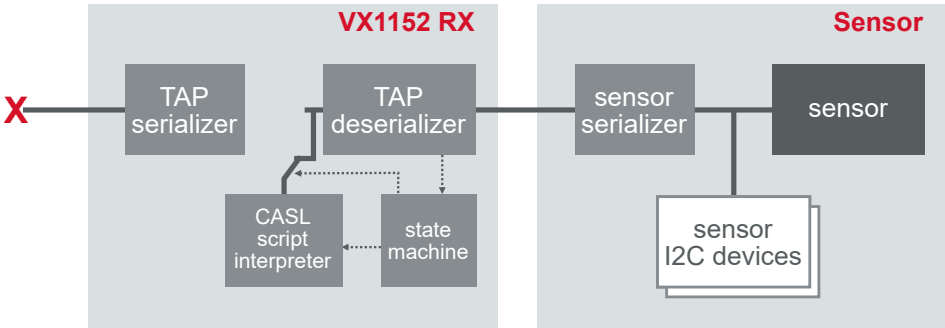


Figure 31: RX mode; I2C architecture; CASL scripts configure deserializer, serializer and sensor

The I2C address of the TAP deserializer can be configured in VXconfig. The chip starts with its hardware strapped address and is then overwritten by the VX1152 firmware with the address configured in VXconfig. Additionally, the SetConfig(8,x) CASL command can be used to change the I2C address dynamically.

The sensor serializer I2C address is defined by its hardware strapped pins and could be changed via CASL scripts.

The sensor I2C address is defined by the sensor itself. CASL scripts can be used to configure the sensor via I2C transactions. To enable I2C accesses to the sensor, deserializer and serializer registers might have to be set properly in advance.

MFP Interconnect

The VX1152 has a Sensor Trigger Generator which can generate synchronized trigger pulses on one MFP pin per channel.

For forwarding MFPs (multi function pins) from the Sensor Trigger Generator to the sensor, the TAP deserializer and the sensor serializer registers must be configured properly. This can be done via CASL scripts.This also enables the configuration of static MFP levels inside the deserializer.

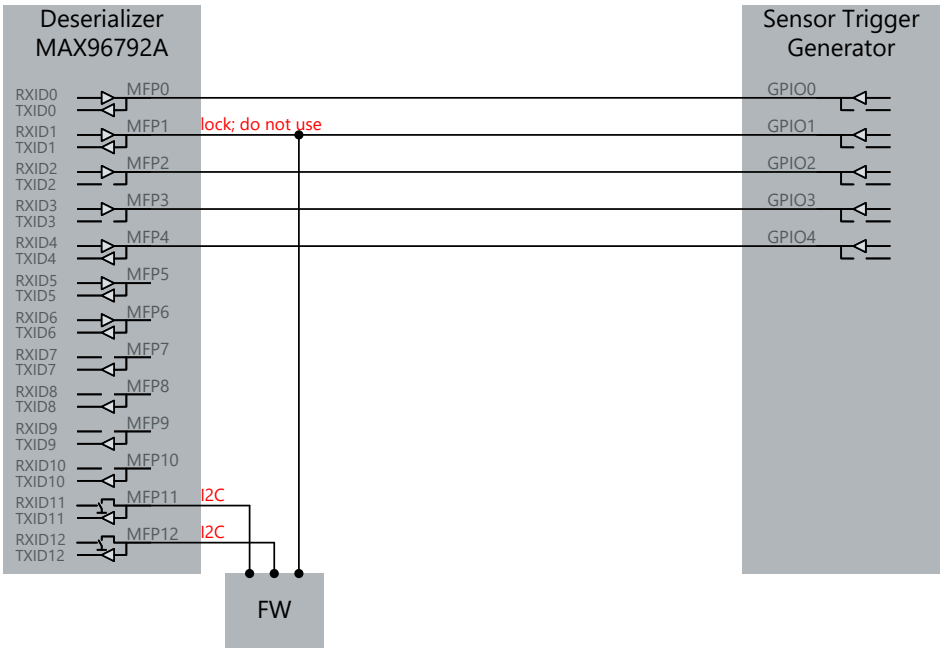


Figure 32: RX MFP architecture; MFP connections between TAP deserializer and Sensor Trigger Generator

Note
Care must be taken to avoid signal collisions between MFP outputs and Sensor Trigger Generator outputs.

Note
MFP1 of the deserializer must stay configured as LOCK output. This function will be configured by the card firmware at boot time. If the function of this pin is changed, the VX1152 firmware will no longer be able to detect unlocks and script execution might be disturbed.

On GMSL, the transmission of MFPs can be configured either in standard or delay compensation mode. For frame trigger signals, the delay compensated mode should be used to reduce jitter. Both modes can be configured via CASL scripts. The transmission of MFP signals takes about $t_{delayMFP}$ for a GMSL2 link.

The MFPs of the TAP deserializer connected to the MFPs of the Sensor Trigger Generator GPIOs as follows:

Signal of Deserializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**
CH1 - MFP0	IO	CH1 - GPIO_0
CH1 - MFP1	O; LOCK	(CH1 - GPIO_1 do not use)
CH1 - MFP2	O	CH1 - GPIO_2
CH1 - MFP3	O	CH1 - GPIO_3
CH1 - MFP4	IO	CH1 - GPIO_4
CH1 - MFP5	IO	
CH1 - MFP6	IO	
CH1 - MFP7	I	
CH1 - MFP8	I	
CH1 - MFP9	I	
CH1 - MFP10	I	
CH1 - MFP11	I2C	do not use
CH1 - MFP12	I2C	do not use

Signal mapping of deserializer MFPs to trigger generator GPIOs; directions with view of deserializer; LOCK: Do not use. Signals required internally.

4.4.7.3 Sensor Trigger Generator

The VX1152 provides a Sensor Trigger Generator for the video channel. This generator can be used to generate a trigger signal to one of the GPIO/MFP pins. If the serializer and deserializer are configured correctly via CASL scripts, this trigger signal will be forwarded to the sensor and triggers the acquisition of data.

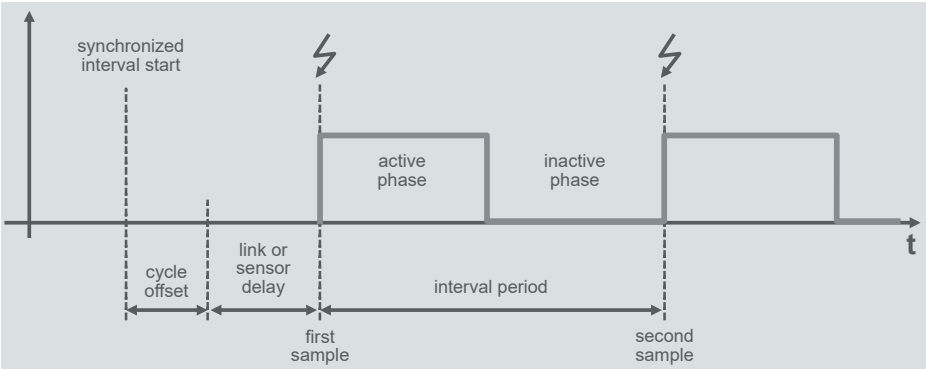


Figure 33: Sensor Trigger Generator with adjustable interval period

The generator uses the PTP synchronized hardware timer as the time base. When connected to the same network as a VX1161 rack with VX1161.52 cards, the VX1152 will generate triggers synchronously on all channels of all synchronized devices.

The current implementation requires all devices to be able to exchange Ethernet broadcast messages with each other at the start of the measurement. These messages are used to negotiate the precise start of the first Frame Trigger period.

If a measurement setup consists of multiple devices, a path must be established for broadcast messages to reach all the other devices. Since Windows does not natively forward Ethernet broadcast messages from a receiving port to all other ports, an additional connection is needed. An additional Ethernet cable is required to provide that path. The Ethernet switch inside the VX1152 and VX1161.22 uplink modules will then distribute the broadcast messages. Any Ethernet port can be used.

The generators start outputting their trigger signal at the start of the receiving application's measurement and stop generating it when the measurement stops. Consequently, the sensor may not generate data when the measurement stops. When the measurement begins, the time on all channels starts simultaneously, and all generators begin their cycle offset times.

The trigger signal has two phases: active and inactive. Both phases can be configured for different durations. Together, they define the trigger interval period, which determines the frequency of trigger pulses and the frame rate. Trigger generation always begins with the active phase. The cycle offset parameter can compensate for delays in serial links or sensors. For sensors on fast paths, a delay can be added so that they trigger at the same time as sensors on slow paths.

For sensors that require a falling edge to trigger, the polarity of the output signal can be selected. If the output signal is not inverted, the active phase starts with a rising edge, which is intended to trigger the sensor.

Note
Be careful to avoid signal collisions between the outputs of the serializer or deserializer and the outputs of the sensor trigger generator.

The Sensor Trigger Generator can be configured using VXconfig and the following parameters:

Setting	Description
Enable	Activates the Sensor Trigger Generator. If disabled, the GPIOs are high impedance.
Cycle Offset	Defines a static delay from the ideal interval start in nano seconds. Can be used to compensate different delays in between several links or sensors.
Active Phase	Defines the duration of the active phase in nano seconds.
Inactive Phase	Defines the duration of the inactive phase in nano seconds.
GPIO output pin to trigger	GPIO pin of the VX1152, where the trigger signal shall be output. Check the GPIO/MFP section of your card to see which pins can be used and how they map to deserializer or serializer pins.
GPIO polarity active phase	Can be used to invert the output signal polarity. - checked (default): active phase starts with a rising edge. Active phase is high. Inactive phase is low. - unchecked: active phase starts with a falling edge. Active phase is low. Inactive phase is high.

4.4.7.4 POC Supply

The VX1152 delivers power to connected sensor heads via the coax cable using Power-over-Coax (POC). It has an internal, isolated supply that is adjustable and independent of VX GND. It can source up to 2 W. The voltage can be switched off or adjusted from 5 V ... 12 V using VXconfig.

In TAP mode, the internal supply can be deactivated, allowing the POC supply of the ECU to be forwarded to the sensor head. This gives the ECU control over the sensor's power.

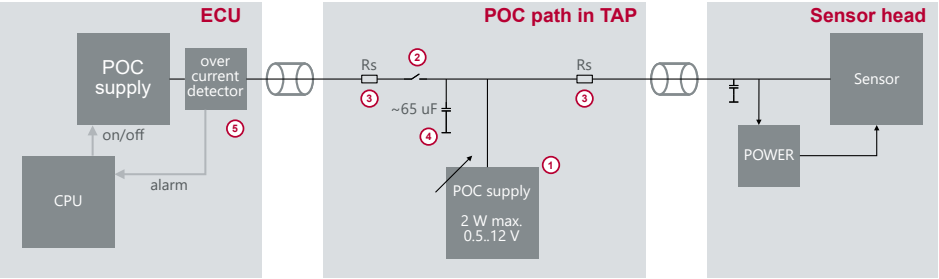


Figure 34: POC structure in VX1152 TAP channels

The internal series resistance (Rs) (3) of the POC bypass path is approximately 2 x 2.5 Ohms, and the internal capacitance (4) is approximately 65 µF. This capacitance can cause or increase inrush currents for the ECU that sources power and might trigger overcurrent detection features (5). In such cases, it might be an option to source the sensor from the VX (1) instead of the ECU, but then the ECU would have no control over the sensor power.

Furthermore, the internal VX POC supply can be output to the ECU for use cases in which CSI ingestion PODs require POC supply.

The POC voltage level, generation, and delivery direction are controlled via the POC voltage settings for the TAP_RX and TAP_TX ports in VXconfig as follows:

Note
Sourcing power in TAP mode is supported since VXtools v4.9.

Settings				
POC voltage V_{RX}on TAP RX channel	0V	5...12V	0V	5...12V
POC voltage V_{TX}on TAP TX channel	0V	0V	5...12V	5...12V
Effects				
POC bypass (2)	closed	open	closed	closed
VX POC source (1) voltage	off	V_{RX}	V_{TX}	Min(V_{RX},V_{TX})
Use case	ECU supplies sensor	VX supplies sensor	VX supplies ingest POD	VX supplies sensor and ingest POD

4.4.7.5 VX1152 Bandwidth Limits

This section provides an overview of bandwidth constraints for the VX1152 streaming system. These constraints define the maximum possible data rates that can be transferred.

For an error free operation without data loss or data corruption the aggregated data rates at every point of the data path must not exceed the specified limits.

Special attention must be paid to the calculation of video bandwidth losses due to inter packet delays (e. g. blanking) and coding overheads.

Additionally, if burst data rates (e. g. CSI packet transfers) exceed individual limits, internal buffer sizes as well as input and output bandwidths must be considered.

Contact Vector for a reliable system evaluation.

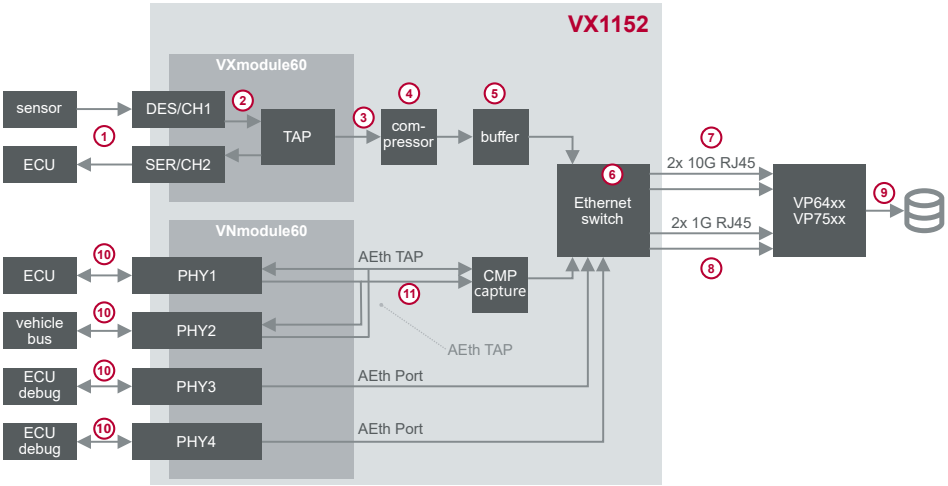


Figure 35: Bandwidth limits (Note: Figure for documentation purposes only; does not represent actual individual implementation architectures)

Item	Interface Limits
1	Cable limit: VXmodule60 GMSL3 1xTAP MX96792A/MX96793: 12 Gbps (line rate); 10 Gbps (net), reduced by CSI low power state transitions and blanking times (time between packets); typically -10 % ... -20 %
2	CSI lane speed limit: 4x 2.5 Gbps, reduced by CSI low power state transitions and blanking times (time between packets); typically -10 % ... -20 %
3	VXmodule60 to baseBoard limit: 24 Gbps
4	Lossless compressor limit (optional): - one stream with up to 700 MPx/s RAW8...RAW16, - one stream with up to 340 MPx/s YUV422
5	System memory bandwidth limit (RX+TX) (after compressor): 68 Gbps
6	Ethernet switch: all ports at full bandwidth
7	10G Base-T RJ45; Host Uplink;cascading, PTP; limit:2x 10Gbps
8	1G Base-T RJ45; other interfaces, PTP;

Item	Interface Limits
	limit:2x 1Gbps
9	Logging PC limit: - VP64xx: up to 12 Gbps - VP75xx: up to 20 Gbps These limits depend heavily on the measurement setup and must be verified by prototyping.
10	4x Automotive Ethernet on VNmodule60 4x 1 Gbps; max cable speed; both TX and RX
11	Automotive Ethernet TAP capture limit: 2x 1.1 Gbps; data from both directions

Gbps: Giga bits per second
(net): maximum effective payload data rate

Video
The possible bandwidth for receiving streaming data depends on the limits of the transfer paths in the chain. If several streams use the same path, the total data rate must be able to pass through it.

When starting at the sensor serializer, the serial protocol used (1) defines the link limit. The maximum net link bandwidths listed in the transceiver datasheet apply. When using the VX1152 in TAP mode and the link speeds on the ECU and sensor differ, the lower speed determines the maximum burst rate and minimum time between packets (blanking).

The TAP deserializer on the VXmodule60 receives stream data from the sensor serializer and outputs it towards the TAP electronics via a CSI interface (2). This interface always uses four CSI data lanes. The CSI lane rate can be configured in the **Channel Group** configuration in VXconfig.

CSI data streams are encapsulated in short and long packets, which are separated by low power states (idle times for vertical and horizontal blanking) and contain headers and footers. This overhead usually accounts for 10% ... 20 % of the gross CSI bandwidth and depends mainly on your sensor's blanking time settings.

The total bandwidth that the VXmodule60 can forward to the BaseModule is 24 Gbps (3).

The VX1152 can compress RAW streaming data (2) in binary lossless hardware, which relieves all succeeding data paths and the recording device of high bandwidths. The compression pipeline (4) is capable of compressing up to ~700 megapixels per second (MP/s) in RAW8/10/12/16, or approximately 350 MP/s for YUV422 8-bit. The compressor's output data rate depends on the contents of the incoming data and is typically 40 % ... 60 % of the input data rate.

The compressed video data is stored in an internal buffer with a total bandwidth of approximately 68 Gbps.

Automotive Ethernet
The VNModule60 has four transceivers, each of which can transmit up to 1 Gbps in the RX and TX directions (10).

The TAP connects two transceivers, capturing and encapsulating data in both directions in the CMP protocol. This data source can generate up to approximately 2.2 Gbps. Both Ethernet ports can also transfer 2 Gbps per port (11).

1G and 10G BASE-T RJ45

The 1G (8) and 10G (7) Base-T RJ45 ports are connected to the Ethernet switch. If one 10G connection is insufficient, multiple cables can be used. In this case, manually balance the load by assigning the correct subnet masks and IP addresses to the different ports. However, the load of the highest bandwidth source, the video stream, cannot be split. Therefore, additional streams must be distributed.

Ethernet Switch
The Ethernet switch (6) connects all ports and data sources. It can transfer data on all ports simultaneously at their maximum bandwidth.

VP loggers
Depending on the storage bandwidth and size, one or more loggers must be used.

4.4.7.6 GMSL2 Compability

GMSL3 mode

The TAP deserializer and TAP serializer of the VXmodule60 start up in default GMSL3 mode. To use them in GMSL2 compatibility mode, add the following lines to the beginning of your CASL scripts in VXconfig:

CH1 (Deserializer)

```
WriteIO(0x01, 0x02); // Set GMSL2 6 Gbps Mode for Port A
//WriteIO(0x01, 0x01); // Set GMSL2 3 Gbps Mode for Port A
WriteIO(0x28, 0x60); // Disable FEC on link A if sensor serializer does not
support it (e.g. MAX9295) or FEC is disabled
WriteIO(0x06, 0xDF); // GMSL2 on link A+B
WriteIO(0x04, 0x83); // GMSL2 for Port A | GMSL3 12 Gbps mode for Port B
WriteIO(0x10, 0x31); // One Shot Reset
```

CH2 (Serializer)

```
WriteIO(0x01, 0x08); // Set GMSL2 6 Gbps Mode for Port A
//WriteIO(0x01, 0x04); // Set GMSL2 3 Gbps Mode for Port A
WriteIO(0x28, 0x60); // Disable FEC on link A if ECU deserializer does not
support it (e.g. MAX9296) or FEC is disabled
WriteIO(0x06, 0x91); // GMSL2 on link A
WriteIO(0x10, 0x31); // One Shot Reset
```

4.4.8 Automotive Ethernet on VNmodule60 (optional)

Introduction

The VX1152 can carry an optional VNmodule60 4AE1G 88Q2221M. This module carries four 100/1000 BASE-T1 Automotive Ethernet transceivers (PHYs).

Transceivers 1 and 2 are connected back-to-back to form one TAP channel. This TAP channel can be transparently inserted into the ECU's vehicle Ethernet bus and capture all data in both directions. The data is output via the CMP protocol.

Note

- The captured data of the TAP channel is output via the CMP protocol.
- The CMP protocol is configured via VXconfig.
- The VX1152 does not support the Vector XL API and cannot be configured via the Vector Hardware Manager.

Transceivers 3 and 4 are AEth ports which are connected to the internal Ethernet switch. These ports can function as media converters via the device's 1G and 10G BASE-T ports. These ports are intended for connecting to the video SoC's debug port (e. g., MobilEye TAPI).

Note

Loops between all Ethernet ports (10G, 1G and AEth) must be avoided under any circumstances.

Since all Ethernet ports are connected to the same internal Ethernet switch, creating a logical loop, also via external switches, causes data packets to loop at maximum speed, which makes the system unstable or even unresponsive.

To resolve this issue, remove the loop and restart the device.

Note

Do not connect the Automotive Ethernet Ports (media converter ports) to the vehicle Ethernet bus!

These ports are connected to the infrastructure network switch, so they will receive and inject traffic from all tools connected to the 1G and 10G RJ45 ports.

PHY configuration

The phyiscal PHY properties of the AEth TAP can be configured in VXconfig.

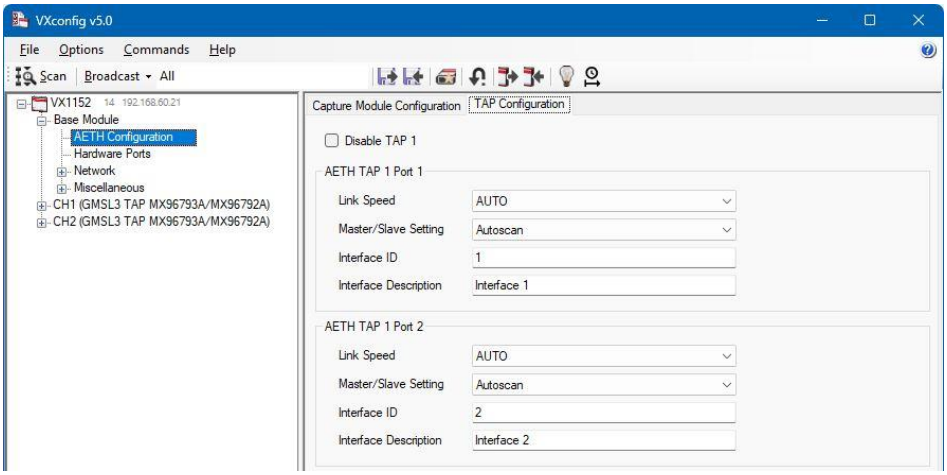


Figure 36: AEth TAP; PHY configuration in VXconfig

Use the checkbox in the **TAP Configuration** tab to enable or disable the TAP. Then, configure the link speed, master/slave setting, and stream ID. To connect to the ports receiving data, the port stream ID must be referenced in the CMP logging tool. With two IDs, data in both directions can be referenced individually.

CMP configuration

You can configure the settings of the Capturing Module in VXconfig.

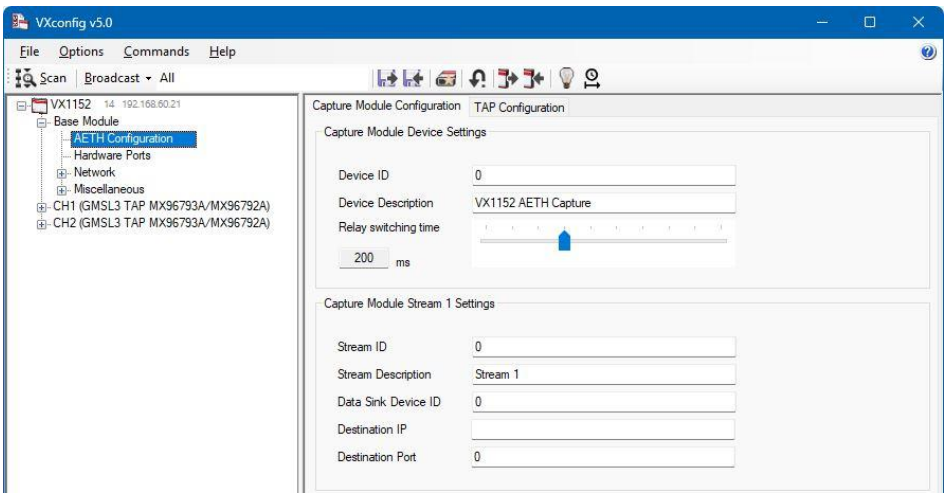


Figure 37: AEth TAP; CMP Capture Module Configuration in VXconfig

4.4.9 Device Stacking

This Vector device can be directly stacked or mounted side-by-side with other Vector devices that also have a Vector System Housing (VSH). This is supported by the side recesses on the housing.

Figure 38: Stacked mounting (example with VN5650)

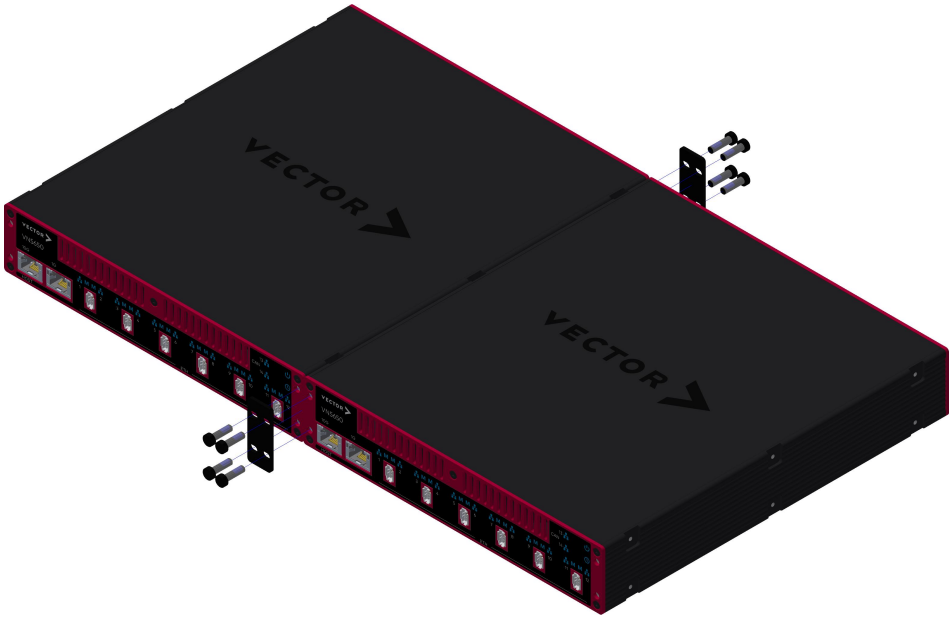


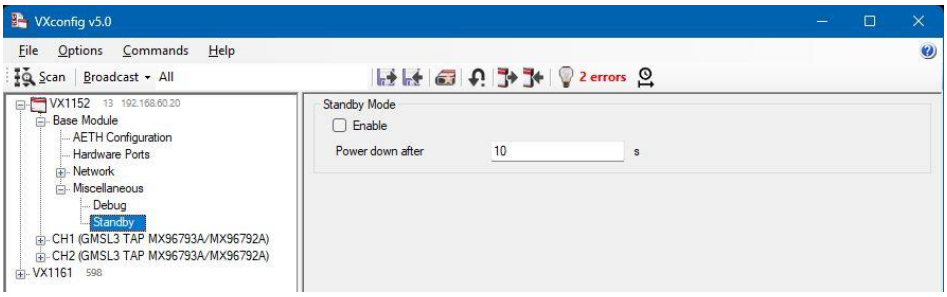
Figure 39: Side-by-side mounting (example with VN5650)

To further increase the mechanical stability of stacked devices, the mounted feet can be removed and replaced by the VSH Connecting Kit Vertical (see accessories manual, part number 05152).

4.4.10 Standby Operation

Activating the standby feature

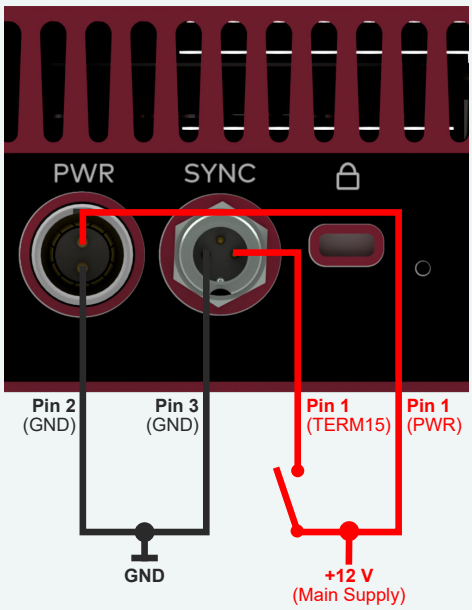
The VX1152 has a standby mode that shuts down and wakes the device remotely via a digital signal. This feature can be activated and configured with a shutdown delay time in VXconfig.



When enabled, if the sleep signal on pin 1 of the SYNC connector is connected to ground (GND), the device will enter standby mode after the configured time. The device will wake up again as soon as the sleep signal is pulled to 12 V ... 32 V (typical Vbat of terminal 15).

Example

The following example shows how to connect the SYNC signal. Note that a voltage of at least 12 V is required at terminal 15 to power on the device. The ground connection must have the same potential as the power supply ground.



4.4.11 Technical Data

Host/Logger interface	Ethernet (1/2.5/5/10GBASE-T); RJ45
Power supply	External: 8 V ... 32 V (typ. 12 V DC)
Power consumption	Max. 2.5A @ 12 V (65 °C) Standby: 2 mA @ 12 V
Temperature range	Operation: -40 °C ... +65 °C Storage: -40 °C ... +85 °C
Relative humidity of ambient air	15 %...95 %, non-condensing
Dimensions (LxWxH)	250 mm x 216 mm x 42 mm
Weight	Approx. 2.5 kg
POC (power over cable)	► V_{poc}; voltage sourced by VX1152: 0 V, 5...12 V configurable ► P_{poc}; maximum power delivered per RX channel; sourced by VX1152: 2 W ► $V_{PocTapMax}$; voltage forwarded from ECU by TAP: 3...18 V ► $I_{PocTapMax}$; max. current forwarded from ECU by TAP: 500 mA ► Rpoc; DC resistance of POC forwarding path in TAP: typ. 2.6 Ohm
Connector	(Single signal) FAKRA Type Z; color: cyan; compatible to all other (single signal) FAKRA connector types; 6 GHz max; 25 mating cycles
Maximum cable length	Video: 8m coax, on both RX and TX ports at the same time
Part number	180231

4.4.12 Accessories

- AEcable 2Y (page 661)
- Vector Power Supply ODU Mini-Snap (page 646)
- Cable Vehicle Input <> ODU Mini-Snap (page 650)
- Cable Banana Plug <> ODU Mini-Snap (page 653)
- Vector SYNCcableXL (page 656)
- FAKRA Cables (page 636)
- VSH Cable Guard 216 (page 678)
- VSH Cable Guard 216 Clip (page 681)
- VSH Connecting Kit 19" (page 682)
- VSH Connecting Kit Horizontal (page 684)
- VSH Connecting Kit Vertical (page 685)
- VSH Equipment Foot Kit (page 686)
- VSH Mounting Flange (page 687)

4.5 VX1161 Multi Base Module

4.5.1 General Information

Overview

Its modular and individually configurable setup makes the VX1161 Multi Base Module the ideal solution for measurement and calibration of many ECUs in parallel.

The VX1161 Multi Base Module consists of a VX1161.01A Base chassis which can be populated with a combination of cards tailored to a specific measurement use case.

The VX1161.01A Base provides 8 card slots. One slot is dedicated for the Power Supply Card, another for the Host Uplink Card. The remaining slots are designed to hold any of the Interface Cards or can be left unpopulated.

Figure 40: VX1161 Multi Base Module

4.5.2 Inserting or Removing VX1161 Cards

4.5.2.1 General Notes

Note
Before inserting or removing VX1161 cards, disconnect the device from the power supply. To prevent product damage, appropriate ESD protection measures must be observed.

Figure 41: VX1161.01A Base with one halfway inserted card

4.5.2.2 Removing VX1161 Cards

Step by Step Procedure

1. Loosen the captive screws in the corners of the front panel with a Torx T8 screwdriver.
2. The cards are plugged into the backplane of the VX1161.01A Base. To release them, proceed as follows:
   - If an adjacent slot is empty, a card can be detached from the backplane by gently holding and pulling its front panel.
   - Otherwise, plug a suitable cable into one of the connectors of the card, then carefully pull the cable until the card is released from the backplane.
3. Hold the card by the sides of its front panel and extract it from the VX1161.01A Base along the guide rails.
4. Store the card in a safe place, protected from ESD, humidity and environmental stress.

4.5.2.3 Removing Slot Covers

Step by Step Procedure

1. Loosen the captive screws of the slot cover with a Torx T8 screwdriver.
2. Use pliers to lightly seize one of the screw heads and loosen the slot cover by pulling gently.
3. Hold the slot cover by its sides and remove it.

4.5.2.4 Installing VX1161 Cards

Step by Step Procedure

1. Hold the card by the sides of its front panel and insert it into the VX1161.01A Base as far as possible without exerting force. While doing so, ensure that the card is correctly seated in the guide rails at the top and bottom.
2. Carefully apply pressure on the front panel until the card snaps into the back-plane:
   - During installation, it may be necessary to manually adjust the EMC shielding located on the right-hand side of the front panel.
   - If installing several cards, insert them one after the other from left to right. In this way the EMC shielding is automatically adjusted correctly.
3. Tighten the captive screws on the front panel with a Torx T8 screwdriver (recommended torque: 0.3 Nm).

4.5.2.5 Installing Slot Covers

Step by Step Procedure

1. Align the right edge of the slot cover to be inserted with the adjacent front panel or slot cover, so that the EMC shielding is properly fitted.
2. Press the left side of the slot cover against the housing.
3. Tighten the captive screws on the front panel with a Torx T8 screwdriver (recommended torque: 0.3 Nm).

Note
Slot covers for the VX1161 can be ordered as part [22505] "Slot Cover for VX1161".

4.5.3 VX1161.01A Base

4.5.3.1 General Information

The VX1161.01A Base is the chassis of the VX1161 Multi Base Module, providing 8 slots for VX1161.xx Cards.

Figure 42: VX1161.01A Base

- Slot count starts on the left with slot 1. Slot 1 is required to be populated with a VX1161.1x Power Supply Card.
- Slots 2 to 7 can be populated with any VX1161.3x/4x/5x Interface Card, VNx1xx Network Interface Card, VIO IO Card or left unpopulated.
- Slot 8 is required to be populated with a VX1161.2x Host Uplink Card.

The VX1161.01A Base provides active cooling for the inserted cards up to an ambient temperature of +60 °C using three speed-controlled fans. They force an airflow from the perforated bottom of the chassis via any inserted cards to the outlets in the back of the chassis where the fans are mounted.

WARNING!
The VX1161 Base Module has an active ventilation. Therefore the air inlet on the bottom and the outlet on the back side of the housing must be kept clear. To ensure this, the device feet as well as the backside handles must not be removed and the device has to be placed on an even ground.
Blocking the airflow can cause an over-temperature, leading to an automatic shutdown. This can also shorten the lifetime of the device.

WARNING!
The VX1161 Base Module must not be mounted with the back side facing downwards. All other mounting orientations are permitted.

4.5.3.2 Technical Data

Dimensions (W/D/H)	342 x 256 x 95 mm
Weight	2370 g (no cards inserted)
Temperature range	-40 °C ... +60 °C
Part number	180127

4.5.4 VX1161.11 Power Supply

4.5.4.1 General Information

The VX1161.11A Power Supply Card for the VX1161 Multi Base Module is an automotive-graded power supply for all other cards. The input is protected against reversed polarity.

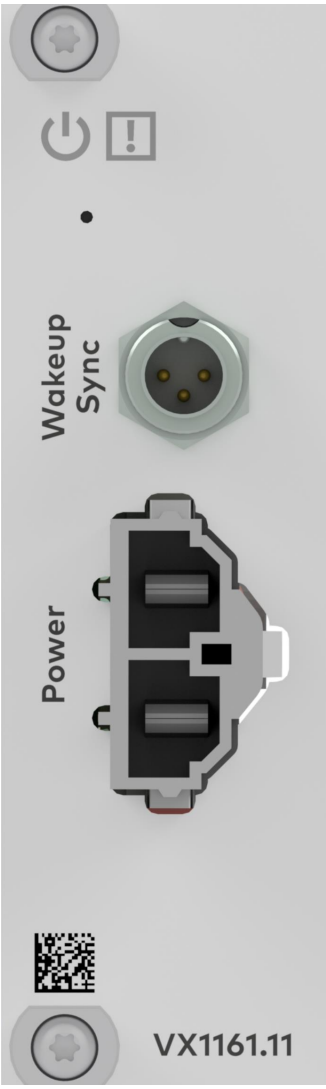


Figure 43: VX1161.11 Power Supply

4.5.4.2 Connectors

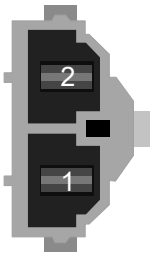
- **Wakeup/Sync**

Pin	Assignment
1	Wakeup: If enabled through configuration, applying VBAT will wake the VX1161 Multi Base Module and prevent it from entering standby.
2	HWSYNC: Hardware time synchronization allows the Measurement and Calibration tool to synchronize the timestamps of different devices to a common time base with a precision of 1 µs. For synchronization of VX devices a two-wire connection (signals: HWSYNC, GND_SYNC) between all devices is required. At each falling edge on the Sync line the VX device generates a timestamp that is provided to the Measurement and Calibration tool via XCP or other protocols, depending on the cards.
3	GND_SYNC: For HWSYNC.

Note
Reference for the wakeup signal is GND from the power connector! Do not connect GND and GND_SYNC which are isolated by design!

- **Power**

Pin	Assignment
1	GND (for VBAT and Wakeup)
2	VBAT



Note
Matching counterpart:
- Power Cable Molex Mini-Fit (part number 22515) (always included)
- Desktop Type Power Supply (part number 05116) (optionally available)

4.5.4.3 LEDs

Power

Symbol	Sequence	Description
⏻	R R R R R R R R	Emergency Shutdown mode (Overload, Over-temperature).
	G G G G G G G G	Normal operation mode.
	R □ □ □ □ □ □ □	Standby Mode is active.

Critical System Error

Symbol	Sequence	Description
!	R R R R R R R R	A critical system state has been detected and the system has been shut down.

4.5.4.4 Technical Data

Weight	184 g
Input voltage range	8 to 34 VDC
Power consumption	125 W max, plus 6 % conversion loss within the card The VX1161.11 Power Supply card provides the other cards in the VX1161 with a power budget of 125 W. The actual power consumption depends on the cards inserted in the VX1161. Standby: <10 mA @ 12 V (typical at room temp.)
Temperature range	-40 °C ... +60 °C
Part number	22463

4.5.5 VX1161.22A Host Uplink (obsolete)

Note
The VX1161.22A Host Uplink Card is discontinued.
This means that the card is still supported by the VXtools and also still gets updates with new features and bug fixes but the hardware is no longer sold.
Successor is the VX1161.22B Host Uplink Card.

The VX1161.22A Host Uplink Card for the VX1161 Multi Base Module is the central data distribution node of the VX1161 Multi Base Module.

All communication within the VX1161 Multi Base Module and to external network nodes is routed through this card. The VX1161.22A Host Uplink Card contains a PTP-enabled Ethernet switch.

On the front panel of the card, there are two 10 Gbps Ethernet ports (RJ45) which can be used as Host Uplink or for Cascading other high-performance network hardware, and one RJ45 port supporting 1 Gbps Ethernet or 100BASE-T1 (auto-sensing). The VX1161.22A Host Uplink Card can also host Microservices.

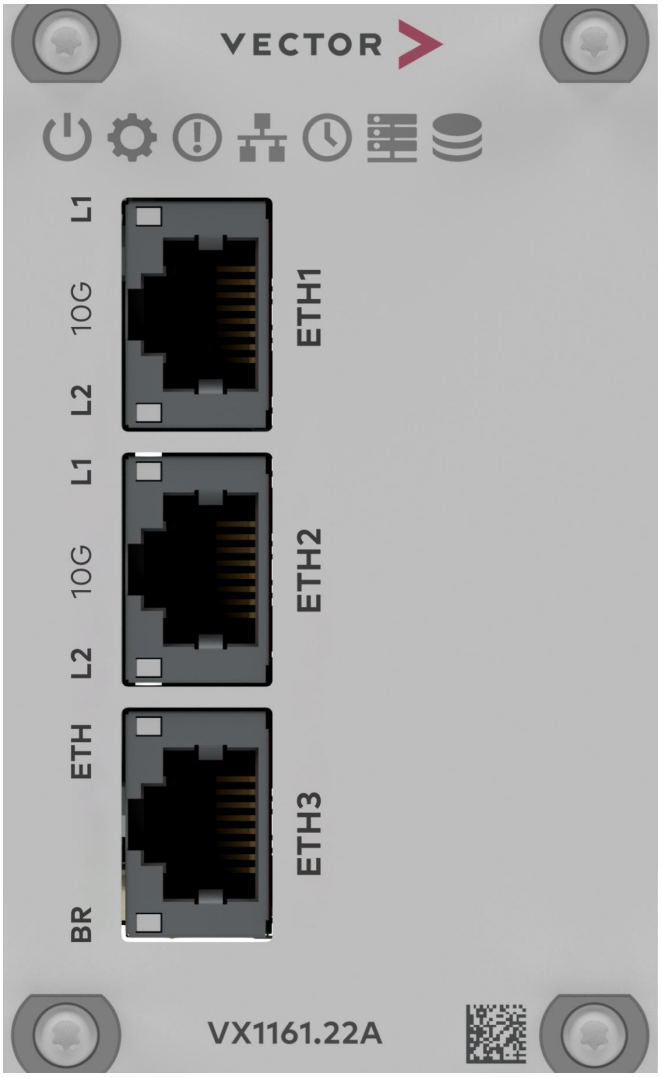


Figure 44: VX1161.22A Host Uplink Card

4.5.5.1 Connectors

- **ETH1 Host**
  10 Gbps Ethernet port with RJ45 connector. This port is intended as uplink to the Tool PC.The VX1161 Multi Base Module can be configured to enter standby if the link partner at this Ethernet port is missing.

  LEDs light up permanently if there is an Ethernet link or blink if there is Ethernet activity.

L2	L1	Description
-	Yellow	100M
-	Green	1G
Yellow	-	2.5G
Green	-	5G
Yellow	Yellow	10G

- **ETH2 Cascade**
  10Gbps Ethernet port with RJ45 connector. This port can either be used to cascade other VX1161 Multi Base Modules or other Ethernet hardware, or it can be used as second uplink to a Tool PC.

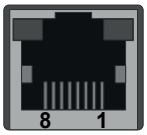
  LEDs light up permanently if there is an Ethernet link or blink if there is Ethernet activity.

L2	L1	Description
-	Yellow	100M
-	Green	1G
Yellow	-	2.5G
Green	-	5G
Yellow	Yellow	10G

- **ETH3 Target**
  1 Gbps Ethernet/100BASE-T1 port with auto media sensing. This port can be used to attach XPODs, VX1000 Base Modules or other Ethernet hardware. The intended data flow is from the Interface Cards or from this Ethernet port to the Tool PC attached to the Host port (ETH1), meaning that the Host Uplink Card acts as a link concentrator towards the Tool PC. As the Host Uplink Card contains a fully-featured Ethernet switch, other topologies are also possible.

  Standard Ethernet connector (RJ45) for 100BASE-TX, 1000BASE-T or 100BASE-T1.

Pin	Assignment for 100BASE-T1
1	MDI+
2	MDI-
3...8	Not connected



WARNING!
Please note that VX1000 Ethernet ports supporting both 1000BASE-T and 100BASE-T1 are only intended as interface to measurement and calibration hardware, like an XPOD. They are not designed to be connected to automotive Ethernet networks.

These ports are part of an Ethernet switch connecting the VX device itself, and any network node or network segment attached to ETH ports of the VX device. Such, the ETH ports may be exposed to unexpected Ethernet traffic, like broadcasts, which might not be tolerable in an automotive Ethernet network. Also, due to supporting both, standard Ethernet, and 100BASE-T1 on the same port, the electrical circuit slightly differs from what is common in automotive Ethernet networks, which also might not be tolerable in an automotive Ethernet network.

To communicate with automotive Ethernet networks, please use dedicated Vector Network Interface Hardware.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off Y = yellow G = green

BR/AE
LED indicating the 100BASE-T1 status.

Sequence	Description
G G G G G G G G	100BASE-T1 link is up with VX1161.22A as Master
G □ G □ G □ G □	100BASE-T1 activity
Y Y Y Y Y Y Y Y	100BASE-T1 link is up with VX1161.22A as Slave
Y □ Y □ Y □ Y □	100BASE-T1 activity

ETH
LED indicating the Ethernet status.

Sequence	Description
G G G G G G G G	Ethernet link is up with 100 Mbit/s
G □ G □ G □ G □	Ethernet activity at 100 Mbit/s
Y Y Y Y Y Y Y Y	Ethernet link is up with 1000 Mbit/s
Y □ Y □ Y □ Y □	Ethernet activity at 1000 Mbit/s

Note
ETH1 and ETH2: 10 Gbps Ethernet connections show an increased sensitivity to EMI. Data integrity is guaranteed by use of protocols which may reduce data throughput.

4.5.5.2 LEDs

Power

Symbol	Sequence	Description
	R R R R R R R R	Device is powered but Firmware not running. The Power LED will light up red at the beginning of the boot sequence of the Host Uplink Card. ► Please contact Vector VX support if the Power LED stays red for more than 10 seconds. In this case a critical error has occurred.
	G G G G G G G G	Firmware loaded successfully.
	G ☐ G ☐ G ☐ G ☐	Response to ”Identify VX device command” (VXconfig).

System

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Firmware not loaded.
	R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Host Uplink Card, its system components are initialized. If a component fails to initialize properly, the System LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
	G G G G G G G G	Device is in normal operating mode.

Error

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	No fault condition.
	R R R R R R R R	One of the plug-in cards is not reachable or incompatible.
	R R R R ☐ ☐ ☐ ☐	Ethernet switch permanent error.

Ethernet

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Ethernet link is down.
	G G G G G G G G	Ethernet link is up.
	G ☐ G ☐ G ☐ G ☐	Ethernet link activity.

Time Sync

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	No modules to be synchronized.
	R ☐ R ☐ R ☐ R ☐	Problems during synchronization.
	R R R R R R R R	Synchronization error. At least one module not synchronized.
	G G G G G G G G	All modules synchronized.
	G ☐ G ☐ G ☐ G ☐	Synchronization in progress.

CANape Microservices

Symbol	Sequence	Description
	☐☐☐☐☐☐☐☐	Microservices not ready for operation OR no Microservices license present.
	R R R R R R R R	Microservices defective.
	G G G G G G G G	Microservices ready.
	G G G G R R R R	Error at start of measurement: Devices not present or error responses
	R ☐ R ☐ R ☐ R ☐	Configuration error: Measurement configuration is inconsistent OR configuration valid but Microservices license missing.

Internal Storage

Symbol	Sequence	Description
	☐☐☐☐☐☐☐☐	No Microservices license present.
	G G G G G G G G	Internal Storage is operational.
	Y ☐ Y ☐ Y ☐ Y ☐	Update of the Microservices is in progress. Do not turn off VX1161 power!
	R R R R R R R R	Internal storage access error. May be resolvable by using VXconfig to execute a factory reset of the VX1161.22A.

4.5.5.3 Technical Data

Weight	407 g
Power consumption	typ. 18 W at room temp.
Temperature range	-40 °C ... +60 °C
Ethernet ports	All ports have a RJ45 socket 1x 10 Gbps Ethernet as Host Uplink 1x 10 Gbps Ethernet for Cascading or as Host Uplink 1x 1 Gbps Ethernet/100BASE-T1 for network devices
Part number	22465

4.5.6 VX1161.22B Host Uplink

The VX1161.22B Host Uplink Card for the VX1161 Multi Base Module is the central data distribution node of the VX1161 Multi Base Module.

All communication within the VX1161 Multi Base Module and to external network nodes is routed through this card. The VX1161.22B Host Uplink Card contains a PTP-enabled Ethernet switch.

On the front panel of the card, there are two 10 Gbps Ethernet ports (RJ45) which can be used as Host Uplink or for Cascading other high-performance network hardware, and four RJ45 ports supporting 1 Gbps Ethernet or 100BASE-T1 (auto-sensing). The VX1161.22B Host Uplink Card can also host Microservices.

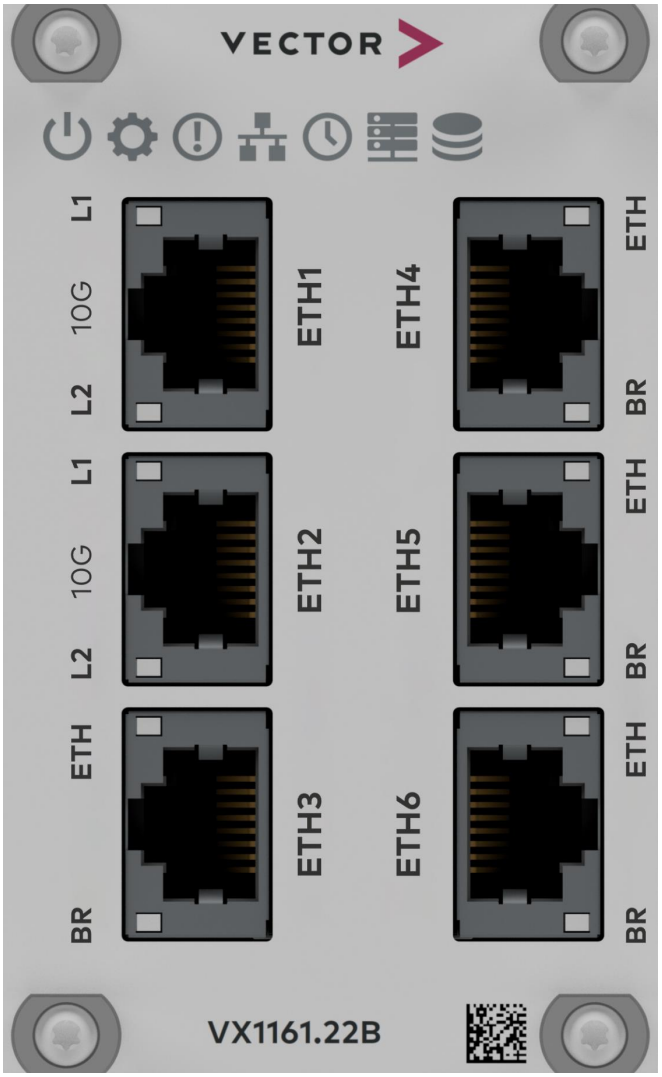


Figure 45: VX1161.22B Host Uplink Card

4.5.6.1 Connectors

- **ETH1 Host**
  10 Gbps Ethernet port with RJ45 connector. This port is intended as uplink to the Tool PC. The VX1161 Multi Base Module can be configured to enter standby if the link partner at this Ethernet port is missing.

  LEDs light up permanently if there is an Ethernet link or blink if there is Ethernet activity.

L2	L1	Description
-	Yellow	100M
-	Green	1G
Yellow	-	2.5G
Green	-	5G
Yellow	Yellow	10G

- **ETH2 Cascade**
  10Gbps Ethernet port with RJ45 connector. This port can either be used to cascade other VX1161 Multi Base Modules or other Ethernet hardware, or it can be used as second uplink to a Tool PC.

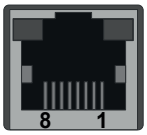
  LEDs light up permanently if there is an Ethernet link or blink if there is Ethernet activity.

L2	L1	Description
-	Yellow	100M
-	Green	1G
Yellow	-	2.5G
Green	-	5G
Yellow	Yellow	10G

- **ETH3 ... ETH6 Target**
  1 Gbps Ethernet/100BASE-T1 ports with auto media sensing. These ports can be used to attach XPODs, VX1000 Base Modules or other Ethernet hardware. The intended data flow is from the Interface Cards or from these Ethernet ports to the Tool PC attached to the Host port (ETH1), meaning that the Host Uplink Card acts as a link concentrator towards the Tool PC. As the Host Uplink Card contains a fully-featured Ethernet switch, other topologies are also possible.

  Standard Ethernet connector (RJ45) for 100BASE-TX, 1000BASE-T or 100BASE-T1.

Pin	Assignment for 100BASE-T1
1	MDI+
2	MDI-
3...8	Not connected



WARNING!
Please note that VX1000 Ethernet ports supporting both 1000BASE-T and 100BASE-T1 are only intended as interface to measurement and calibration hardware, like an XPOD. They are not designed to be connected to automotive Ethernet networks.

These ports are part of an Ethernet switch connecting the VX device itself, and any network node or network segment attached to ETH ports of the VX device. Such, the ETH ports may be exposed to unexpected Ethernet traffic, like broadcasts, which might not be tolerable in an automotive Ethernet network. Also, due to supporting both, standard Ethernet, and 100BASE-T1 on the same port, the electrical circuit slightly differs from what is common in automotive Ethernet networks, which also might not be tolerable in an automotive Ethernet network.

To communicate with automotive Ethernet networks, please use dedicated Vector Network Interface Hardware.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off Y = yellow G = green

BR/AE
LED indicating the 100BASE-T1 status.

Sequence	Description
G G G G G G G G	100BASE-T1 link is up with VX1161.22B as Master
G □ G □ G □ G □	100BASE-T1 activity
Y Y Y Y Y Y Y Y	100BASE-T1 link is up with VX1161.22B as Slave
Y □ Y □ Y □ Y □	100BASE-T1 activity

ETH
LED indicating the Ethernet status.

Sequence	Description
G G G G G G G G	Ethernet link is up with 100 Mbit/s
G □ G □ G □ G □	Ethernet activity at 100 Mbit/s
Y Y Y Y Y Y Y Y	Ethernet link is up with 1000 Mbit/s
Y □ Y □ Y □ Y □	Ethernet activity at 1000 Mbit/s

Note
ETH1 and ETH2: 10 Gbps Ethernet connections show an increased sensitivity to EMI. Data integrity is guaranteed by use of protocols which may reduce data throughput.

4.5.6.2 LEDs

Power

Symbol	Sequence	Description
	R R R R R R R R	Device is powered but Firmware not running. The Power LED will light up red at the beginning of the boot sequence of the Host Uplink Card. ► Please contact Vector VX support if the Power LED stays red for more than 10 seconds. In this case a critical error has occurred.
	G G G G G G G G	Firmware loaded successfully.
	G ☐ G ☐ G ☐ G ☐	Response to ”Identify VX device command” (VXconfig).

System

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Firmware not loaded.
	R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Host Uplink Card, its system components are initialized. If a component fails to initialize properly, the System LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
	G G G G G G G G	Device is in normal operating mode.

Error

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	No fault condition.
	Y Y Y Y Y Y Y Y	Device is in update-only operating mode. During the boot sequence of the Host Uplink Card, its system components are initialized. If a component fails to initialize properly, the System LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
	R R R R R R R R	One of the plug-in cards is not reachable or incompatible.
	R R R R ☐ ☐ ☐ ☐	The Ethernet switch has reported an error (port no longer working). Restart necessary.

Ethernet

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Ethernet link is down.
	G G G G G G G G	Ethernet link is up.

Ethernet

Symbol	Sequence	Description
	G ☐ G ☐ G ☐ G ☐	Ethernet link activity.

Time Sync

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	No modules to be synchronized.
	R ☐ R ☐ R ☐ R ☐	Problems during synchronization.
	R R R R R R R R	Synchronization error. At least one module not synchronized.
	G G G G G G G G	All modules synchronized.
	G ☐ G ☐ G ☐ G ☐	Synchronization in progress.

CANape Microservices

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Microservices not ready for operation OR no Microservices license present.
	R R R R R R R R	Microservices defective.
	G G G G G G G G	Microservices ready.
	G G G G R R R R	Error at start of measurement: Devices not present or error responses
	R ☐ R ☐ R ☐ R ☐	Configuration error: License missing if measurement configuration exists OR measurement configuration is inconsistent

Internal Storage

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	No Microservices license present.
	G G G G G G G G	Internal Storage is operational.
	Y ☐ Y ☐ Y ☐ Y ☐	Update of the Microservices is in progress. Do not switch off VX1161 power!
	R R R R R R R R	Internal storage access error. May be resolvable by using VXconfig to execute a factory reset of the VX1161.22B.

4.5.6.3 Technical Data

Weight	453 g
Power consumption	typ. 21 W at room temp.
Temperature range	-40 °C ... +60 °C
Ethernet ports	All ports have a RJ45 socket 1x 10 Gbps Ethernet as Host Uplink 1x 10 Gbps Ethernet for Cascading or as Host Uplink 4x 1 Gbps Ethernet/100BASE-T1for network devices
Part number	22464

4.5.7 VX1161.22C Host Uplink

The VX1161.22C Host Uplink Card for the VX1161 Multi Base Module is the central data distribution node of the VX1161 Multi Base Module.

All communication within the VX1161 Multi Base Module and to external network nodes is routed through this card. The VX1161.22C Host Uplink Card contains a PTP-enabled Ethernet switch.

On the front panel of the card, there are four 10 Gbps Ethernet ports (RJ45), and two RJ45 ports supporting 1 Gbps Ethernet or 100BASE-T1 (auto-sensing). The VX1161.22C Host Uplink Card can also host Microservices.

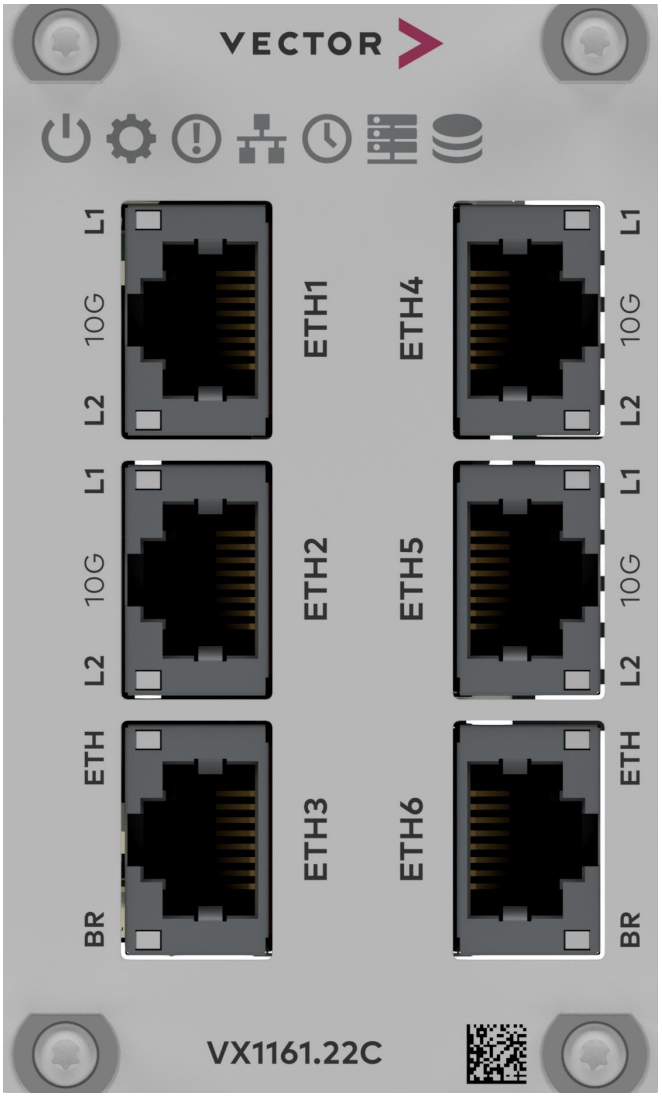


Figure 46: VX1161.22C Host Uplink Card

4.5.7.1 Connectors

- **ETH1**
  10 Gbps Ethernet port with RJ45 connector. This port is intended as uplink to a measurement application. The VX1161 Multi Base Module can be configured to enter standby if the link partner at this Ethernet port is missing.

LEDs light up permanently if there is an Ethernet link or blink if there is Ethernet activity.

L2	L1	Description
-	Yellow	100M
-	Green	1G
Yellow	-	2.5G
Green	-	5G
Yellow	Yellow	10G

- **ETH2/4/5**
  10Gbps Ethernet port with RJ45 connector. This port can either be used to cascade other VX1161 Multi Base Modules or other Ethernet hardware, or it can be used as second uplink to a measurement application.

  LEDs light up permanently if there is an Ethernet link or blink if there is Ethernet activity.

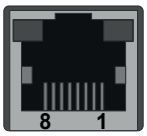
L2	L1	Description
-	Yellow	100M
-	Green	1G
Yellow	-	2.5G
Green	-	5G
Yellow	Yellow	10G

- **ETH3/6**
  1 Gbps Ethernet/100BASE-T1 ports with auto media sensing. These ports can be used to attach XPODs, VX1000 Base Modules or other Ethernet hardware.

  The intended data flow is from the Interface Cards or from these Ethernet ports to the Tool PC attached to the Host port (ETH1), meaning that the Host Uplink Card acts as a link concentrator towards the Tool PC. As the Host Uplink Card contains a fully-featured Ethernet switch, other topologies are also possible.

  Standard Ethernet connector (RJ45) for 100BASE-TX, 1000BASE-T or 100BASE-T1.

Pin	Assignment for 100BASE-T1
1	MDI+
2	MDI-
3...8	Not connected



WARNING!
Please note that VX1000 Ethernet ports supporting both 1000BASE-T and 100BASE-T1 are only intended as interface to measurement and calibration hardware, like an XPOD. They are not designed to be connected to automotive Ethernet networks.

These ports are part of an Ethernet switch connecting the VX device itself, and any network node or network segment attached to ETH ports of the VX device. Such, the ETH ports may be exposed to unexpected Ethernet traffic, like broadcasts, which might not be tolerable in an automotive Ethernet network. Also, due to supporting both, standard Ethernet, and 100BASE-T1 on the same port, the electrical circuit slightly differs from what is common in automotive Ethernet networks, which also might not be tolerable in an automotive Ethernet network.

To communicate with automotive Ethernet networks, please use dedicated Vector Network Interface Hardware.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off Y = yellow G = green

BR/AE
LED indicating the 100BASE-T1 status.

Sequence	Description
G G G G G G G G	100BASE-T1 link is up with VX1161.22C as Master
G □ G □ G □ G □	100BASE-T1 activity
Y Y Y Y Y Y Y Y	100BASE-T1 link is up with VX1161.22C as Slave
Y □ Y □ Y □ Y □	100BASE-T1 activity

ETH
LED indicating the Ethernet status.

Sequence	Description
G G G G G G G G	Ethernet link is up with 100 Mbit/s
G □ G □ G □ G □	Ethernet activity at 100 Mbit/s
Y Y Y Y Y Y Y Y	Ethernet link is up with 1000 Mbit/s
Y □ Y □ Y □ Y □	Ethernet activity at 1000 Mbit/s

Note
ETH1/2/4/5: 10 Gbps Ethernet connections show an increased sensitivity to EMI. Data integrity is guaranteed by use of protocols which may reduce data throughput.

4.5.7.2 LEDs

Power

Symbol	Sequence	Description
	R R R R R R R R	Device is powered but Firmware not running. The Power LED will light up red at the beginning of the boot sequence of the Host Uplink Card. ► Please contact Vector VX support if the Power LED stays red for more than 10 seconds. In this case a critical error has occurred.
	G G G G G G G G	Firmware loaded successfully.
	G ☐ G ☐ G ☐ G ☐	Response to ”Identify VX device command” (VXconfig).

System

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Firmware not loaded.
	R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Host Uplink Card, its system components are initialized. If a component fails to initialize properly, the System LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
	G G G G G G G G	Device is in normal operating mode.

Error

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	No fault condition.
	Y Y Y Y Y Y Y Y	Device is in update-only operating mode. During the boot sequence of the Host Uplink Card, its system components are initialized. If a component fails to initialize properly, the System LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
	R R R R R R R R	One of the plug-in cards is not reachable or incompatible.
	R R R R ☐ ☐ ☐ ☐	The Ethernet switch has reported an error (port no longer working). Restart necessary.

Ethernet

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Ethernet link is down.
	G G G G G G G G	Ethernet link is up.

Ethernet

Symbol	Sequence	Description
	G ☐ G ☐ G ☐ G ☐	Ethernet link activity.

Time Sync

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	No modules to be synchronized.
	R ☐ R ☐ R ☐ R ☐	Problems during synchronization.
	R R R R R R R R	Synchronization error. At least one module not synchronized.
	G G G G G G G G	All modules synchronized.
	G ☐ G ☐ G ☐ G ☐	Synchronization in progress.

CANape Microservices

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Microservices not ready for operation OR no Microservices license present.
	R R R R R R R R	Microservices defective.
	G G G G G G G G	Microservices ready.
	G G G G R R R R	Error at start of measurement: Devices not present or error responses
	R ☐ R ☐ R ☐ R ☐	Configuration error: License missing if measurement configuration exists OR measurement configuration is inconsistent

Internal Storage

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	No Microservices license present.
	G G G G G G G G	Internal Storage is operational.
	Y ☐ Y ☐ Y ☐ Y ☐	Update of the Microservices is in progress. Do not switch off VX1161 power!
	R R R R R R R R	Internal storage access error. May be resolvable by using VXconfig to execute a factory reset of the VX1161.22C.

4.5.7.3 Technical Data

Weight	490 g
Power consumption	approx. 23 W @ 20 °C
Temperature range	-40 °C ... +60 °C
Ethernet ports	All ports have a RJ45 socket 1x 10 Gbps Host Uplink 3x 10 Gbps Ethernet for Cascading or as Host Uplink 2x 1 Gbps Ethernet/100BASE-T1 for network devices
Part number	22645

4.5.8 VX1161.31A Serial

The VX1161.31A Serial Card allows a fast ECU access for measurement and calibration purposes. Communication with the tool is provided through XCP on Ethernet as standardized protocol. There is a single Serial connector on the front panel of the card which can be used to attach a VX134x Serial Cable or VX1311 Serial2 Cable to connect a VX154x Serial POD.

Figure 47: VX1161.31A Serial Card

4.5.8.1 Connectors

- **POD**
  This connector is used to attach a VX154x Serial POD through a VX134x Serial Cable or VX1311 Serial2 Cable.

4.5.8.2 LEDs

Power

Symbol	Sequence	Description
	R R R R R R R R	Device is powered but Firmware not running. The Power LED will light up red at the beginning of the boot sequence of the Interface Card. ► Please contact Vector VX support if the Power LED stays red for more than 10 seconds. In this case a critical error has occurred.
	G G G G G G G G	Firmware loaded successfully.
	G ☐ G ☐ G ☐ G ☐	Response to ”Identify VX device command” (VXconfig).

System

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Firmware not loaded.
	R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Interface Card, its system components are initialized. If a component fails to initialize properly, the System LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
	G G G G G G G G	Device is in normal operating mode.

POD

Symbol	Sequence	Description
P	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Device is not in normal operating mode.
	R R R R R R R R	POD not detected. After the Interface Card boot sequence has been completed successfully, the system searches for an attached POD. As long as no POD is found, the POD LED will stay red. ► Please check if the POD is connected to the Interface Card via an undamaged VX134x Serial cable or VX1311 Serial2 cable.

POD

Symbol	Sequence	Description
		► Please also check if the connector of the cable is locked-in correctly into the corresponding socket of the Interface Card (and POD).
	G G G G G G G G	POD in normal operating mode
	G G G G R R R R	POD detected but configuration incorrect The POD was detected by the Interface Card, but the target interface configuration is incorrect. ► Please open VXconfig and adjust the POD settings according to the target ECU configuration.

ECU

Symbol	Sequence	Description
E	□ □ □ □ □ □ □ □	POD not in normal operating mode.
	R R R R R R R R	Target microcontroller not detected. The POD was detected by the VX1000 Interface Card and is configured correctly, but the target ECU cannot be accessed. ► Please check if the POD is connected (via an appropriate cable) to the target microcontroller. ► Please check if the target system is powered correctly and operational. ► Please check if the target interface settings of the POD match the ECU configuration. For some targets, the target interface is disabled if the ECU parameter structure is not found. ► Please check if the ECU parameter structure (gVX1000) is present in the target RAM and that its address is correctly configured in VXconfig.
	R R R R □ □ □ □	Inconsistent Measurement Data detected. The target microcontroller is accessible and was configured correctly to be accessed by the VX1000 system. The VX1000 detected a situation where the consistency of the measurement data cannot be guaranteed anymore. Therefore the measurement process was stopped. All data processed up to that event are valid and consistent. This problem can either be caused by: ► Overload Situations - Possible Solution: Increase target interface clock frequency of the POD, if possible. ► Technical issues of either ECU or VX1000

ECU

Symbol	Sequence	Description
		- See the VXconsole log or the messages sent via XCP to the XCP-Master (e.g. CANape) for details.
	G G G G ☐ ☐ ☐ ☐	RAM synchronization in process
	G G G G G G G G	Target microcontroller detected and configuration correct
	G G G G G G G R	Target microcontroller detected but configuration incorrect The target microcontroller is accessible but was not configured correctly to be accessed by the VX1000 system. This means the settings made in VXconfig do not match the properties of the ECU Code Instrumentation. ► Please check if the ECU Parameter Structure address set in VXconfig matches the address in the Linker MAP file of the target ECU code. ► Please check if the byte order set in VXconfig matches the byte order of the target ECU.

Ethernet

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Backplane link not established.
	R R R R R R R R	Backplane link error.
	G G G G G G G G	Backplane link established.
	G ☐ G ☐ G ☐ G ☐	Backplane link activity.

4.5.8.3 Technical Data

Weight	193 g
Power consumption	typ. 4.4 W at room temp, without POD
Temperature range	-40 °C ... +60 °C
Data rate	up to 20 Mbyte/s XCP DAQ data
Part number	22466

4.5.9 VX1161.32B HSSL

The VX1161.32B HSSL Card allows a high-performance ECU access for measurement and calibration purposes. Communication with the tool is provided through XCP on Ethernet as standardized protocol. There is a single HSSL connector on the front panel of the card which can be used to attach a VX137x HSSL Cable to connect a VX14xx HSSL POD.

Figure 48: VX1161.32B HSSL Card

4.5.9.1 Connectors

- **POD**
  This connector is used to attach a VX14xx HSSL POD through a VX137x HSSL Cable.

4.5.9.2 LEDs

Power

Symbol	Sequence	Description
	R R R R R R R R	Device is powered but Firmware not running. The Power LED will light up red at the beginning of the boot sequence of the Interface Card. ► Please contact Vector VX support if the Power LED stays red for more than 10 seconds. In this case a critical error has occurred.
	G G G G G G G G	Firmware loaded successfully.
	G ☐ G ☐ G ☐ G ☐	Response to ”Identify VX device command” (VXconfig).

System

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Firmware not loaded.
	R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Interface Card, its system components are initialized. If a component fails to initialize properly, the System LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
	G G G G G G G G	Device is in normal operating mode.

POD

Symbol	Sequence	Description
P	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Device is not in normal operating mode.
	R R R R R R R R	POD not detected. After the Interface Card boot sequence has been completed successfully, the system searches for an attached POD. As long as no POD is found, the POD LED will stay red. ► Please check if the POD is connected to the Interface Card via an undamaged VX137x HSSL cable.

POD

Symbol	Sequence	Description
		► Please also check if the connector of the cable is locked-in correctly into the corresponding socket of the Interface Card (and POD).
	G G G G G G G G	POD in normal operating mode
	G G G G R R R R	POD detected but configuration incorrect The POD was detected by the Interface Card, but the target interface configuration is incorrect. ► Please open VXconfig and adjust the POD settings according to the target ECU configuration.

ECU

Symbol	Sequence	Description
E	□ □ □ □ □ □ □ □	POD not in normal operating mode.
	R R R R R R R R	Target microcontroller not detected. The POD was detected by the VX1000 Interface Card and is configured correctly, but the target ECU cannot be accessed. ► Please check if the POD is connected (via an appropriate cable) to the target microcontroller. ► Please check if the target system is powered correctly and operational. ► Please check if the target interface settings of the POD match the ECU configuration. For some targets, the target interface is disabled if the ECU parameter structure is not found. ► Please check if the ECU parameter structure (gVX1000) is present in the target RAM and that its address is correctly configured in VXconfig.
	R R R R □ □ □ □	Inconsistent Measurement Data detected. The target microcontroller is accessible and was configured correctly to be accessed by the VX1000 system. The VX1000 detected a situation where the consistency of the measurement data cannot be guaranteed anymore. Therefore the measurement process was stopped. All data processed up to that event are valid and consistent. This problem can either be caused by: ► Overload Situations - Possible Solution: Increase target interface clock frequency of the POD, if possible. ► Technical issues of either ECU or VX1000

ECU

Symbol	Sequence	Description
		- See the VXconsole log or the messages sent via XCP to the XCP-Master (e.g. CANape) for details.
	G G G G ☐ ☐ ☐ ☐	RAM synchronization in process
	G G G G G G G G	Target microcontroller detected and configuration correct
	G G G G G G G R	Target microcontroller detected but configuration incorrect The target microcontroller is accessible but was not configured correctly to be accessed by the VX1000 system. This means the settings made in VXconfig do not match the properties of the ECU Code Instrumentation. ► Please check if the ECU Parameter Structure address set in VXconfig matches the address in the Linker MAP file of the target ECU code. ► Please check if the byte order set in VXconfig matches the byte order of the target ECU.

Ethernet

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Backplane link not established.
	R R R R R R R R	Backplane link error.
	G G G G G G G G	Backplane link established.
	G ☐ G ☐ G ☐ G ☐	Backplane link activity.

4.5.9.3 Technical Data

Weight	227 g
Power consumption	typ. 4.9 W at room temp, without POD
Temperature range	-40 °C ... +60 °C
Data rate	up to 100 MB/s overall for XCP DAQ data and radar raw data
Part number	22467

4.5.10 VX1161.32C HSSL2

The VX1161.32C HSSL2 Card allows a high-performance ECU access for measurement and calibration purposes. Communication with the tool is provided through XCP on Ethernet as standardized protocol. There is a single HSSL2 connector on the front panel of the card which can be used to attach a VX1381 HSSL2 Cable to connect a VX14xx HSSL POD.

Figure 49: VX1161.32C HSSL2 Card

4.5.10.1 Connectors

► **POD**
This connector is used to attach a VX14xx HSSL POD through a VX1381 HSSL2 Cable.

4.5.10.2 LEDs

Power

Symbol	Sequence	Description
	R R R R R R R R	Device is powered but Firmware not running. The Power LED will light up red at the beginning of the boot sequence of the Interface Card. ► Please contact Vector VX support if the Power LED stays red for more than 10 seconds. In this case a critical error has occurred.
	G G G G G G G G	Firmware loaded successfully.
	G ☐ G ☐ G ☐ G ☐	Response to ”Identify VX device command” (VXconfig).

System

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Firmware not loaded.
	R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Interface Card, its system components are initialized. If a component fails to initialize properly, the System LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
	G G G G G G G G	Device is in normal operating mode.

POD

Symbol	Sequence	Description
P	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Device is not in normal operating mode.
	R R R R R R R R	POD not detected. After the Interface Card boot sequence has been completed successfully, the system searches for an attached POD. As long as no POD is found, the POD LED will stay red. ► Please check if the POD is connected to the Interface Card via an undamaged VX1381 HSSL2 cable.

POD

Symbol	Sequence	Description
		► Please also check if the connector of the cable is locked-in correctly into the corresponding socket of the Interface Card (and POD).
	G G G G G G G G	POD in normal operating mode
	G G G G R R R R	POD detected but configuration incorrect The POD was detected by the Interface Card, but the target interface configuration is incorrect. ► Please open VXconfig and adjust the POD settings according to the target ECU configuration.

ECU

Symbol	Sequence	Description
E	□ □ □ □ □ □ □ □	POD not in normal operating mode.
	R R R R R R R R	Target microcontroller not detected. The POD was detected by the VX1000 Interface Card and is configured correctly, but the target ECU cannot be accessed. ► Please check if the POD is connected (via an appropriate cable) to the target microcontroller. ► Please check if the target system is powered correctly and operational. ► Please check if the target interface settings of the POD match the ECU configuration. For some targets, the target interface is disabled if the ECU parameter structure is not found. ► Please check if the ECU parameter structure (gVX1000) is present in the target RAM and that its address is correctly configured in VXconfig.
	R R R R □ □ □ □	Inconsistent Measurement Data detected. The target microcontroller is accessible and was configured correctly to be accessed by the VX1000 system. The VX1000 detected a situation where the consistency of the measurement data cannot be guaranteed anymore. Therefore the measurement process was stopped. All data processed up to that event are valid and consistent. This problem can either be caused by: ► Overload Situations - Possible Solution: Increase target interface clock frequency of the POD, if possible. ► Technical issues of either ECU or VX1000

ECU

Symbol	Sequence	Description
		- See the VXconsole log or the messages sent via XCP to the XCP-Master (e.g. CANape) for details.
	G G G G □ □ □ □	RAM synchronization in process
	G G G G G G G G	Target microcontroller detected and configuration correct
	G G G G G G G R	Target microcontroller detected but configuration incorrect The target microcontroller is accessible but was not configured correctly to be accessed by the VX1000 system. This means the settings made in VXconfig do not match the properties of the ECU Code Instrumentation. ► Please check if the ECU Parameter Structure address set in VXconfig matches the address in the Linker MAP file of the target ECU code. ► Please check if the byte order set in VXconfig matches the byte order of the target ECU.

Ethernet

Symbol	Sequence	Description
	□ □ □ □ □ □ □ □	Backplane link not established.
	R R R R R R R R	Backplane link error.
	G G G G G G G G	Backplane link established.
	G □ G □ G □ G □	Backplane link activity.

4.5.10.3 Technical Data

Weight	227 g
Power consumption	typ. 5 W at room temp, without POD
Temperature range	-40 °C ... +60 °C
Data rate	up to 100 MB/s overall for XCP DAQ data and radar raw data
Part number	22468

4.5.11 VX1161.41A 6xCAN

The VX1161.41A 6xCAN Card provides 6 CAN channels with fixed transceivers. The CAN channels can be accessed via the Vector XL API on the tool PC.

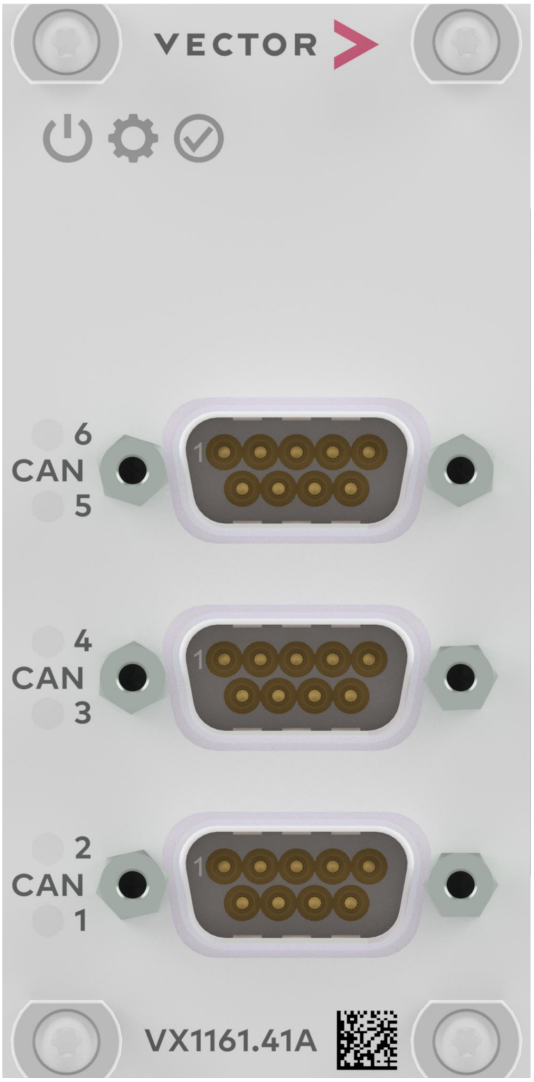
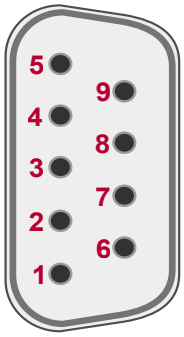


Figure 50: VX1161.41A 6xCAN Card

4.5.11.1 Connectors

- **CAN1/2, CAN3/4 and CAN5/6**
  The VX1161.41A offers six CAN channels which are accessible through D-SUB9 connectors on the front panel of the card. The CAN channels use high-speed TJA1057G transceivers (capacitively decoupled).

Pin	Assignment	Direction
1	CAN Low (CH B)	Input/output of VX1161.4x
2	CAN Low (CH A)	Input/output of VX1161.4x
3	GND	-
4	reserved	-
5	Shield	-
6	GND	-
7	CAN High (CH A)	Input/output of VX1161.4x
8	CAN High (CH B)	Input/output of VX1161.4x
9	Not connected	-



Note
Please note the following hardware channel mapping:

VX1161.41A	Vector Hardware Manager / XL API
CAN1	Channel 1
CAN2	Channel 2
CAN3	Channel 3
CAN4	Channel 4
CAN5	Channel 5
CAN6	Channel 6

4.5.11.2 LEDs

Power

Symbol	Sequence	Description
	R R R R R R R R	Device is powered but Firmware not running. The Power LED will light up red at the beginning of the boot sequence of the Interface Card. ► Please contact Vector VX support if the Power LED stays red for more than 10 seconds. In this case a critical error has occurred.
	G G G G G G G G	Firmware loaded successfully.
	G ☐ G ☐ G ☐ G ☐	Response to ”Identify VX device command” (VXconfig).

System

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Firmware not loaded.
	R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Interface Card, its system components are initialized. If a component fails to initialize properly, the System LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
	G G G G G G G G	Device is in normal operating mode.
	G ☐ G ☐ G ☐ G ☐	Firmware update in progress.

State

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	No connection to PC/driver.
	G G G G G G G G	Measurement running.
	G ☐ G ☐ G ☐ G ☐	Connected to PC, no measurement running.

CAN

Symbol	Sequence	Description
2 CAN 1	G ☐ G ☐ G ☐ G ☐	Data frames have been sent or received correctly.
	O ☐ O ☐ O ☐ O ☐	Error frames have been sent or received.
	R R R R R R R R	Bus off.

4.5.11.3 Technical Data

Weight	215 g
Power consumption	typ. 4.7 W at room temp, without CAN counterpart station
Temperature range	-40 °C ... +60 °C
Part number	22469

4.5.12 VX1161.41B 6xCAN 1xFR

The VX1161.41B 6xCAN 1xFR Card provides 6 CAN channels and one FlexRay A/B with fixed transceivers. All channels can be accessed via the Vector XL API on the tool PC.

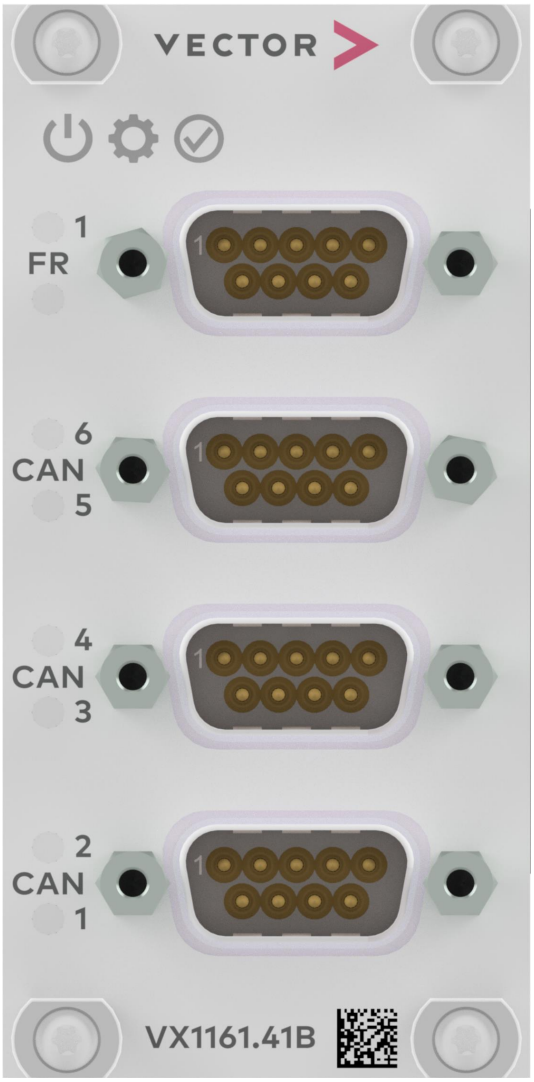
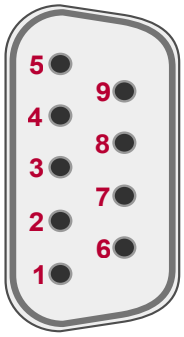


Figure 51: VX1161.41B 6xCAN 1xFR Card

4.5.12.1 Connectors

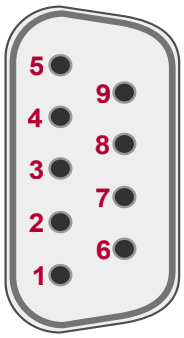
- **FLEXRAY**
  The VX1161.41B offers one FlexRay channel A/B pair, which is accessible through a D-SUB9 connector on the front panel of the card. The FlexRay channels use TJA1082 transceivers (capacitively decoupled).

Pin	Assignment	Direction
1	Not connected	-
2	FlexRay BM A	Input/output of VX1161.4x
3	FlexRay GND	-
4	FlexRay BM B	Input/output of VX1161.4x
5	Shield	-
6	Not connected	-
7	FlexRay BP A	Input/output of VX1161.4x
8	FlexRay BP B	Input/output of VX1161.4x
9	Not connected	-



- **CAN1/2, CAN3/4 and CAN5/6**
  The VX1161.41B offers six ISO CAN FD channels which are accessible through D-SUB9 connectors on the front panel of the card. The CAN channels use high-speed TJA1057G transceivers (capacitively decoupled).

Pin	Assignment	Direction
1	CAN Low (CH B)	Input/output of VX1161.4x
2	CAN Low (CH A)	Input/output of VX1161.4x
3	GND	-
4	reserved	-
5	Shield	-
6	GND	-
7	CAN High (CH A)	Input/output of VX1161.4x
8	CAN High (CH B)	Input/output of VX1161.4x
9	Not connected	-



Note
Please note the following hardware channel mapping:

VX1161.41B	Vector Hardware Manager / XL API
FLEXRAY	Channel 1
CAN1	Channel 2
CAN2	Channel 3
CAN3	Channel 4
CAN4	Channel 5
CAN5	Channel 6
CAN6	Channel 7

4.5.12.2 LEDs

Power

Symbol	Sequence	Description
	R R R R R R R R	Device is powered but Firmware not running. The Power LED will light up red at the beginning of the boot sequence of the Interface Card. ► Please contact Vector VX support if the Power LED stays red for more than 10 seconds. In this case a critical error has occurred.
	G G G G G G G G	Firmware loaded successfully.
	G ☐ G ☐ G ☐ G ☐	Response to ”Identify VX device command” (VXconfig).

System

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Firmware not loaded.
	R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Interface Card, its system components are initialized. If a component fails to initialize properly, the System LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
	G G G G G G G G	Device is in normal operating mode.
	G ☐ G ☐ G ☐ G ☐	Firmware update in progress.

State

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	No connection to PC/driver.
	G G G G G G G G	Measurement running.
	G ☐ G ☐ G ☐ G ☐	Connected to PC, no measurement running.

FlexRay

Symbol	Sequence	Description
1 FR	☐☐☐☐☐☐☐☐	FlexRay Communication Controller offline.
	G G G G G G G G	FlexRay Communication Controller synchronized.
	O O O O O O O O	FlexRay Communication Controller not synchronized.
	O ☐ O ☐ O ☐ O ☐	FlexRay Error Frames and normal Frames are received.
	R R R R R R R R	FlexRay Communication Controller halted.
	R ☐ R ☐ R ☐ R ☐	FlexRay Error frames detected.

CAN

Symbol	Sequence	Description
2 CAN 1	G ☐ G ☐ G ☐ G ☐	Data frames have been sent or received correctly.
	O ☐ O ☐ O ☐ O ☐	Error frames have been sent or received.
	R R R R R R R R	Bus off.

4.5.12.3 Technical Data

Weight	223 g
Power consumption	typ. 5.3 W at room temp, without FR/CAN counterpart station
Temperature range	-40 °C ... +60 °C
Part number	22470

4.5.13 VN1150 CAN/LIN/UART Interface

The VN1150 card provides four slots for piggybacks for analyzing CAN (FD), LIN, and UART buses. With an additional extension board, four more slots are available.

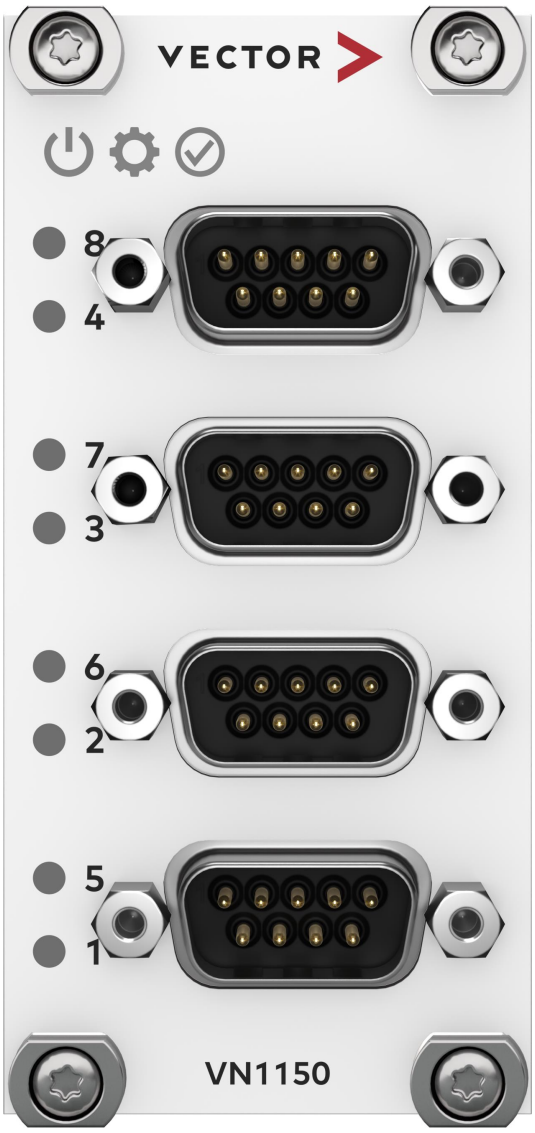


Figure 52: VN1150 Card

Reference
For more details about the VN1150, please visit our website.

4.5.14 VN5130 Ethernet Interface

4.5.14.1 General Information

The VN5130 Ethernet Interface Card provides six 1000BASE-T1/100BASE-T1 Automotive Ethernet channels with fixed Marvell 88Q2220M PHYs that are compatible with the 88Q2221M with respect to the user network. The channels are pairwise combined as TAPs, allowing the card to monitor three Ethernet links. Configurable TAP bypassing relays allow communication between the tapped link partners while the card is not powered.

The VN5130 is designed as Capture Module, especially for Ethernet logging via CANape. All data received via the TAPs is transferred to CANape or other data sink via CMP protocol.

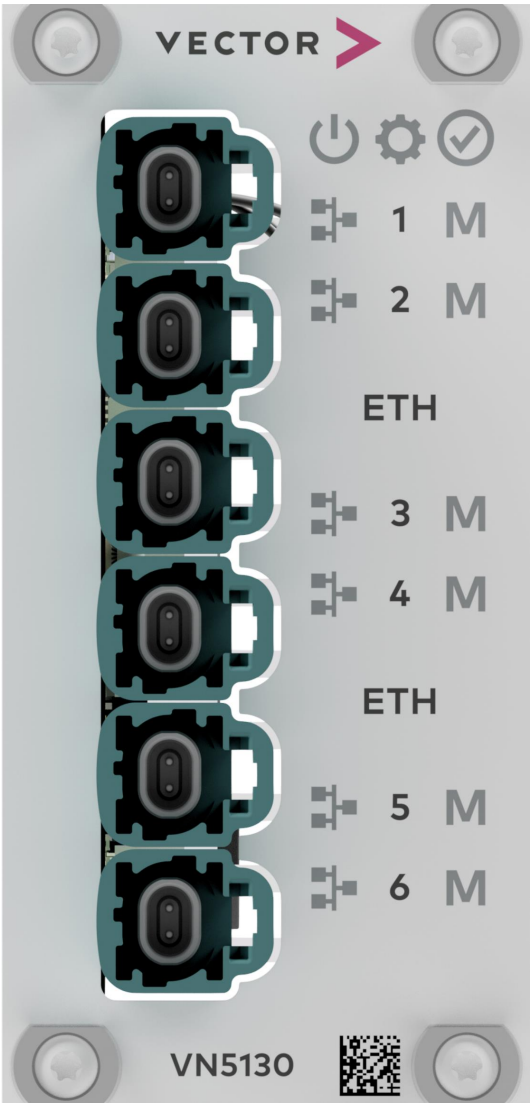


Figure 53: VN5130 Card

4.5.14.2 Connectors

- **ETH1 ... ETH6**
  The VN5130 provides six automotive Ethernet channels which are accessible through Rosenberger H-MTD Z plug connectors on the front panel of the card. The connectors are a robust version allowing more than 500 mating cycles.

Pin	Assignment
1	Ethernet MDI+
2	Ethernet MDI-

4.5.14.3 LEDs

- **Ethernet Port 1...6**
  This LED is permanent on if Ethernet link of the corresponding port is up or blinks if there is Ethernet activity on this port. The LED color indicates the link speed.

Color	Description
Green	1000 Mbit/s
Orange	100 Mbit/s

- **M Ethernet Port 1...6**
  This LED shows if the PHY of the corresponding port is configured as master or slave. Its color indicates the link speed (as defined in the table above).

State	Description
On	PHY is configured as master.
Off	PHY is configured as slave.

4.5.14.4 Technical Data

Weight	410 g
Power consumption	typ. 7.0 W at room temp (all ETH channels active)
Temperature range	-40 °C ... +60 °C
Part number	22560

4.5.15 VX1161.51A/.52A 2xTAP FPDLINK3 TI954/TI953

The VX1161.51A/.52A 2xTAP TI954/TI953 card is a two-channel TAP card for FPDLINK3 data streams. TAP means that a receiver (deserializer) and a transmitter (serializer) are connected back-to-back and the received data is forwarded to the transmitter and duplicated towards the logging system. Both channels contain one Texas Instruments DS90UB954 deserializer and one Texas Instruments DS90UB953 serializer each.

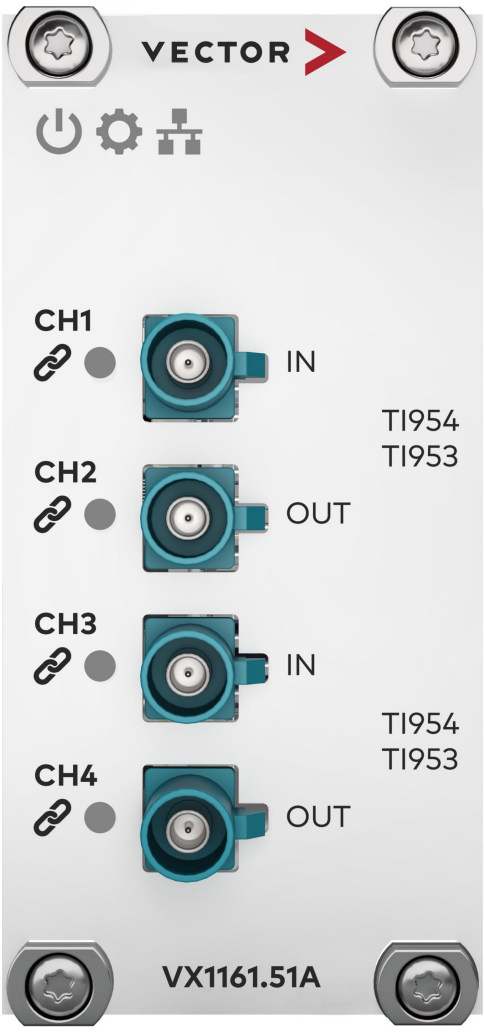


Figure 54: VX1161.51A/.52A 2xTAP TI954/TI953

The operating mode of the card can be configured either in TAP, RX or TX mode.

The VX1161.52A supports binary lossless compression of video streams on hardware. This compression method reduces the required bandwidth for video transmission, logging and storage typically by 40 to 60 %, depending on video content. Every channel of the VX1161.52 can be compressed. The VX1161.52 compressor pipelines support RAW8/10/12/14/16 and YUV422 8bit pixel formats only. For other pixel formats, please contact Vector.

The card can be configured with VXconfig and updated to new software versions with VXupdate. Initialization of the transceivers and remote sensors can be done via CASL language scripts within VXconfig.

Note
VX1161.51A is supported by VXtools v4.3 and later.
VX1161.52A is supported by VXtools v4.6 and later.

4.5.15.1 Connectors

- **CH1...4**
  The VX1161.51A/.52A provides FAKRA coaxial connectors of Type Z for connecting data sources and data sinks such as video or radar sensors and ECUs. Type Z connectors are compatible to all other coding types of single signal FAKRA connectors. The connector properties can be found in section Technical Data on page 243.

4.5.15.2 LEDs

The status of the card is displayed by system LEDs in the lid top. The status of every channel is displayed by a channel LED next to every channel connector.

Power

Symbol	Sequence	Description
	R R R R R R R R	Device is powered but Firmware not running. The Power LED will light up red at the beginning of the boot sequence of the Interface Card. ► Please contact Vector VX support if the Power LED stays red for more than 10 seconds. In this case a critical error has occurred.
	G G G G G G G G	Firmware loaded successfully.
	G ☐ G ☐ G ☐ G ☐	Response to ”Identify VX device command” (VXconfig).

System

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Firmware not loaded.
	R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Interface Card, its system components are initialized. If a component fails to initialize properly, the System LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
	G G G G G G G G	Device is in normal operating mode.
	G ☐ G ☐ G ☐ G ☐	Firmware update in progress.

Ethernet

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Backplane link not established.
	R R R R R R R R	Backplane link error.
	G G G G G G G G	Backplane link established.

Ethernet

Symbol	Sequence	Description
	G ☐ G ☐ G ☐ G ☐	Backplane link activity.

Channel

Symbol	Sequence	Description
	☐☐☐☐☐☐☐☐	Channel disabled.
	R R R R ☐☐☐☐	Channel enabled, invalid configuration. See VXconsole for details.
	R R R R R R R R	Channel enabled, no link established.
	G G G G G G G G	Channel enabled, link established, no data traffic.
	G ☐ G ☐ G ☐ G ☐	Channel enabled, link established, data traffic.

4.5.15.3 Boot and Link-up-Time

When the VX1161 is connected to power, the streaming card must prepare itself for a fast wake-up. For about t_{init} the power consumption will be P_{init}. After that time, the card enters standby mode and the power consumption is reduced to its minimum $P_{standby}$.

When the VX1161 is woken up (see section Connectors on page 176), the card is already prepared and can establish both ECU and sensor link within $t_{wakeLink}$. After wakeup, the card can provide data to the recorder after $t_{wakeReady}$.

The power consumption during normal operation consist of the maximum power of P_{run} drawn by the card as well as the power consumed by supplied cameras P_{POC}.

4.5.15.4 Changing the Mode of Operation

The VX1161.51A/.52A can be used in one of the following use cases:

- Video RX/TAP (video reception / logging)
- Video TX (video transmission / replay)

In these modes, sensors or ECUs with FPDLINK3 transceivers can be directly attached to log or replay streams of data.

The mode of operation can be changed in VXconfig on the Streaming Interface Card dialog.

4.5.15.5 Channel in TAP Mode

The TAP is designed to be as transparent as possible to the ECU.

If a channel group is configured forTAP mode , the incoming FPDLINK3 data stream is received by the DS90UB954 TAP deserializer. The data payload (e. g. video or radar frames) is then duplicated. One copy is forwarded to the DS90UB953 TAP serializer and one copy is forwarded to the recording data processor.

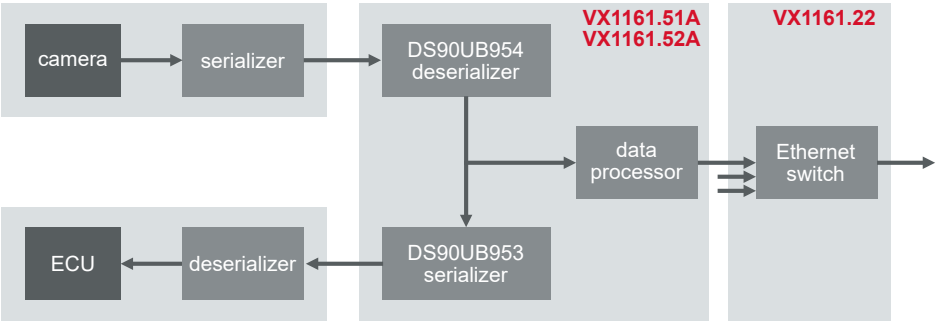


Figure 55: Channel in TAP mode; payload duplication

Control accesses from the ECU to the sensor via the I2C control channel or GPIOs are forwarded from the TAP serializer to the TAP deserializer (see section GPIO Interconnect on page 227).

TAP Initialization

The initialization sequence of the sensor and the ECU links follows this chart:

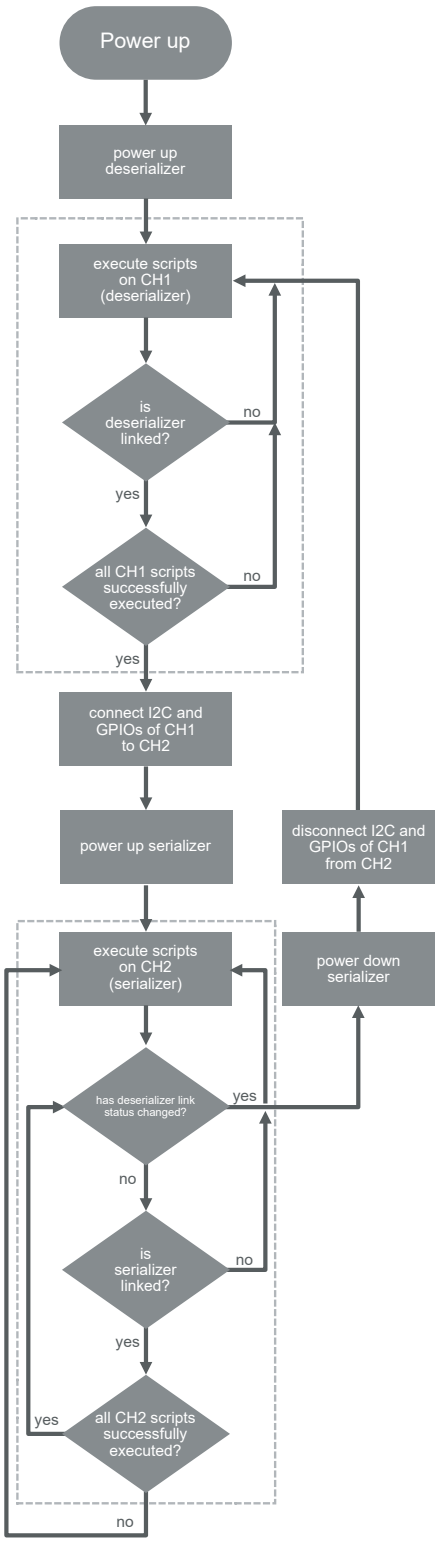


Figure 56: TAP initialization sequence

After power-up, the POC voltage is applied as configured and the deserializer is activated.

Next, the enabled scripts for the TAP RX channel (CH1/CH3) are immediately executed in an initial run (script 1, 2, 3) to configure the TAP deserializer, the sensor serializer and the sensor imager as early as possible.

After the initial script run on the TAP RX channel, the script execution engine waits for a TAP deserializer lock.

When the TAP deserializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the following conditions apply:

1. Script is not configured as "run once" (CASL SetConfig(7,1))
2. OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
3. OR script terminated with a return value ≠ 0 in previous run

If the TAP deserializer link unlocks, the script sequence repeats from the initial TAP deserializer run on, the I2C and GPIO bridges will be opened and the TAP serializer will be deactivated.

When the TAP deserializer locks, the I2C bus and GPIOs of the TAP serializer and TAP deserializer are connected. This allows the ECU to configure and control the sensor (see section GPIO Interconnect).

When the TAP deserializer locks, the TAP serializer is activated and the script execution on the TAP TX channel (CH2/CH4) are executed in an initial run (script 1, 2, 3).

After the initial script run on the TAP TX channel, the script execution engine waits for a TAP serializer lock.

When the TAP serializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the following conditions apply:

1. Script is not configured as "run once" (CASL SetConfig(7,1))
2. OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
3. OR script terminated with a return value ≠ 0 in previous run

If the TAP serializer link unlocks, the sequence repeats from the initial TAP serializer run on.

If the TAP deserializer link unlocks now, the I2C and GPIO bridges will be opened, the TAP serializer will be deactivated and the script sequence repeats from the initial TAP deserializer run on.

I2C Architecture

Without the TAP, the ECU would configure its ECU deserializer, the sensor serializer and the sensor. The TAP structure induces a second set of TAP deserializer and TAP serializer which also needs to be configured via CASL scripts.

FPDLINK3 deserializers are automatically bound to their connected serializers and transfer the serializer I2C slave address as I2C aliasing destination address into the deserializer. This leads to the situation where the ECU would try to configure the TAP serializer as needed for the sensor. Since this configuration might violate the requirements of section Required Serializer and Deserializer Settings on page 241, remote accesses to the TAP serializer are disabled via the LOCAL_WRITE_DISABLE bit in the TAP serializer. ECU write attempts to the TAP serializer will appear successful to the ECU but will not change any registers in the serializer. As a result, the configuration of the TAP serializer must also be done via CASL scripts.

The configuration of the sensor is left to the ECU which allows dynamic sensor reconfiguration and status polling.

Note
ECU write accesses to the TAP serializer are disabled via the LOCAL_WRITE_DISABLE bit in the TAP serializer. This must be considered regarding its impact to functional safety implementations.

Note
This behaviour differs from GMSL cards, where the ECU controller can still configure the sensor serializer and the sensor (see section Changing the Mode of Operation on page 221).

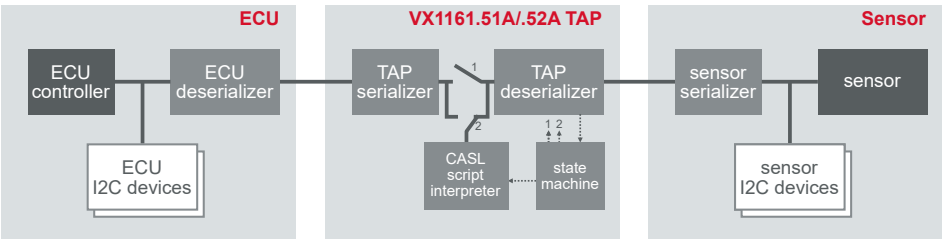


Figure 57: TAP I2C architecture; ECU configures ECU deserializer + sensor

The I2C address of the ECU deserializer does not change by inducing the TAP.

The I2C address of the TAP deserializer and the TAP serializer can be configured in VXconfig. Both chips start with their hardware strapped address and are then overwritten by the VX1161 firmware with the address configured in VXconfig. Additionally, the SetConfig(8,x) CASL command can be used to change the I2C address dynamically.

The sensor serializer I2C address is defined by its hardware straps. It could be changed via CASL scripts.

The sensor I2C address is defined by the sensor itself. To allow I2C accesses from the ECU to the sensor the I2C alias registers in the TAP deserializer need to be set via CASL scripts.

The additional TAP link will cause additional transmission delays (via clock stretch-

ing) to any sensor registers, which will reduce the effective control channel bandwidth and could affect system timings.

Modified Registers

The following register values are modified by the VX1161 firmware during initialization:

Register Name	Register Address	Default Value	Modified Value	Description
Deserializer				
I2C_DEVICE_ID	0x00	-	configured value	Device ID is set according configuration
FWD_CTL1	0x20	0x30	0x33	Bit 4 : FWD_PORT0_DIS is set; enables data transmission from port 0 (connected cable) Bit 5 : FWD_PORT1_DIS is set; enables data transmission from port 1 (not connected)
CSI_PLL_CTL	0x1F	0x02	configured value	Bit 1 and 0 : CSI_TX_SPEED set to the CSI speed, selected in VXconfig; required to be compatible with the VX1161.51 hardware
CSI_CTL	0x33	0x00	0x03	Bit 1: CSI_CONTS_CLOCK is set; enables continuous csi clock; required by VX1161.51 hardware Bit 0: CSI_ENABLE is set; enables CSI port to output data
CSI_TEST_CTL	0x38	0x00	0x00	Bit 3:0 : CSI_TEST_CTL set and cleared; triggers CSI output to activate continuous clock; required by VX1161.51 hardware
Serializer				
I2C_DEVICE_ID	0x00	-	configured value	Device ID is set according configuration
I2C_CONTROL2	0x09	0x1E	0x9E	Bit 7 : LCL_WRITE_ DISABLE is set; disables configuration accesses from the connected deserializer; prevents configurations by the ECU meant for the sensor serializer which are not compatible to the VX1161.51 hardware
General_CFG	0x02	0x33	0x73	Bit 6: CONTS_CLK is set; enables the continuous CSI clock mode; required to be compatible with the VX1161.51 hardware

Detection of TAP by ECU Firmware for Measurement Mode

In cases where the ECU needs to know whether a TAP is present, CASL scripts can be used to manipulate default register values in the TAP serializer (e. g. PGEN_LINE_SIZE1 (indirect address 0x04) in DS90UB953). Care must be taken about the risk of racing conditions between script execution and ECU register accesses.

GPIO Interconnect

For forwarding GPIOs from the ECU to the sensor, the TAP deserializer and the sensor serializer registers must be configured properly. This can be done via CASL scripts.

Forwarding of analog serializer features is not supported.

On FPDLINK3, GPIOs are sampled and transferred via control channel packets. Those packets are 30 bit in length and transmitted via the selected control channel baud rate of 50 Mbps, 10 Mbps or 2.5 Mbps. This results in an additional delay of $t_{delayGPIO}$.

In cases where camera sampling triggers are transferred via GPIOs, it must be considered that the control channels for two TAP links are not synchronized. Therefore, a skew between the two independent VX1161.51 TAP channels of twice $t_{delayGPIO}$ must be assumed.

The GPIOs of the TAP serializer are bidirectionally connected to the GPIOs of the deserializer and the Sensor Trigger Generator GPIOs as follows:

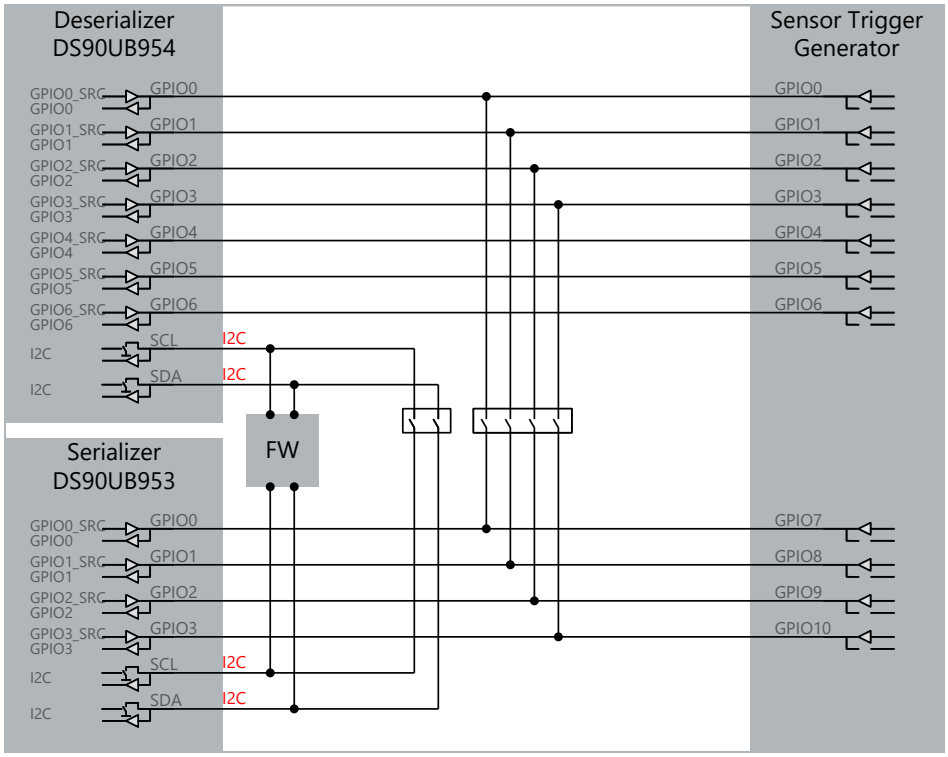


Figure 58: TAP GPIO architecture; GPIO connections between TAP deserializer, TAP serializer and Sensor Trigger Generator; switches are closed when TAP deserializer link is stable

Signal of Serializer		Connected to Signal of Deserializer		Signal of Sensor Trigger Generator
Name	Dir	Name	Dir	Name
CH2 - GPIO_0	IO	CH1 - GPIO_0	IO	CH1 - GPIO_0 = CH2 - GPIO_7
CH2 - GPIO_1	IO	CH1 - GPIO_1	IO	CH1 - GPIO_1 = CH2 - GPIO_8
CH2 - GPIO_2	IO	CH1 - GPIO_2	IO	CH1 - GPIO_2 = CH2 - GPIO_9
CH2 - GPIO_3	IO	CH1 - GPIO_3	IO	CH1 - GPIO_3 = CH2 - GPIO_10
-	-	CH1 - GPIO_4	IO	CH1 - GPIO_4
-	-	CH1 - GPIO_5	IO	CH1 - GPIO_5
-	-	CH1 - GPIO_6	IO	CH1 - GPIO_6
CH4 - GPIO_0	IO	CH3 - GPIO_0	IO	CH3 - GPIO_0 = CH4 - GPIO_7
CH4 - GPIO_1	IO	CH3 - GPIO_1	IO	CH3 - GPIO_1 = CH4 - GPIO_8
CH4 - GPIO_2	IO	CH3 - GPIO_2	IO	CH3 - GPIO_2 = CH4 - GPIO_9
CH4 - GPIO_3	IO	CH3 - GPIO_3	IO	CH3 - GPIO_3 = CH4 - GPIO_10
-	-	CH3 - GPIO_4	IO	CH3 - GPIO_4
-	-	CH3 - GPIO_5	IO	CH3 - GPIO_5
-	-	CH3 - GPIO_6	IO	CH3 - GPIO_6

4.5.15.6 Channel in RX Mode

If a channel group is configured in RX mode, the incoming FPDLINK3 data stream on channel 1/3 is received by the DS90UB954 deserializer and forwarded to the recording data processing. The serializers DS90UB953 on channel 2/4 are powered-off.

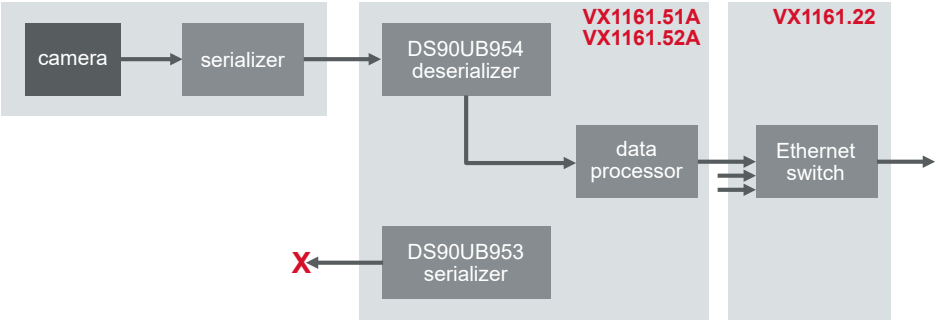


Figure 59: TAP channel in RX mode; payload reception

Initialization

The initialization sequence of the sensor link follows this chart:

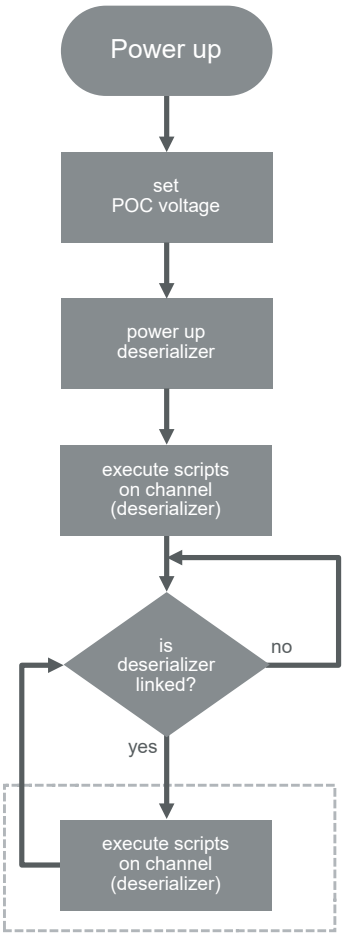


Figure 60: RX initialization sequence

After power-up, the POC voltage is applied as configured and the deserializer is activated.

Next, the enabled scripts for the RX channel are executed immediately (script 1, 2, 3) to configure the deserializer, sensor serializer and sensor as early as possible.

After the initial run, the script execution engine waits for a deserializer lock.

When the deserializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the follwing conditions apply:

- Script is not configured as "run once" (CASL SetConfig(7,1))
- OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
- OR script terminated with a return value ≠ 0 in previous run

If the deserializer link unlocks, the script sequence repeates from the inital deserializer run on.

I2C Architecture

In RX mode, the VX1161.51 deserializer, sensor serializer and sensor need to be configured via CASL scripts. Dynamic sensor reconfiguration is currently only possible via CASL scripts.

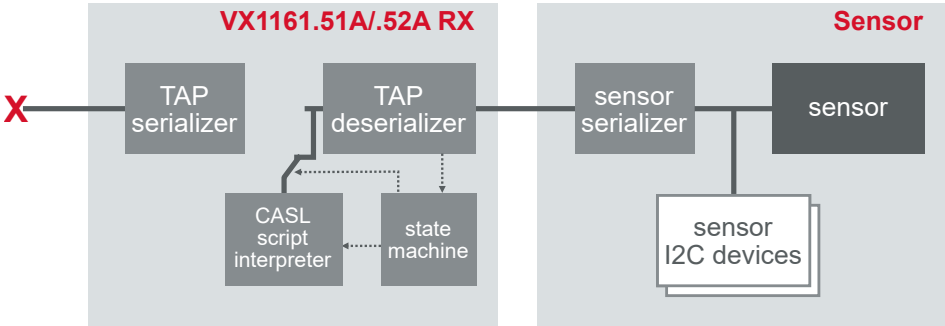


Figure 61: RX mode; I2C architecture; CASL scripts configure deserializer, serializer and sensor

The I2C address of the TAP deserializer can be configured in VXconfig. The chip starts with its hardware strapped address and is then overwritten by the VX1161 firmware with the address configured in VXconfig. Additionally, the SetConfig(8,x) CASL command can be used to change the I2C address dynamically.

The sensor serializer I2C address is defined by its hardware straps and could be changed via CASL scripts. The deserializer will bind automatically if configured properly. Therefore, no I2C address adjustments should be required.

The sensor I2C address is defined by the sensor itself. CASL scripts can be used to configure the sensor via I2C control channel accesses. To enable I2C control channel accesses via CASL scripts, the I2C alias registers in the deserializer must be set properly.

The VX1161.51 does not emulate any I2C devices (besides the TAP serializer). In order to avoid error states in the attached ECU, the ECU firmware must be modified to omit or ignore erroneous I2C accesses.

Modified Registers

The following register values are modified by the VX1161 firmware during initialization:

Register Name	Register Address	Default Value	Modified Value	Description
Deserializer				
I2C_DEVICE_ID	0x00	-	configured value	Device ID is set according configuration
FWD_CTL1	0x20	0x30	0x33	Bit 4 : FWD_PORT0_DIS is set; enables data transmission from port 0 (connected cable) Bit 5 : FWD_PORT1_DIS is set; enables data transmission from port 1 (not connected)
CSI_PLL_CTL	0x1F	0x02	configured value	Bit 1 and 0 : CSI_TX_SPEED set to the CSI speed, selected in VXconfig; required to be compatible with the VX1161.51 hardware
CSI_CTL	0x33	0x00	0x03	Bit 1: CSI_CONTS_CLOCK is set; enables continuous csi clock; required by VX1161.51 hardware Bit 0: CSI_ENABLE is set; enables CSI port to output data
CSI_TEST_CTL	0x38	0x00	0x00	Bit 3:0 : CSI_TEST_CTL set and cleared; triggers CSI output to activate continuous clock; required by VX1161.51 hardware

GPIO Interconnect

The Sensor Trigger Generator can be used to toggle GPIOs of the deserializer. These signals can be forwarded to the connected serializer at the sensor and to the sensor to trigger sampling a frame simultaneously on several channels. To use this function, the deserializer, the sensor serializer and the sensor registers must be configured properly. This can be done via CASL scripts.

On FPDLINK3, GPIOs are sampled and transferred via control channel packets. Those packets are 30 bit in length and transmitted via the selected control channel baud rate of 50 Mbps, 10 Mbps or 2.5 Mbps. This results in a delay of $t_{delayGPIO}$.

In cases where sensor triggers are transferred via GPIOs it must be considered that the control channels for two links are not synchronized. Therefore, a skew between the two independent VX1161.51 channels of twice $t_{delayGPIO}$ must be assumed.

The GPIOs of the deserializers are connected to the GPIOs of the sensor trigger generator as follows:

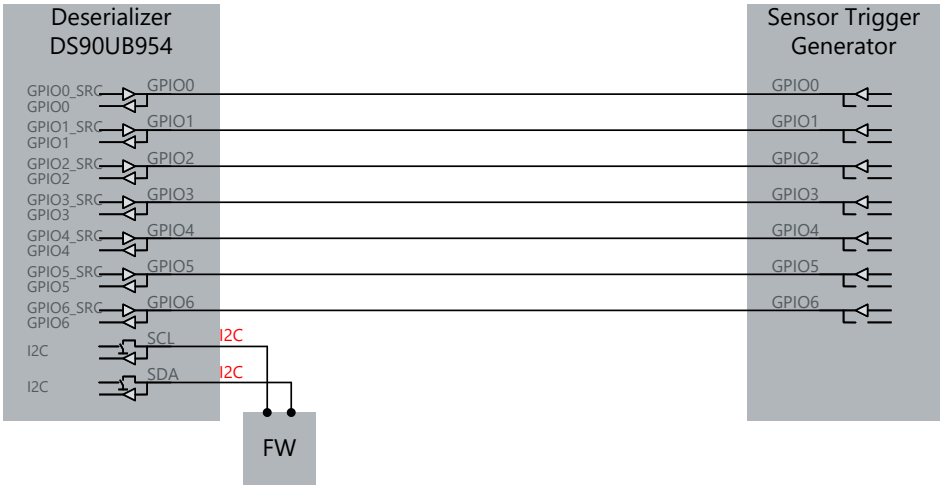


Figure 62: RX GPIO architecture; GPIO connections between RX deserializer and Sensor Trigger Generator

Signal of Deserializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**
CH1- GPIO_0	IO	CH1 - GPIO_0
CH1- GPIO_1	IO	CH1 - GPIO_1
CH1- GPIO_2	IO	CH1 - GPIO_2
CH1- GPIO_3	IO	CH1 - GPIO_3
CH1- GPIO_4	IO	CH1 - GPIO_4
CH1- GPIO_5	IO	CH1 - GPIO_5
CH1- GPIO_6	IO	CH1 - GPIO_6
CH3 - GPIO_0	IO	CH3 - GPIO_0
CH3 - GPIO_1	IO	CH3 - GPIO_1
CH3 - GPIO_2	IO	CH3 - GPIO_2
CH3 - GPIO_3	IO	CH3 - GPIO_3
CH3 - GPIO_4	IO	CH3 - GPIO_4
CH3 - GPIO_5	IO	CH3 - GPIO_5
CH3 - GPIO_6	IO	CH3 - GPIO_6

4.5.15.7 Channel in TX Mode

If a channel group is configured in TX mode, an incoming data stream from Ethernet is output via the DS90UB953 serializer towards an ECU for example. The deserializers on channel 1/3 are disabled.

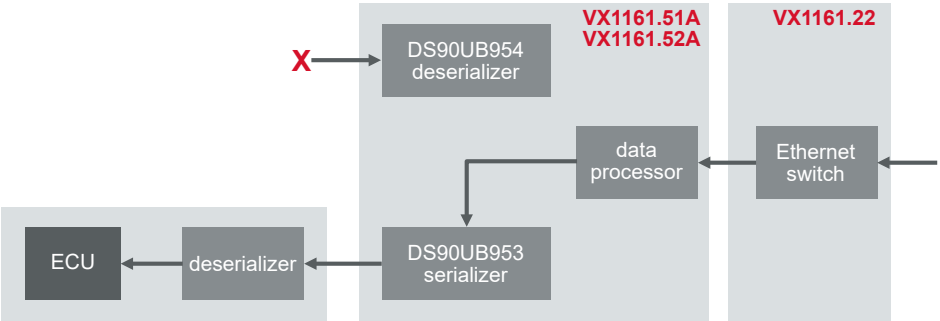


Figure 63: TAP channel in TX mode; payload transmission

Initialization

The initialization sequence of the sensor link follows this chart:

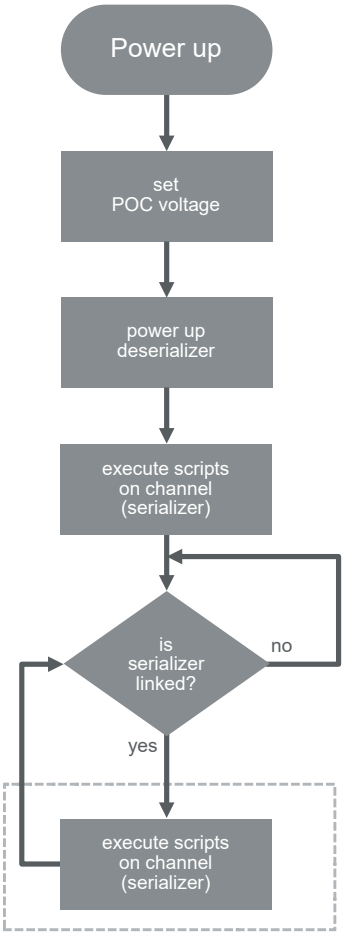


Figure 64: TX initialization sequence

After power-up, the serializer is immediately activated and the enabled scripts for the TX channel are immediately executed (script 1, 2, 3) to configure the serializer and the ECU deserializer as early as possible.

After the initial run, the script execution engine waits for a serializer lock.

When the serializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the following conditions apply:

1. Script is not configured as "run once" (CASL SetConfig(7,1))
2. OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
3. OR script terminated with a return value ≠ 0 in previous run

If the serializer link unlocks, the sequence repeats from the initial serializer run on.

I2C Architecture

In TX mode, the serializer needs to be configured via CASL scripts. Since its configuration must match the hardware architecture, remote configuration accesses are disabled.

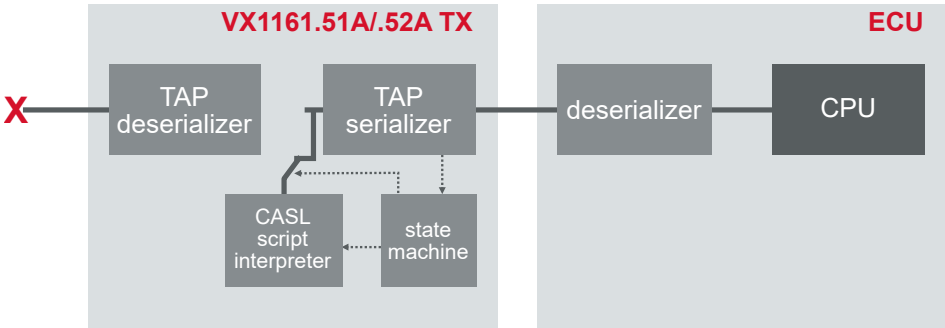


Figure 65: TX mode; I2C architecture; CASL scripts configure the serializer.

The I2C address of the TAP serializer can be configured in VXconfig. The chip starts with its hardware strapped address and is then overwritten by the VX1161 firmware with the address configured in VXconfig. Additionally, the SetConfig(8,x) CASL command can be used to change the I2C address dynamically.

Since the ECU deserializer will automatically bind to the TAP serializer, no address adjustments in the ECU firmware are required.

FPDLINK3 deserializers are automatically bound to their connected serializers and transfer the serializer I2C slave address as I2C aliasing destination address into the deserializer. This leads to the situation where the ECU would try to configure the TAP serializer as needed for the sensor. Since this configuration might violate the requirements of section Required Serializer and Deserializer Settings on page 241, remote accesses to the TAP serializer are disabled via the LOCAL_WRITE_DISABLE bit in the TAP serializer. ECU write attempts to the TAP serializer will appear successful to the ECU but will not change any registers in the serializer. As a result, the configuration of the TAP serializer must also be done via CASL scripts.

The VX1161.51 does not emulate any I2C devices (besides the serializer). In order to avoid error states in the attached ECU, either the Channel Group configuration 1TX with sensor can be used or the ECU firmware must be modified to omit or ignore erroneous I2C accesses. The 1TX with sensor mode acts logically like the

TAP mode but video data from the sensor is ignored and data from Ethernet is sent to the ECU .

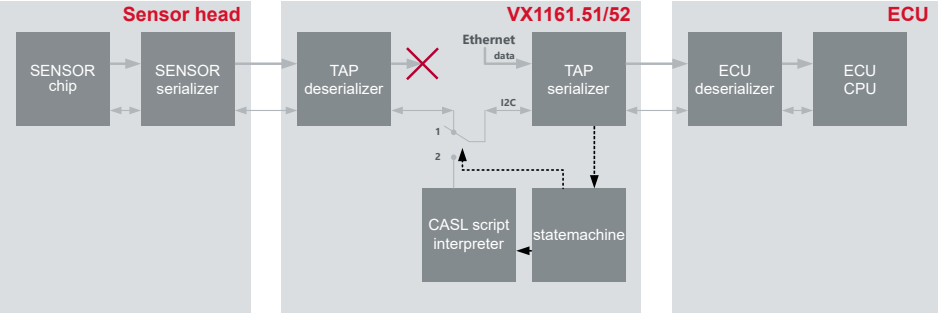


Figure 66: 1TX with sensor mode; ECU I2C accesses forwarded to sensor head; stream data for ECU supplied via Ethernet

The TX mode can be used to replace a sensor head which usually consists of the sensor serializer and the sensor chip. In typical applications both chips are configured by the ECU firmware via I2C. Since the VX1161.51 only provides the serializer and does not emulate any other I2C devices, the attached ECU firmware will fail accessing the sensor chip and enter an error state. To overcome this issue, either the ECU firmware must be enabled to cope with the missing sensor (code instrumentation, HIL mode) or the "1TX with sensor" mode in the Channel Group configuration can be used. The 1TX with sensor mode uses a real sensor head. I2C and GPIOs are forwarded to the sensor head like in TAP mode but data from the sensor is ignored and data from Ethernet is sent to the ECU.

Modified Registers

The following register values are modified by the VX1161 firmware during initialization:

Register Name	Register Address	Default Value	Modified Value	Description
Serializer				
I2C_DEVICE_ ID	0x00	-	configured value	Device ID is set according configuration
I2C Control 2	0x09	0x1E	0x9E	Bit 7 : LCL_WRITE_ DISABLE is set; disables configuration accesses from the connected deserializer; prevents configurations by the ECU meant for the sensor serializer which are not compatible to the VX1161.51 hardware
General_CFG	0x02	0x33	0x73	Bit 6: CONTS_CLK is set; enables the continuous CSI clock mode; required to be compatible with the VX1161.51 hardware

GPIO Interconnect

The GPIOs of the serializer are connected to the GPIOs of the VX1161.51. The connections to the TAP deserializer are switched off.

For frame synchronization, one GPIO can be used as frame trigger input of the VX1161.51 to output a frame (see section Channel in TX Mode on page 234). Additionally, the Sensor Trigger Generator can be used to toggle one serializer GPIO per channel. Because one GPIO per channel can be used to trigger the frame transmission, the generator can be used as frame rate generator to the Frame Trigger Detector (see section Channel in TX Mode on page 234).

The GPIOs of the serializer are connected to the GPIOs of the sensor trigger generator as follows:

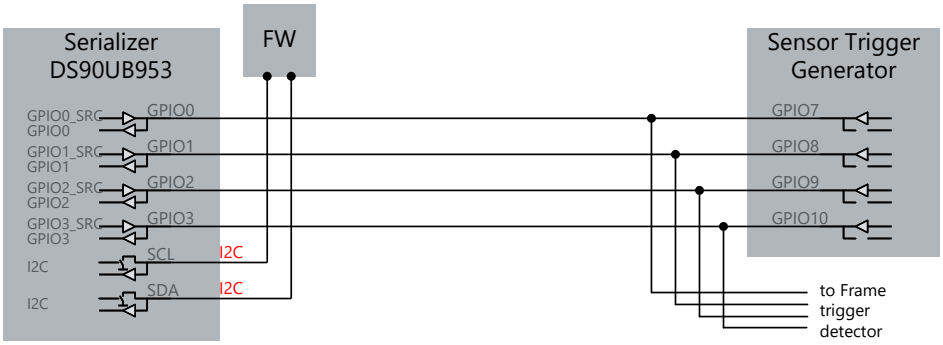


Figure 67: TX GPIO architecture; GPIO connections between the TX serializer, the Sensor Trigger Generator and the Frame Trigger Detector

Signal of Serializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**
CH2- GPIO_0	IO	CH2 - GPIO_7
CH2- GPIO_1	IO	CH2 - GPIO_8
CH2- GPIO_2	IO	CH2 - GPIO_9
CH2- GPIO_3	IO	CH2- GPIO_10
CH4 - GPIO_0	IO	CH4 - GPIO_7
CH4 - GPIO_1	IO	CH4 - GPIO_8
CH4 - GPIO_2	IO	CH4 - GPIO_9
CH4 - GPIO_3	IO	CH4 - GPIO_10

4.5.15.8 Sensor Trigger Generator

The VX1161.51/52 provides a Sensor Trigger Generator for every channel. This generator can be used to generate a trigger signal to one of the GPIO/MFP pins. If the serializer and deserializer are configured correctly via CASL scripts, this trigger signal will be forwarded to the sensor and triggers the acquisition of data.

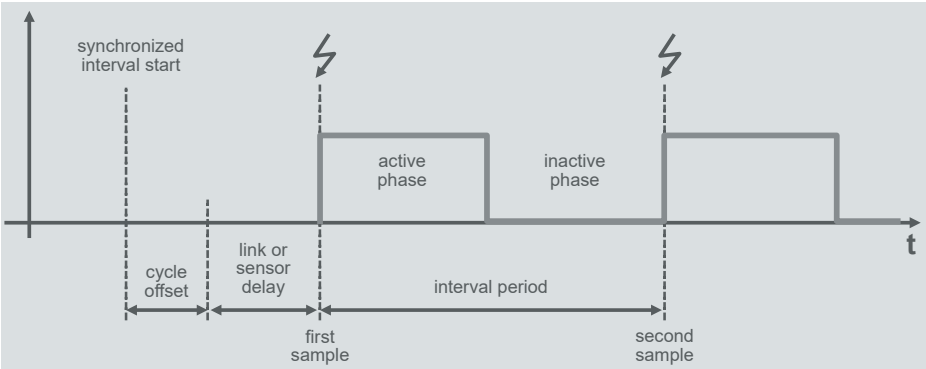


Figure 68: Sensor Trigger Generator with adjustable interval period

The generators of one module use one hardware timer as time base. This timer can be synchronized via PTP between multiple cards. If the PTP service is available, the trigger generation on all channels on all synchronized modules will be synchronous. If the PTP service is not available, only the channels on one module will be synchronous.

The current implementation requires that every VX1161.51/52 can exchange Ethernet broadcast messages with every other VX1161.51/52 during measurement start. These broadcast messages are used to negotiate the precise start of the first Frame Trigger period.

If a measurement setup consists of several VX1161 racks, there must be a path the broadcast messages can take to reach all other cards. Since Windows natively does not forward Ethernet broadcast messages from a receiving port to all other Ethernet ports, another connection is needed. To provide that path, an additional Ethernet cable between all VX1161 racks is required. The Ethernet switch inside the VX1161.22 uplink module will distribute those broadcast messages. Any Ethernet port on the VX1161.22 can be used due to the low bandwidth requirement.

The generators start outputting their trigger signal at start of measurement of the receiving application and it will stop generation at measurement stop. As a result, the sensor might not generate data when the measurement is stopped.
When the measurement is started, the time on all channels starts simultaneously and all generators start with their cycle offset times.

The trigger signal has two phases: the active and the inactive phase. Both can be configured in their duration. Together they define the trigger interval period, which defines the frequency of trigger pulses and the frame rate. Trigger generation always starts with the active phase. The Cycle offset parameter can be used to compensate delays in serial links or sensors. For sensors on fast paths, a delay can be added to trigger at the same time as sensors on slow paths.

For sensors requiring falling edges as trigger, the polarity of the output signal can be selected. If not inverted, the active phase starts with a rising edge which is intended to trigger the sensor.

Note
Care must be taken to avoid signal collisions between serializer or deserializer outputs and Sensor Trigger Generator outputs.

The Sensor Trigger Generator is configured via VXconfig and the following parameters:

Setting	Description
Enable	Activates the Sensor Trigger Generator. If disabled, the GPIOs are high impedance.
Cycle Offset	Defines a static delay from the ideal interval start in nano seconds. Can be used to compensate different delays in between several links or sensors.
Active Phase	Defines the duration of the active phase in nano seconds.
Inactive Phase	Defines the duration of the inactive phase in nano seconds.
GPIO output pin to trigger	GPIO pin of the VX1161.51/52, where the trigger signal shall be output. Check the GPIO/MFP section of your card to see which pins can be used and how they map to deserializer or serializer pins.
GPIO polarity active phase	Can be used to invert the output signal polarity. - checked (default): active phase starts with a rising edge. Active phase is high. Inactive phase is low. - unchecked: active phase starts with a falling edge. Active phase is low. Inactive phase is high.

4.5.15.9 Frame Trigger Detector

The VX1161.51/52 provides a Frame Trigger Detector for every channel in TX mode. This detector can be used to detect the change of one GPIO/MFP pin and trigger the transmission of the next video frame. This feature allows to synchronize frame transmissions to ECU trigger signals and with this to the ECU time domain.

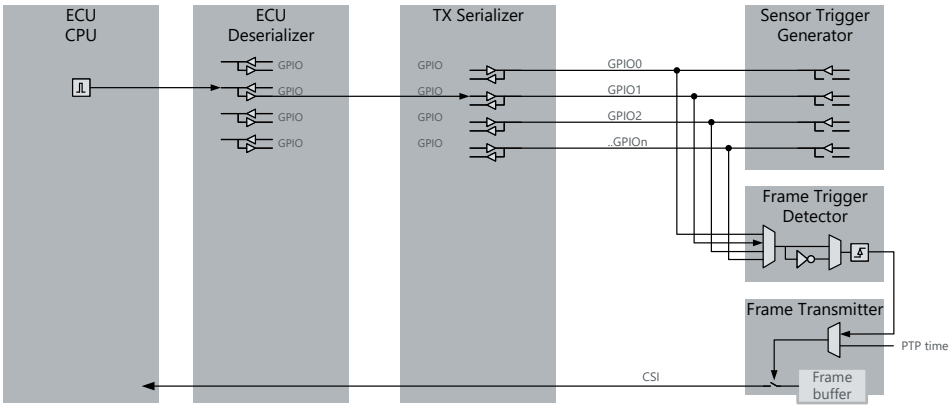


Figure 69: CPU generates frame trigger; Frame Trigger Detector detects edges and triggers the frame transmission

The detector can be fed by one GPIO/MFP of the channel group the channel belongs to. For selecting the active rising or falling edge, chose the polarity of the trigger signal.

Note
Care must be taken to avoid signal collisions between serializer outputs and Sensor Trigger Generator outputs.

The Frame Trigger Detector is configured via VXconfig and the following parameters in the Output Description section.

Setting	Description
GPIO input pin	GPIO pin of the VX1161.51/52, which shall feed the edge detector. Check the GPIO/MFP section of your card to see which pins can be used and how they map to your serializer pins.
GPIO polarity active phase	Specifies whether rising edges (checked) or falling edges (unchecked) on the given GPIO pin trigger the frame transmission.

4.5.15.10 Required Serializer and Deserializer Settings

The following requirements apply to the serializer and deserializer on the VX1161.51. The settings for the connected serializer at the sensor and deserializer in the ECU are allowed to differ according FPDLINK3 capabilities.

FPDLINK3serializers are sensitive to incompatible CSI clocking schemes. If a serializer and its feeding data source are configured differently, the serializer will not accept any valid data.

CSI Continuous Clock

Due to internal technical reasons, the CSI clocks on all channels must run in continuous CSI clock mode only. This mode is initially configured by the VX1161.51 firmware. Changing this setting will interrupt reception of data.

Note
ECU write accesses to the TAP serializer are disabled via the LOCAL_WRITE_DISABLE bit in the TAP serializer. This must be considered regarding its impact to functional safety implementations.

CSI Clock Lane Speed

Due to internal reasons, the CSI clock speed of the serializer must be configured via the Channel Group Mode and not be changed. The VX1161.51 firmware will configure this at start-up. If this setting is changed via CASL scripts, data transmission will no longer be possible.

CSI Signal Settings

The CSI lanes of the TAP serializer DS90UB953 do not require any lane swap, lane inversion or timing adjustments.

4.5.15.11 POC Supply

The VX1161.51A delivers power to connected sensor heads via the coax cable using Power-over-Coax (POC). It has an internal, isolated supply that is adjustable and independent of VX GND. It can source up to 2 W. The voltage can be switched off or adjusted from 5 V ... 12 V using VXconfig.

In TAP mode, the internal supply can be deactivated, allowing the POC supply of the ECU to be forwarded to the sensor head. This gives the ECU control over the sensor's power.

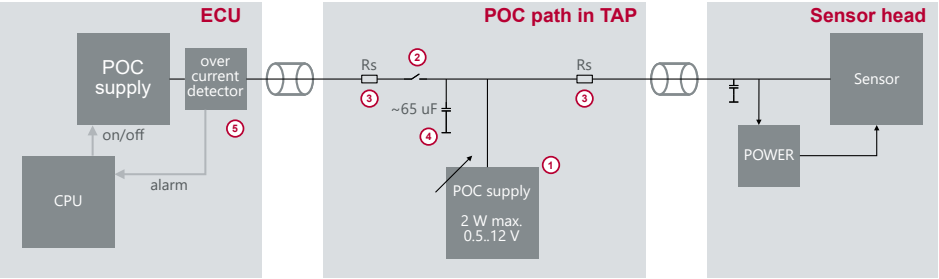


Figure 70: POC structure in VX1161.51A TAP channels

The internal series resistance (Rs) (3) of the POC bypass path is approximately 2 x 2.5 Ohms, and the internal capacitance (4) is approximately 65 µF. This capacitance can cause or increase inrush currents for the ECU that sources power and might trigger overcurrent detection features (5). In such cases, it might be an option to source the sensor from the VX (1) instead of the ECU, but then the ECU would have no control over the sensor power.

Furthermore, the internal VX POC supply can be output to the ECU for use cases in which CSI ingestion PODs require POC supply.

The POC voltage level, generation, and delivery direction are controlled via the POC voltage settings for the TAP_RX and TAP_TX ports in VXconfig as follows:

Note
Sourcing power in TAP mode is supported since VXtools v4.9.

Settings				
POC voltage V_{RX}on TAP RX channel	0V	5...12V	0V	5...12V
POC voltage V_{TX}on TAP TX channel	0V	0V	5...12V	5...12V
Effects				
POC bypass (2)	closed	open	closed	closed
VX POC source (1) voltage	off	V_{RX}	V_{TX}	Min(V_{RX},V_{TX})
Use case	ECU supplies sensor	VX supplies sensor	VX supplies ingest POD	VX supplies sensor and ingest POD

4.5.15.12 Technical Data

Weight	210 g
Power consumption	► P_{init}; initialization (~500ms): 500mW ► $P_{standby}$; sleep mode: <18 mW ► P_{run}; normal operation excl. POC: VX1161.51 max. 8 W; VX1161.52 max. 10W
POC (power over cable)	► V_{poc}; voltage sourced by VX1161.51/.52: 0 V, 5...12 V configurable ► P_{poc}; maximum power delivered per RX channel; sourced by VX1161.51/.52: 2 W ► $V_{PocTapMax}$; voltage forwarded from ECU by TAP: 3...18 V ► $I_{PocTapMax}$; max. current forwarded from ECU by TAP: 500 mA ► Electrical isolation between TAP_CH1/2 and TAP _CH3/4 and chassis max. 50 V ► Rpoc; DC resistance of POC forwarding path in TAP: typ. 1.132 Ohm
Minimum power -off time	► minimum time between powering-off and re-appying power: >2 s
Start-up time	► t_{init}; initialization after power up: ~ 500 ms ► $t_{wakeLink}$; wake up after sleep; FPDLINK3 link stable: < 50 ms, depends on CASL script execution ► $t_{wakeReady}$; wake up after sleep; FPDLINK3 data to Ethernet: < 10 s
Temperature range	-40 °C ... +60 °C
Maximum bandwidths	► Card to host via Ethernet: 7.2 Gbps net; total for all channels together ► Every TAP / RX / TX channel FPDLINK3: 3.2 Gbps net
Latencies/delays	► t_{delay}; FPDLINK3 TAP; TAP_RX to TAP_TX; forward, backward, I2C and GPIO channels: ~ 1 µs. Additional serializer and deserializer delay: ~ one CSI packet. ► $t_{delayGPIO;}$ additional delay of GPIOs; forward, backward: dependent on control channel baud rate; 600 ns @ 50 Mbps, 3 µs @ 10 Mbps or 12.5µs @ 2.5 Mbps
Connector	(Single signal) FAKRA Type Z; Rosenberger 59S2AQ-40MT5-Z; color: cyan; compatible to all other (single signal) FAKRA connector types; 6 GHz max; >=25 mating cycles
Maximum cable length	8 m, in both RX and TX direction at the same time
Part number	► VX1161.51A: 180152 ► VX1161.52A: 180177

4.5.16 VX1161.51B 4xRX FPDLINK3 TI954/TI954

The VX1161.51B 4xRX FPDLINK3 TI954/TI954 card is a four-channel RX card for FPDLINK3 data streams. All channels contain one Texas Instruments DS90UB954 deserializer for receiving data.

The card can be configured with VXconfig and updated to new software versions with VXupdate. Initialization of the transceivers and remote sensors can be done via CASL language scripts within VXconfig.

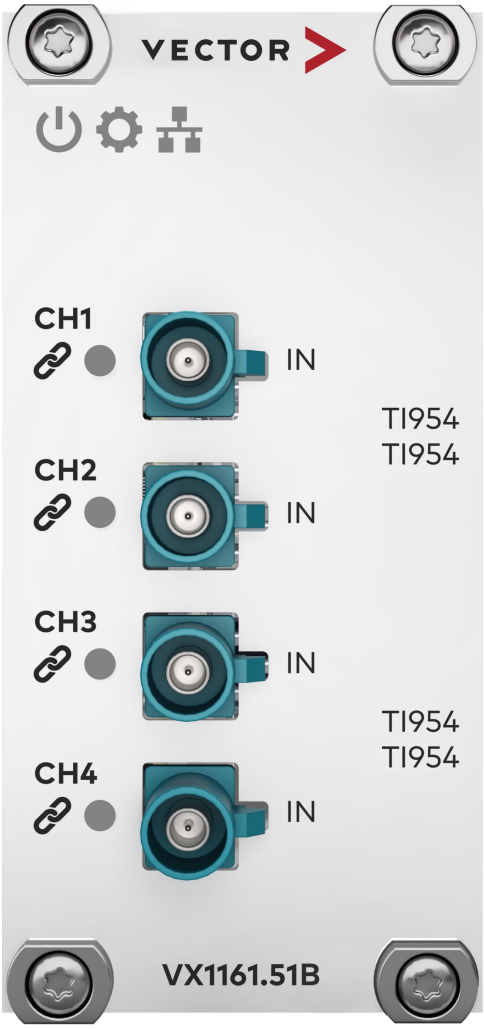


Figure 71: VX1161.51B 4xRX TI954

Note
VX1161.51B is supported by VXtools v4.3 and later.

4.5.16.1 Connectors

- **CH1...4**
  The VX1161.51B provides FAKRA coaxial connectors of Type Z for connecting data sources such as video or radar sensors. Type Z connectors are compatible to all other coding types of single signal FAKRA connectors. The connector properties can be found in section Connectors on page 245.

4.5.16.2 LEDs

The status of the card is displayed by system LEDs in the lid top. The status of every channel is displayed by a channel LED next to every channel connector.

Power

Symbol	Sequence	Description
	R R R R R R R R	Device is powered but Firmware not running. The Power LED will light up red at the beginning of the boot sequence of the Interface Card. ► Please contact Vector VX support if the Power LED stays red for more than 10 seconds. In this case a critical error has occurred.
	G G G G G G G G	Firmware loaded successfully.
	G ☐ G ☐ G ☐ G ☐	Response to ”Identify VX device command” (VXconfig).

System

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Firmware not loaded.
	R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Interface Card, its system components are initialized. If a component fails to initialize properly, the System LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
	G G G G G G G G	Device is in normal operating mode.
	G ☐ G ☐ G ☐ G ☐	Firmware update in progress.

Ethernet

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Backplane link not established.
	R R R R R R R R	Backplane link error.
	G G G G G G G G	Backplane link established.
	G ☐ G ☐ G ☐ G ☐	Backplane link activity.

Channel

Symbol	Sequence	Description
	□□□□□□□□	Channel disabled.
	R R R R □□□□	Channel enabled, invalid configuration. See VXconsole for details.
	R R R R R R R R	Channel enabled, no link established.
	G G G G G G G G	Channel enabled, link established, no data traffic.
	G □ G □ G □ G □	Channel enabled, link established, data traffic.

4.5.16.3 Boot and Link-up-Time

When the VX1161 is connected to power, the streaming card must prepare itself for a fast wake-up. For about t_{init} the power consumption will be P_{init}. After that time, the card enters standby mode and the power consumption is reduced to its minimum $P_{standby}$.

When the VX1161 is woken up (see section Boot and Link-up-Time on page 246), the card is already prepared and can establish both ECU and sensor link within $t_{wakeLink}$. After wakeup, the card can provide data to the recorder after $t_{wakeReady}$.

The power consumption during normal operation consist of the maximum power of P_{run} drawn by the card as well as the power consumed by supplied cameras P_{POC}.

4.5.16.4 Changing the Mode of Operation

The VX1161.51B can be used in one of the following use cases:

► Video RX (video reception / logging)

In this mode, sensors or ECUs with FPDLINK3 transceivers can be directly attached to log streams of data.

This setting affects all channels in parallel and it is not possible to run different channels or channel groups in different modes. Different cards can be used in different modes.

4.5.16.5 Channel in RX Mode

The VX1161.51B can only be used in Video RX mode only.

If a channel is enabled in RX mode (RX_only in VXconfig), the incoming FPDLINK3 data stream is received by the DS90UB954 deserializer and forwarded to the recording data processing.

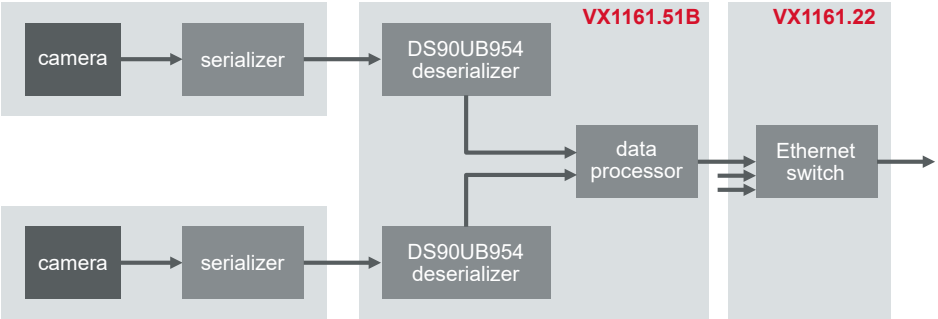


Figure 72: Channel in RX mode; payload reception

Initialization

The initialization sequence of the sensor link follows this chart:

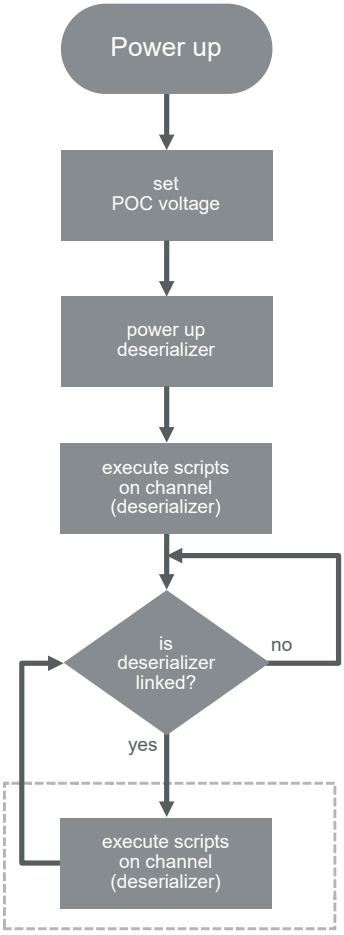


Figure 73: RX initialization sequence

After power-up, the POC voltage is applied as configured and the deserializer is activated.

Next, the enabled scripts for the RX channel are executed immediately (script 1, 2, 3) to configure the deserializer, sensor serializer and sensor as early as possible.

After the initial run, the script execution engine waits for a deserializer lock.

When the deserializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the follwing conditions apply:

- Script is not configured as "run once" (CASL SetConfig(7,1))
- OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
- OR script terminated with a return value ≠ 0 in previous run

If the deserializer link unlocks, the script sequence repeates from the inital deserializer run on.

I2C Architecture

In RX mode, the deserializer, sensor serializer and sensor need to be configured via CASL scripts. A dynamic sensor reconfiguration is not supported.

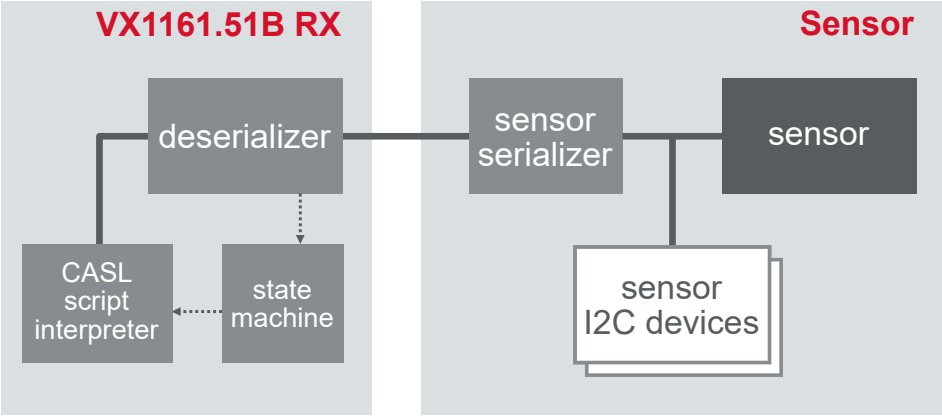


Figure 74: RX mode; I2C architecture; CASL scripts configure deserializer, serializer and sensor

The I2C address of the TAP deserializer can be configured in VXconfig. The chip starts with its hardware strapped address and is then overwritten by the VX1161 firmware with the address configured in VXconfig. Additionally, the SetConfig(8,x) CASL command can be used to change the I2C address dynamically.

The sensor serializer I2C address is defined by its hardware straps and could be changed via CASL scripts. The deserializer will bind automatically if configured properly. Therefore, no I2C address adjustments should be required.

The sensor I2C address is defined by the sensor itself. CASL scripts can be used to configure the sensor via I2C control channel accesses. To enable I2C control channel accesses via CASL scripts, the I2C alias registers in the deserializer must be set properly.

Modified Registers

The following register values are modified by the VX1161 firmware during initialization:

Register Name	Register Address	Default Value	Modified Value	Description
Deserializer				
I2C_DEVICE_ID	0x00	-	configured value	Device ID is set according configuration
FWD_CTL1	0x20	0x30	0x33	Bit 4 : FWD_PORT0_DIS is set; enables data transmission from port 0 (connected cable) Bit 5 : FWD_PORT1_DIS is set; enables data transmission from port 1 (not connected)
CSI_PLL_CTL	0x1F	0x02	0x02	Bit 1 and 0 : CSI_TX_SPEED set to value 2; sets the CSI lane speed to 800 Mbps; required to be compatible with the VX1161.51 hardware
CSI_CTL	0x33	0x00	0x03	Bit 1: CSI_CONTS_CLOCK is set; enables continuous csi clock; required by VX1161.51 hardware Bit 0: CSI_ENABLE is set; enables CSI port to output data
CSI_TEST_CTL	0x38	0x00	0x00	Bit 3:0 : CSI_TEST_CTL set and cleared; triggers CSI output to activate continuous clock; required by VX1161.51 hardware

GPIO Interconnect

The Sensor Trigger Generator can be used to toggle GPIOs of the deserializer. These signals can be forwarded to the connected sensor serializer and from there to the sensor to trigger sampling a frame simultaneously on several channels. To use this function, the deserializer, the sensor serializer and the sensor registers must be configured properly. This can be done via CASL scripts.

On FPDLINK3, GPIOs are sampled and transferred via control channel packets. Those packets are 30 bit in length and transmitted via the selected control channel baud rate of 50 Mbps, 10 Mbps or 2.5 Mbps. This results in a delay of $t_{delayGPIO}$.

In cases where sensor triggers are transferred via GPIOs, it must be considered that the control channels for two links are not synchronized. Therefore, a skew between the two independent VX1161.51 channels of twice $t_{delayGPIO}$ must be assumed.

The GPIOs of the deserializers are connected to the GPIOs of the Sensor Trigger Generator as follows:

Signal of Deserializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**
CH1 - GPIO_0	IO	CH1 - GPIO_0
CH1 - GPIO_1	IO	CH1 - GPIO_1
CH1 - GPIO_2	IO	CH1 - GPIO_2
CH1 - GPIO_3	IO	CH1 - GPIO_3
CH1 - GPIO_4	IO	CH1 - GPIO_4

Signal of Deserializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**
CH1 - GPIO_5	IO	CH1 - GPIO_5
CH1 - GPIO_6	IO	CH1 - GPIO_6
CH2 - GPIO_0	IO	CH2 - GPIO_7
CH2 - GPIO_1	IO	CH2 - GPIO_8
CH2 - GPIO_2	IO	CH2 - GPIO_9
CH2 - GPIO_3	IO	CH2 - GPIO_10
CH2 - GPIO_4	IO	CH2 - GPIO_11
CH2 - GPIO_5	IO	CH2 - GPIO_12
CH2 - GPIO_6	IO	CH2 - GPIO_13
CH3 - GPIO_0	IO	CH3 - GPIO_0
CH3 - GPIO_1	IO	CH3 - GPIO_1
CH3 - GPIO_2	IO	CH3 - GPIO_2
CH3 - GPIO_3	IO	CH3 - GPIO_3
CH3 - GPIO_4	IO	CH3 - GPIO_4
CH3 - GPIO_5	IO	CH3 - GPIO_5
CH3 - GPIO_6	IO	CH3 - GPIO_6
CH4 - GPIO_0	IO	CH4 - GPIO_7
CH4 - GPIO_1	IO	CH4 - GPIO_8
CH4 - GPIO_2	IO	CH4 - GPIO_9
CH4 - GPIO_3	IO	CH4 - GPIO_10
CH4 - GPIO_4	IO	CH4 - GPIO_11
CH4 - GPIO_5	IO	CH4 - GPIO_12
CH4 - GPIO_6	IO	CH4 - GPIO_13

4.5.16.6 Sensor Trigger Generator

The VX1161.51/52 provides a Sensor Trigger Generator for every channel. This generator can be used to generate a trigger signal to one of the GPIO/MFP pins. If the serializer and deserializer are configured correctly via CASL scripts, this trigger signal will be forwarded to the sensor and triggers the acquisition of data.

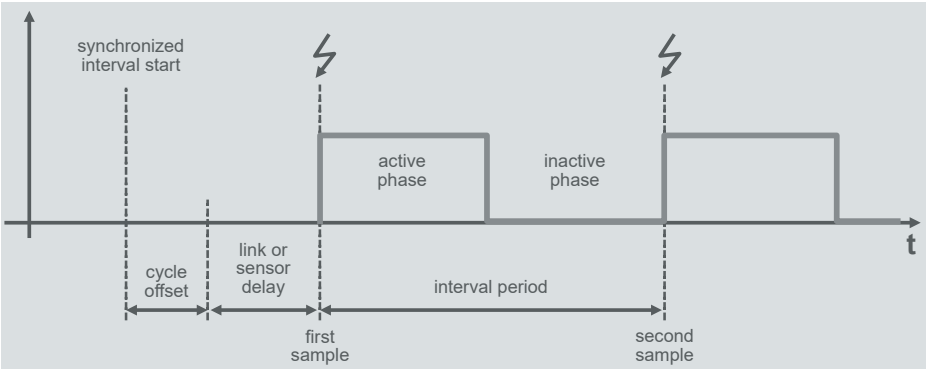


Figure 75: Sensor Trigger Generator with adjustable interval period

The generators of one module use one hardware timer as time base. This timer can be synchronized via PTP between multiple cards. If the PTP service is available, the trigger generation on all channels on all synchronized modules will be synchronous. If the PTP service is not available, only the channels on one module will be synchronous.

The current implementation requires that every VX1161.51/52 can exchange Ethernet broadcast messages with every other VX1161.51/52 during measurement start. These broadcast messages are used to negotiate the precise start of the first Frame Trigger period.

If a measurement setup consists of several VX1161 racks, there must be a path the broadcast messages can take to reach all other cards. Since Windows natively does not forward Ethernet broadcast messages from a receiving port to all other Ethernet ports, another connection is needed. To provide that path, an additional Ethernet cable between all VX1161 racks is required. The Ethernet switch inside the VX1161.22 uplink module will distribute those broadcast messages. Any Ethernet port on the VX1161.22 can be used due to the low bandwidth requirement.

The generators start outputting their trigger signal at start of measurement of the receiving application and it will stop generation at measurement stop. As a result, the sensor might not generate data when the measurement is stopped.
When the measurement is started, the time on all channels starts simultaneously and all generators start with their cycle offset times.

The trigger signal has two phases: the active and the inactive phase. Both can be configured in their duration. Together they define the trigger interval period, which defines the frequency of trigger pulses and the frame rate. Trigger generation always starts with the active phase. The Cycle offset parameter can be used to compensate delays in serial links or sensors. For sensors on fast paths, a delay can be added to trigger at the same time as sensors on slow paths.

For sensors requiring falling edges as trigger, the polarity of the output signal can be selected. If not inverted, the active phase starts with a rising edge which is intended to trigger the sensor.

Note
Care must be taken to avoid signal collisions between serializer or deserializer outputs and Sensor Trigger Generator outputs.

The Sensor Trigger Generator is configured via VXconfig and the following parameters:

Setting	Description
Enable	Activates the Sensor Trigger Generator. If disabled, the GPIOs are high impedance.
Cycle Offset	Defines a static delay from the ideal interval start in nano seconds. Can be used to compensate different delays in between several links or sensors.
Active Phase	Defines the duration of the active phase in nano seconds.
Inactive Phase	Defines the duration of the inactive phase in nano seconds.
GPIO output pin to trigger	GPIO pin of the VX1161.51/52, where the trigger signal shall be output. Check the GPIO/MFP section of your card to see which pins can be used and how they map to deserializer or serializer pins.
GPIO polarity active phase	Can be used to invert the output signal polarity. - checked (default): active phase starts with a rising edge. Active phase is high. Inactive phase is low. - unchecked: active phase starts with a falling edge. Active phase is low. Inactive phase is high.

4.5.16.7 Required Deserializer Settings

The following requirements apply to the deserializer on the VX1161.51. The settings for the connected serializer at the sensor are allowed to differ according FPD-LINK3 capabilities.

FPD-LINK3 serializers are sensitive to incompatible CSI clocking schemes. If a serializer and its feeding data source are configured differently, the serializer will not accept any valid data.

CSI Continuous Clock

Due to internal technical reasons, the CSI clocks on all channels must run in continuous CSI clock mode only. This mode is initially configured by the VX1161.51 firmware. Changing this setting will interrupt reception of data.

CSI Clock Lane Speed

Due to internal reasons, the CSI clock speed of the serializer must be configured via the Channel Group Mode and not be changed. The VX1161.51 firmware will configure this at start-up. If this setting is changed via CASL scripts, data reception and transmission will no longer be possible.

CSI Signal Settings

The CSI lanes of the TAP deserializer DS90UB954 do not require any lane swap, lane inversion or timing adjustments.

4.5.16.8 POC Supply

The VX1161.51 can supply a sensor with power via Power-over-Coax (POC) of up to P_{POC}. The voltage can be adjusted to V_{poc} via VXconfig.

4.5.16.9 Technical Data

Weight	210 g
Power consumption	▶ P_{init}; initialization (~500ms): 500mW ▶ $P_{standby}$; sleep mode: <18 mW ▶ P_{run}; normal operation excl. POC: max.8 W
POC (power over cable)	▶ V_{poc}; voltage sourced by VX1161.51: 0 V, 5...12 V configurable ▶ P_{poc}; maximum power delivered per RX channel; sourced by VX1161.51: 2 W ▶ Electrical isolation beween CH1, CH2, CH3 and CH4 max. 50 V ▶ Electrical isolation beween channels and chassis max. 50 V
Minimum power -off time	▶ minimum time between powering-off and re-appying power: >2 s
Start-up time	▶ t_{init}; initialization after power-up: ~ 500 ms ▶ $t_{wakeLink}$; wake up after sleep; FPD-LINKIII link stable: < 50 ms, depends on CASL script execution ▶ $t_{wakeReady}$; wake up after sleep; FPD-LINKIII data to Ethernet: < 10 s
Temperature range	-40 °C ... +60 °C
Maximum bandwidths	▶ Card to host via Ethernet: 7.2 Gbps net; total for all channels together ▶ Every TAP/ RX channel FPD-LINK3: 3.3 Gbps net
Latencies/delays	▶ t_{delay}; FPD-LINKIII; forward, backward, I2C and GPIO channels: ~ 1 µs. Additional serializer and deserializer delay: ~ one CSI packet. ▶ $t_{delayGPIO;}$ additional delay of GPIOs; forward, backward: dependent on control channel baud rate; 600 ns @ 50 Mbps, 3 µs @ 10 Mbps or 12.5µs @ 2.5 Mbps
Connector	(Single signal) FAKRA Type Z; Rosenberger 59S2AQ-40MT5-Z; color: cyan; compatible to all other (single signal) FAKRA connector types; 6 GHz max; >=25 mating cycles
Maximum cable length	8 m, in both RX and TX direction at the same time
Part number	180153

4.5.17 VX1161.51C/.52C 2xTAP GMSL2 MX9296A/MX9295A

The VX1161.51C/.52C 2xTAP GMSL2 MX9296A/MX9295A card is a two-channel TAP card for GMSL2 data streams. TAP means that a receiver (deserializer) and a transmitter (serializer) are connected back-to-back and the received data is forwarded to the transmitter and duplicated towards the logging system. Both channels contain one Analog Devices® MAX9296A deserializer and one Analog Devices® MAX9295A serializer each.

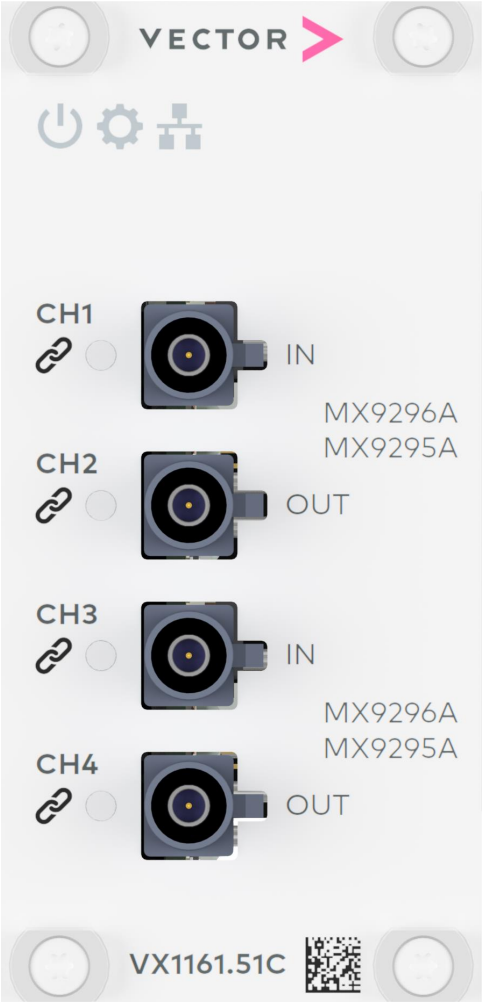


Figure 76: VX1161.51C/.52C 2xTAP GMSL2 MX9296A/MX9295A

The operating mode of the card can be configured either in TAP, RX or TX mode.

The VX1161.52C supports binary lossless compression of video streams on hardware. This compression method reduces the required bandwidth for video transmission, logging and storage typically by 40 to 60 %, depending on video content. Every channel of the VX1161.52 can be compressed. The VX1161.52 compressor pipelines support RAW8/10/12/14/16 and YUV422 8bit pixel formats only. For other pixel formats, please contact Vector.

The card can be configured with VXconfig and updated to new software versions with VXupdate. Initialization of the transceivers and remote sensors can be done via CASL language scripts within VXconfig.

Note
VX1161.51C is supported by VXtools v4.3 and later.
VX1161.52C is supported by VXtools v4.6 and later.

Note
The full GMSL2 payload bandwidth of 4.8 Gbps net is supported by VXtools V4.6 and later.

Note
GMSL systems require after link lock several CSI packets to activate their internal pipelines. This causes the first packets to get lost inside the deserializer or serializer which will corrupt the first frame.

For frame precise measurements or replays, some dummy sensor data must be flushed through the entire sensor ECU connection in advance.

4.5.17.1 Connectors

- **CH1...4**
  The VX1161.51C/.52C provides FAKRA coaxial connectors of Type Z for connecting data sources and data sinks such as video or radar sensors and ECUs. Type Z connectors are compatible to all other coding types of single signal FAKRA connectors. The connector properties can be found in section Technical Data on page 284.

4.5.17.2 LEDs

The status of the card is displayed by system LEDs in the lid top. The status of every channel is displayed by a channel LED next to every channel connector.

Power

Symbol	Sequence	Description
	R R R R R R R R	Device is powered but Firmware not running. The Power LED will light up red at the beginning of the boot sequence of the Interface Card. ► Please contact Vector VX support if the Power LED stays red for more than 10 seconds. In this case a critical error has occurred.
	G G G G G G G G	Firmware loaded successfully.
	G ☐ G ☐ G ☐ G ☐	Response to ”Identify VX device command” (VXconfig).

System

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Firmware not loaded.
	R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Interface Card, its system components are initialized. If a component fails to initialize properly, the System LED stays red.

System

Symbol	Sequence	Description
		► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
	G G G G G G G G	Device is in normal operating mode.
	G ☐ G ☐ G ☐ G ☐	Firmware update in progress.

Ethernet

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Backplane link not established.
	R R R R R R R R	Backplane link error.
	G G G G G G G G	Backplane link established.
	G ☐ G ☐ G ☐ G ☐	Backplane link activity.

Channel

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Channel disabled.
	R R R R ☐ ☐ ☐ ☐	Channel enabled, invalid configuration. See VXconsole for details.
	R R R R R R R R	Channel enabled, no link established.
	G G G G G G G G	Channel enabled, link established, no data traffic.
	G ☐ G ☐ G ☐ G ☐	Channel enabled, link established, data traffic.

4.5.17.3 Boot and Link-up-Time

When the VX1161 is connected to power, the streaming card must prepare itself for a fast wake-up. For about t_{init} the power consumption will be P_{init}. After that time, the card enters standby mode and the power consumption is reduced to its minimum $P_{standby}$.

When the VX1161 is woken up (see section Connectors on page 176), the card is already prepared and can establish both ECU and sensor link within $t_{wakeLink}$. After wakeup, the card can provide data to the recorder after $t_{wakeReady}$.

The power consumption during normal operation consist of the maximum power of P_{run} drawn by the card as well as the power consumed by supplied cameras P_{POC}.

4.5.17.4 Changing the Mode of Operation

The VX1161.51C/.52C can be used in one of the following use cases:

- Video RX/TAP (video reception / logging)
- Video TX (video transmission / replay)
- µC Measurement/ Radar

In these modes, sensors or ECUs with GMSL2 transceivers can be directly attached to log or reply streams of data.

In µC Measurement/Radar use case, it is required to attach a Vector POD such as the VX1522 (see section VX1522.01 GMSL2 POD LVDS+DAP on page 486). This mode can be used to log radar data streams in parallel to ECU internal algorithm data and parameters. Contact Vector for further information.

The mode of operation can be changed in VXconfig on the Streaming Interface Card dialog.

4.5.17.5 Channel in TAP Mode

The TAP is designed to be as transparent as possible to the ECU.

If a channel group is configured in TAP mode, the incoming GMSL2 data stream is received by the MX9296A TAP deserializer. The data payload (e. g. video or radar frames) is then duplicated. One copy is forwarded to the MX9295A TAP serializer and one copy is forwarded to the recording data processor.

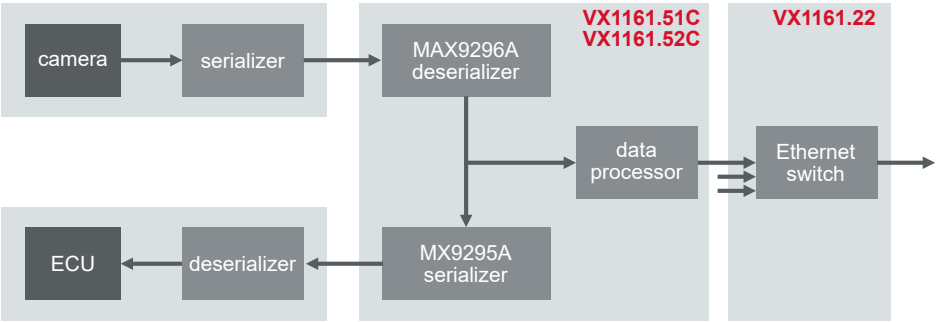


Figure 77: Channel in TAP mode; payload duplication

Control accesses from the ECU to the sensor via the I2C control channel or GPIOs are forwarded from the TAP serializer to the TAP deserializer.

TAP Initialization

The initialization sequence of the sensor and the ECU links follows this chart:

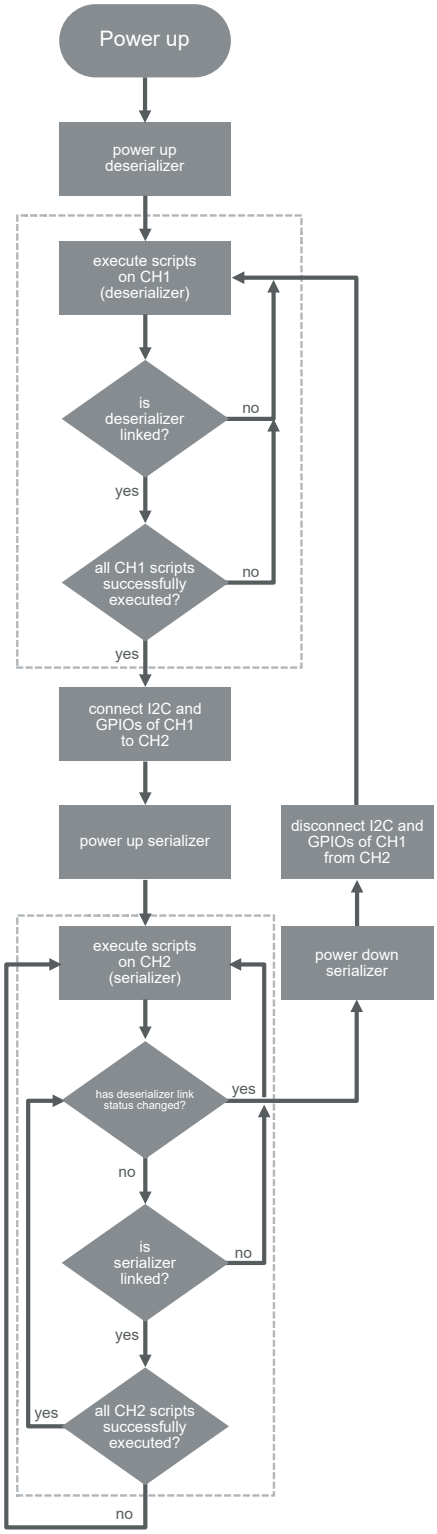


Figure 78: TAP initialization sequence

After power-up, the POC voltage is applied as configured and the deserializer is activated.

Next, the enabled scripts for the TAP RX channel (CH1/CH3) are immediately executed in an initial run (script 1, 2, 3) to configure the TAP deserializer, the sensor serializer and the sensor imager as early as possible.

After the initial script run on the TAP RX channel, the script execution engine waits for a TAP deserializer lock.

When the TAP deserializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the following conditions apply:

1. Script is not configured as "run once" (CASL SetConfig(7,1))
2. OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
3. OR script terminated with a return value ≠ 0 in previous run

If the TAP deserializer link unlocks, the script sequence repeats from the initial TAP deserializer run on, the I2C and GPIO bridges will be opened and the TAP serializer will be deactivated.

When the TAP deserializer locks, the I2C bus and GPIOs of the TAP serializer and TAP deserializer are connected. This allows the ECU to configure and control the sensor (see section GPIO Interconnect).

When the TAP deserializer locks, the TAP serializer is activated and the script execution on the TAP TX channel (CH2/CH4) are executed in an initial run (script 1, 2, 3).

After the initial script run on the TAP TX channel, the script execution engine waits for a TAP serializer lock.

When the TAP serializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the following conditions apply:

1. Script is not configured as "run once" (CASL SetConfig(7,1))
2. OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
3. OR script terminated with a return value ≠ 0 in previous run

If the TAP serializer link unlocks, the sequence repeats from the initial TAP serializer run on.

If the TAP deserializer link unlocks now, the I2C and GPIO bridges will be opened, the TAP serializer will be deactivated and the script sequence repeats from the initial TAP deserializer run on.

I2C Architecture

Without the TAP, the ECU is directly connected to the sensor head. Via the ECU deserializer and the sensor serializer it can configure and control all devices in that path. By inserting the TAP, the ECU deserializer is now connected to the TAP serializer and the sensor serializer is now connected to the TAP deserializer.

To allow the ECU controller to configure and control the sensor serializer and the sensor also in TAP mode, it might be necessary to configure the TAP serializer and the TAP deserializer via CASL scripts. Forwarding of pass through I2C channels must be configured, whereas the configuration I2C channel is enabled by default.

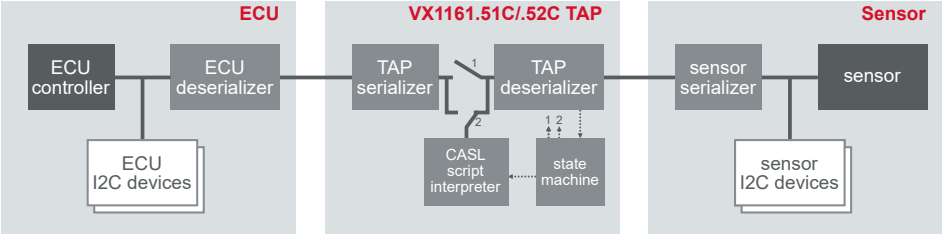


Figure 79: TAP I2C architecture; ECU configures ECU deserializer + TAP serializer + sensor

The I2C address of the TAP deserializer and the TAP serializer can be configured in VXconfig. Both chips start with their hardware strapped address and are then overwritten by the VX1161 firmware with the address configured in VXconfig. Additionally, the SetConfig(8,x) CASL command can be used to change the I2C address dynamically.

The sensor serializer I2C address is defined by its hardware straps. It can be changed via CASL scripts and must not be the same as the TAP serializer address.

The sensor I2C address is defined by the sensor itself. Accesses by the ECU will transparently be forwarded through the sensor link.

The additional TAP link will cause additional transmission delays (via clock stretching) to any sensor register access. These delays will reduce the effective control channel bandwidth and could affect system timings.

Modified Registers

Register Name	Register Address	Default Value	Modified Value	Description
Deserializer				
I2C_DEVICE_ID	0x0000	-	configured value	Device ID is set according configuration
RLMS58	0x1458	0x18	0x28	Value recommended by GMSL2 user guide V19
RLMS59	0x1459	0x58	0x68	Value recommended by GMSL2 user guide V19
RLMS4	0x1404	0x4B	0x29	Value recommended by GMSL2 user guide V19
RLMSA4	0x14A4	0xBD	0xC8	Value recommended by GMSL2 user guide V19
not documented	0x140A	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
not documented	0x140B	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
RLMS4	0x1504	0x4B	0x29	Value recommended by GMSL2 user guide V19
RLMSA4	0x15A4	0xBD	0xC8	Value recommended by GMSL2 user guide V19
not documented	0x150A	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
not documented	0x150B	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
CTRL0	0x0010	0x11	0x15	Bit 2 : REG_ENABLE set; enables the LDO; required by VX1161.51 hardware
CTRL2	0x0012	0x00	0x10	Bit 4: REG_MNL set; enables manual LDO control; required by VX1161.51 hardware
BACKTOP25	0x0320	0x3F	configured value	Bit 4:0 : PHY1_CSI_TX_DPLL_PREDEF_FREQ set to the CSI speed, selected in VXconfig; required to be compatible with the VX1161.51 hardware
MIPI_PHY0	0x0330	0x0x	0x8x	Bit 7: FORCE_CSI_OUT_EN set; enables continuous CSI clock; required by VX1161.51 hardware
Serializer				
I2C_DEVICE_ID	0x0000	-	configured value	Device ID is set according configuration
RLMS4	0x1404	0x4B	0x29	Value recommended by GMSL2 user guide V19
RLMSA4	0x14A4	0xBD	0xC8	Value recommended by GMSL2 user guide V19
not documented	0x140A	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
not documented	0x140B	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
REG5	0x0005	0x00	0x80	Bit 7 : LOCK_EN is set; enables the

Register Name	Register Address	Default Value	Modified Value	Description
				lock output; required by the VX1161.51 firmware to monitor the link status
CTRL0	0x0010	0x11	0x15	Bit 2: REG_ENAB is set; enables the regulator; required to be compatible with the VX1161.51 hardware; value recommended by GMSL2 user guide V19 Bit 6: RESET_LINK is set and cleared; used to hold back link after power-up while VX1161.51 FW is configuring system
CTRL2	0x0012	0x00	0x10	Bit 4: REG_MNL is set; enables manual regulator control; required to be compatible with the VX1161.51 hardware; value recommended by GMSL2 user guide V19
REG1	0x0001	0x08	0x08	Bit 4: DIS_REM_CC is set and cleared; used to re-establish link to serializer

Detection of TAP by ECU Firmware for Measurement Mode

In cases where the ECU needs to know whether a TAP is present, CASL scripts can be used to manipulate unused spare default register values in the CAM serializer. The ECU can then read the register and check whether it contains the default value (no TAP) or the CASL script value (TAP node).

MFP Interconnect

For forwarding MFPs (multi function pins) from the ECU to the sensor, the TAP deserializer and the sensor serializer registers must be configured properly. This can be done via CASL scripts. Static MFP settings can also be configured via CASL scripts.

Additionally, the VX1161.51C/.52C has a Sensor Trigger Generator which can generate synchronized trigger pulses on one MFP pin per channel (see section Sensor Trigger Generator on page 279).

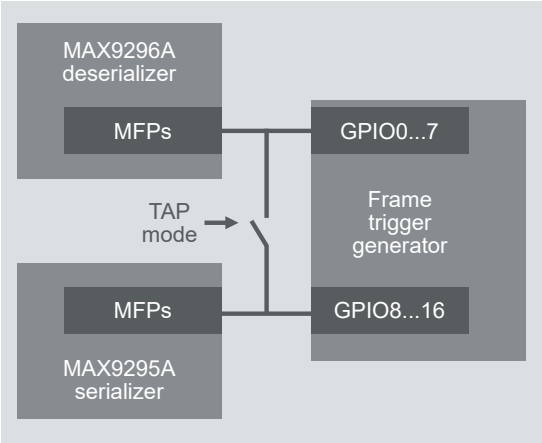


Figure 80: In TAP mode, some MFPs of serializer and deserializer are connected as long as the deserializer is GMSL locked to the remote serializer

Note
Care must be taken to avoid signal collisions between MFP outputs and Sensor Trigger Generator outputs.

Note
MFP1 of deserializer and MFP3 of serializer must stay configured as LOCK outputs. This function will be configured by the card firmware at boot time. If the function of these pins is changed, the TAP will no longer be able to detect unlocks and script execution might be disturbed.

Forwarding of analog MFP features is not supported.

On GMSL2, the transmission of MFPs can be configured either in standard or delay compensation mode. For frame trigger signals, the delay compensated mode should be used to reduce jitter. Both modes can be configured via CASL scripts. The transmission of MFP signals takes about $t_{delayMFP}$ per GMSL2 link. In TAP mode, the ECU link and the sensor link add both this delay.

In cases where sensor triggers are transferred via MFPs, it must be considered that the sensor links of two TAP links are not synchronized. Therefore, a skew between the two independent VX1161.51 TAP channels of $t_{delayMFP}$ must be assumed.

The MFPs of the TAP serializer are bidirectionally connected to the MFPs of the deserializer and the Sensor Trigger Generator GPIOs as follows:

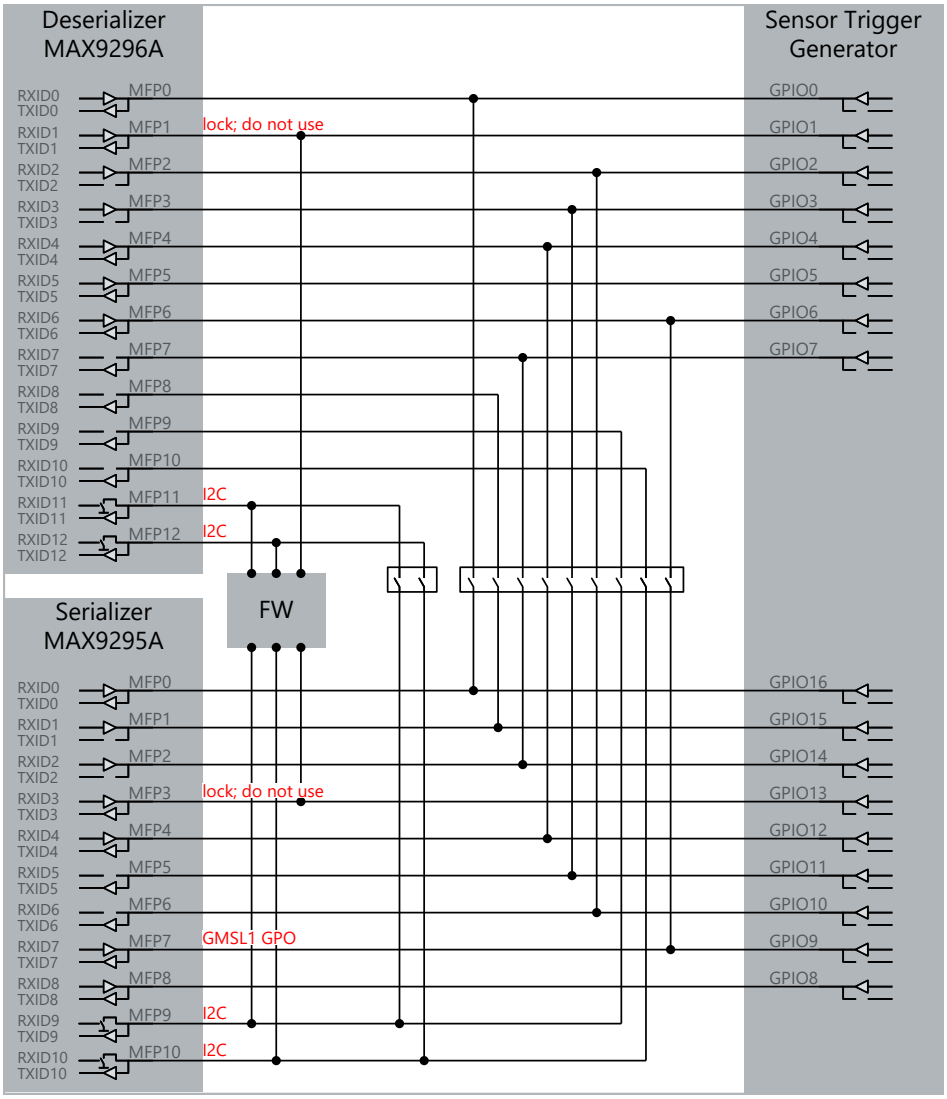


Figure 81: TAP MFP architecture; MFP connections between TAP deserializer, TAP serializer and Sensor Trigger Generator; switches are closed when deserializer link is stable

Signal of Serializer		Connected to Signal of Deserializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**	**Direction**	**Name**
CH2 - MFP0	IO	CH1 - MFP0	IO	CH2 - GPIO_16 = CH1 - GPIO_0
CH2 - MFP1	O	CH1 - MFP8	I	CH2 - GPIO_15
CH2 - MFP2	O	CH1 - MFP7	I	CH2 - GPIO_14 = CH1 - GPIO_7
CH2 - MFP3	IO; LOCK	-	-	(CH2 - GPIO_13; do not use)
CH2 - MFP4	IO	CH1 - MFP4	IO	CH2 - GPIO_12 = CH1 - GPIO_4
CH2 - MFP5	I	CH1 - MFP3	O	CH2 - GPIO_11 = CH1 - GPIO_3
CH2 - MFP6	I	CH1 - MFP2	O	CH2 - GPIO_10 = CH1 - GPIO_2
CH2 - MFP7	IO	-	-	CH2 - GPIO_9
CH2 - MFP8	IO	-	-	CH2 - GPIO_8
CH2 - MFP9	IO_OD	CH1 - MFP9	I	-
CH2 - MFP10	IO_OD	CH1 - MFP10	I	-
-	-	CH1 - MFP1	IO; LOCK	(CH1 - GPIO_1; do not use)

Signal of Serializer		Connected to Signal of Deserializer		Signal of Sensor Trigger Generator
Name	Direction	Name	Direction	Name
-	-	CH1 - MFP5	IO	CH1 - GPIO_5
-	-	CH1 - MFP6	IO	CH1 - GPIO_6
-	-	CH1 - MFP11	IO_DD	-
-	-	CH1 - MFP12	ID_DD	-
CH4 - MFP0	IO	CH3 - MFP0	IO	CH4 - GPIO_16 = CH3 - GPIO_0
CH4 - MFP1	O	CH3 - MFP8	I	CH4 - GPIO_15
CH4 - MFP2	O	CH3 - MFP7	I	CH4 - GPIO_14 = CH3 - GPIO_7
CH4 - MFP3	IO; LOCK	-	-	(CH4 - GPIO_13; do not use)
CH4 - MFP4	IO	CH3 - MFP4	IO	CH4 - GPIO_12 = CH1 - GPIO_4
CH4 - MFP5	I	CH3 - MFP3	O	CH4 - GPIO_11 = CH3 - GPIO_3
CH4 - MFP6	I	CH3 - MFP2	O	CH4 - GPIO_10 = CH3 - GPIO_2
CH4 - MFP7	IO	-	-	CH4 - GPIO_9
CH4 - MFP8	IO	-	-	CH4 - GPIO_8
CH4 - MFP9	IO_OD	CH3 - MFP9	I	-
CH4 - MFP10	IO_OD	CH3 - MFP10	I	-
	-	CH3 - MFP1	IO; LOCK	(CH3 - GPIO_1; do not use)
	-	CH3 - MFP5	IO	CH3 - GPIO_5
	-	CH3 - MFP6	IO	CH3 - GPIO_6
	-	CH3 - MFP11	IO_DD	-
		CH3 - MFP12	ID_DD	-

Signal mapping of TAP deserializer MFPs to trigger generator GPIOs; directions with view of deserializer (IO_OD: Input+open drain output); LOCK: Do not use. Signals required internally.

4.5.17.6 Channel in RX Mode

If a channel group is configured in RX mode, the incoming GMSL2 data stream is received by the MAX9296A deserializer and forwarded to the recording data processing. The serializers MAX9295A on channel 2/4 are powerd off.

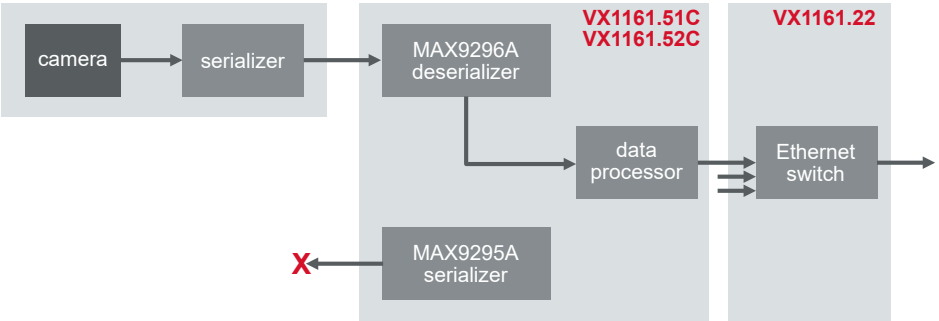


Figure 82: TAP channel in RX mode; payload reception

Initialization

The initialization sequence of the sensor link follows this chart:

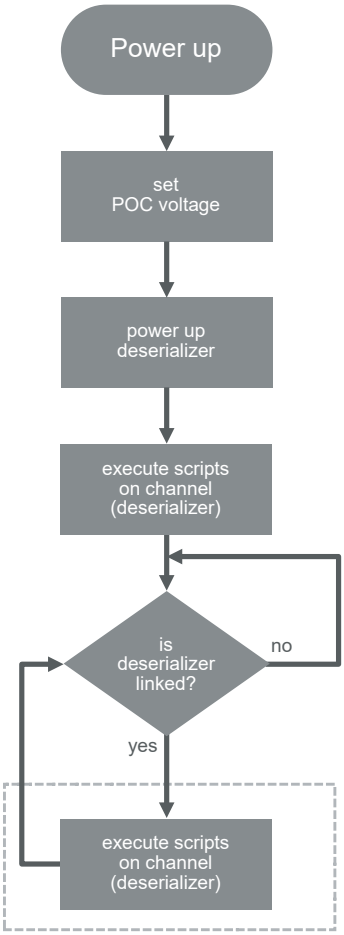


Figure 83: RX initialization sequence

After power-up, the POC voltage is applied as configured and the deserializer is activated.

Next, the enabled scripts for the RX channel are executed immediately (script 1, 2, 3) to configure the deserializer, sensor serializer and sensor as early as possible.

After the initial run, the script execution engine waits for a deserializer lock.

When the deserializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the follwing conditions apply:

- Script is not configured as "run once" (CASL SetConfig(7,1))
- OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
- OR script terminated with a return value ≠ 0 in previous run

If the deserializer link unlocks, the script sequence repeates from the inital deserializer run on.

I2C Architecture

In RX mode, the TAP deserializer, sensor serializer and sensor need to be configured via CASL scripts. A dynamic sensor reconfiguration is currently not supported.

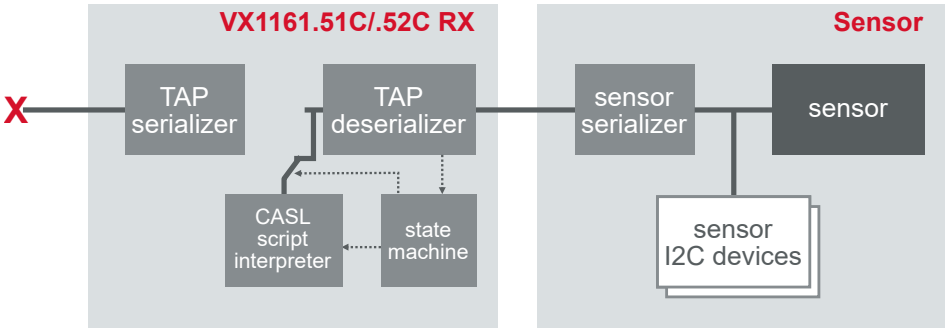


Figure 84: RX mode; I2C architecture; CASL scripts configure deserializer, serializer and sensor

The I2C address of the TAP deserializer can be configured in VXconfig. The chip starts with its hardware strapped address and is then overwritten by the VX1161 firmware with the address configured in VXconfig. Additionally, the SetConfig(8,x) CASL command can be used to change the I2C address dynamically.

The sensor serializer I2C address is defined by its hardware strapped pins and could be changed via CASL scripts.

The sensor I2C address is defined by the sensor itself. CASL scripts can be used to configure the sensor via I2C transactions. To enable I2C accesses to the sensor deserializer and serializer registers might have to be set properly in advance.

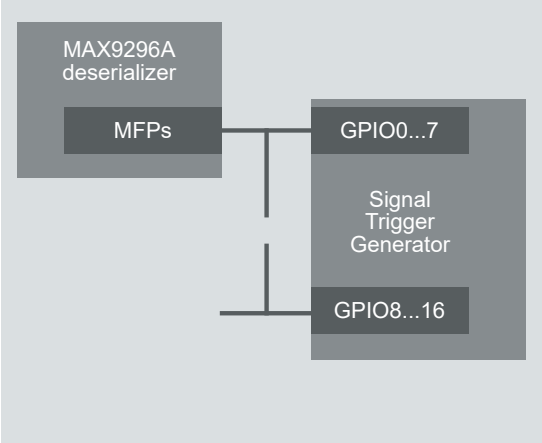
Modified Registers

Register Name	Register Address	Default Value	Modified Value	Description
Deserializer				
I2C_DEVICE_ID	0x0000	-	configured value	Device ID is set according configuration
RLMS58	0x1458	0x18	0x28	Value recommended by GMSL2 user guide V19
RLMS59	0x1459	0x58	0x68	Value recommended by GMSL2 user guide V19
RLMS4	0x1404	0x4B	0x29	Value recommended by GMSL2 user guide V19
RLMSA4	0x14A4	0xBD	0xC8	Value recommended by GMSL2 user guide V19
not documented	0x140A	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
not documented	0x140B	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
RLMS4	0x1504	0x4B	0x29	value recommended by GMSL2 user guide V19
RLMSA4	0x15A4	0xBD	0xC8	value recommended by GMSL2 user guide V19
not documented	0x150A	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
not documented	0x150B	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
CTRL0	0x0010	0x11	0x15	Bit 2 : REG_ENABLE set; enables the LDO; required by VX1161.51 hardware
CTRL2	0x0012	0x00	0x10	Bit 4: REG_MNL set; enables manual LDO control; required by VX1161.51 hardware
BACKTOP25	0x0320	0x3F	configured value	Bit 4:0 : PHY1_CSI_TX_DPLL_ PREDEF_FREQ set to the CSI speed, selected in VXconfig; required to be compatible with the VX1161.51 hardware
MIPI_PHY0	0x0330	0x0x	0x8x	Bit 7: FORCE_CSI_OUT_EN set; enables continuous CSI clock; required by VX1161.51 hardware

MFP Interconnect

The VX1161.51C/.52C has a Sensor Trigger Generator which can generate synchronized trigger pulses on one MFP pin per channel.

For forwarding MFPs (multi function pins) from the Sensor Trigger Generator to the sensor, the TAP deserializer and the sensor serializer registers must be configured properly. This can be done via CASL scripts. Static MFP settings can also be configured via CASL scripts.



Note
Care must be taken to avoid signal collisions between MFP outputs and Sensor Trigger Generator outputs.

Note
MFP1 of the deserializer must stay configured as LOCK outputs. This function will be configured by the card firmware at boot time. If the function of this pin is changed, the VX1161 firmware will no longer be able to detect unlocks and script execution might be disturbed.

On GMSL2, the transmission of MFPs can be configured either in standard or delay compensation mode. For frame trigger signals, the delay compensated mode should be used to reduce jitter. Both modes can be configured via CASL scripts. The transmission of MFP signals takes about $t_{delayMFP}$ for a GMSL2 link.

The MFPs of the TAP deserializer connected to the MFPs of the Sensor Trigger Generator GPIOs as follows:

Signal of Deserializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**
CH1 - MFP0	IO	CH1 - GPIO_0
CH1 - MFP8	I	-
CH1 - MFP7	I	CH1 - GPIO_7
CH1 - MFP4	IO	CH1 - GPIO_4
CH1 - MFP3	O	CH1 - GPIO_3
CH1 - MFP2	O	CH1 - GPIO_2
CH1 - MFP9	I	-
CH1 - MFP10	I	-

Signal of Deserializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**
CH1 - MFP1	IO; LOCK	(CH1 - GPIO_1; do not use)
CH1 - MFP5	IO	CH1 - GPIO_5
CH1 - MFP6	IO	CH1 - GPIO_6
CH1 - MFP11	IO_DD	-
CH1 - MFP12	IO_DD	-
CH3 - MFP0	IO	CH3 - GPIO_0
CH3 - MFP8	I	-
CH3 - MFP7	I	CH3 - GPIO_7
CH3 - MFP4	IO	CH3 - GPIO_4
CH3 - MFP3	O	CH3 - GPIO_3
CH3 - MFP2	O	CH3 - GPIO_2
CH3 - MFP9	I	-
CH3 - MFP10	I	-
CH3 - MFP1	IO; LOCK	(CH3 - GPIO_1; do not use)
CH3 - MFP5	IO	CH3 - GPIO_5
CH3 - MFP6	IO	CH3 - GPIO_6
CH3 - MFP11	IO_DD	-
CH3 - MFP12	IO_DD	-

Signal mapping of deserializer MFPs to trigger generator GPIOs; directions with view of deserializer (IO_OD: Input+open drain output); LOCK: Do not use. Signals required internally.

4.5.17.7 Channel in TX Mode

If a channel group is configured in TX mode, an incoming data stream from Ethernet is output via the MAX9295A serializer towards an ECU for example. The deserializer it kept powered-up for internal purposes, but cannot be used. Thus CH1 and CH3 are not available.

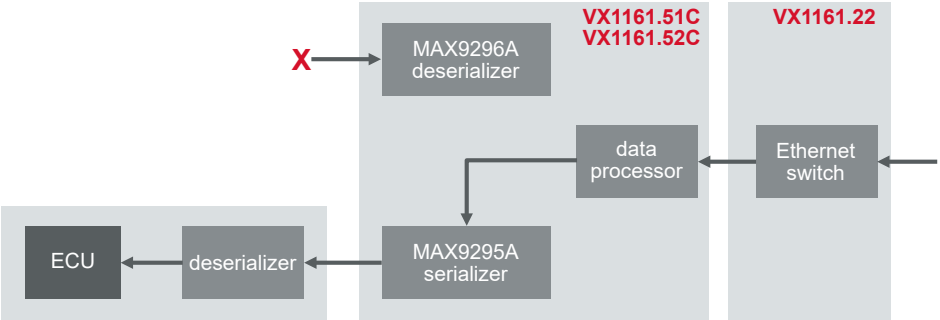


Figure 85: TAP channel in TX mode; payload transmission

Initialization

The initialization sequence of the sensor link follows this chart:

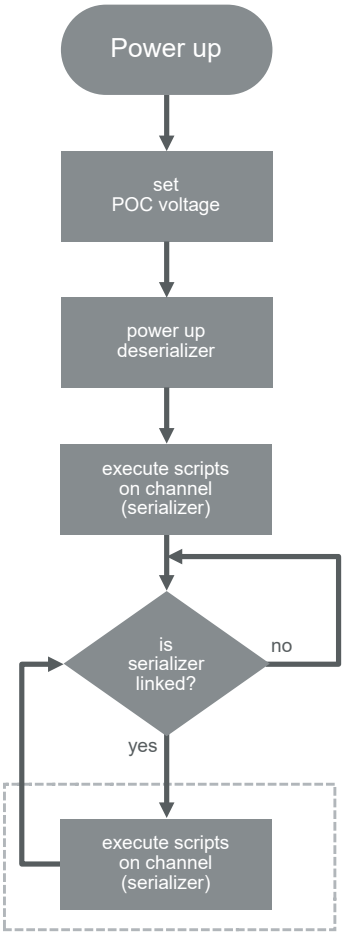


Figure 86: TX initialization sequence

After power-up, the serializer is immediately activated and the enabled scripts for the TX channel are immediately executed (script 1, 2, 3) to configure the serializer

and the ECU deserializer as early as possible.

After the initial run, the script execution engine waits for a serializer lock.

When the serializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the following conditions apply:

1. Script is not configured as "run once" (CASL SetConfig(7,1))
2. OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
3. OR script terminated with a return value ≠ 0 in previous run

If the serializer link unlocks, the sequence repeats from the initial serializer run on.

I2C Architecture

In TX mode, the serializer needs to be configured via CASL scripts. Since its configuration must match the hardware architecture, remote configuration accesses are disabled.

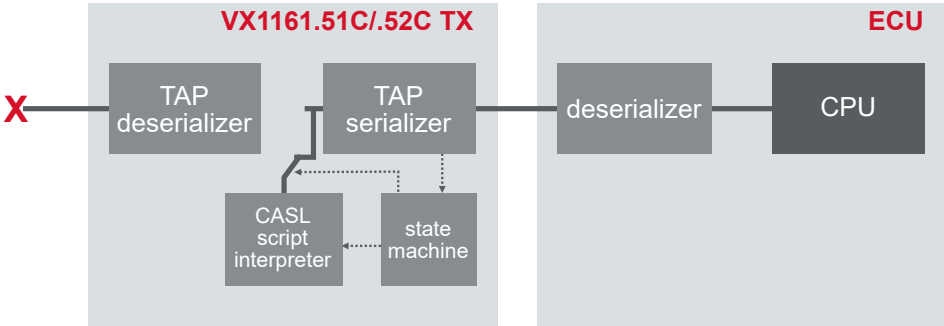


Figure 87: TX mode; I2C architecture; CASL scripts configure the serializer.

The I2C address of the TAP serializer can be configured in VXconfig. The chip starts with its hardware strapped address and is then overwritten by the VX1161 firmware with the address configured in VXconfig. Additionally, the SetConfig(8,x) CASL command can be used to change the I2C address dynamically.

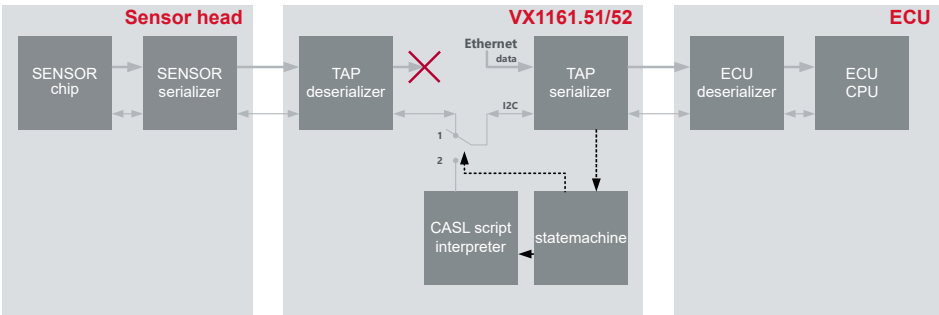


Figure 88: 1TX with sensor mode; ECU I2C accesses forwarded to sensor head; stream data for ECU supplied via Ethernet

The TX mode can be used to replace a sensor head which usually consists of the sensor serializer and the sensor chip. In typical applications both chips are configured by the ECU firmware via I2C. Since the VX1161.51 only provides the serializer and does not emulate any other I2C devices, the attached ECU firmware will fail accessing the sensor chip and enter an error state. To overcome this issue, either the ECU firmware must be enabled to cope with the missing sensor (code instrumentation, HIL mode) or the "1TX with sensor" mode in the Channel Group configuration can be used. The 1TX with sensor mode uses a real sensor head. I2C and GPIOs are forwarded to the sensor head like in TAP mode but data from the sensor is ignored and data from Ethernet is sent to the ECU.

Modified Registers

The following register values are modified by the VX1161 firmware during initialization:

Register Name	Register Address	Default Value	Modified Value	Description
Serializer				
I2C_DEVICE_ID	0x0000	-	configured value	Device ID is set according configuration
RLMS4	0x1404	0x4B	0x29	Value recommended by GMSL2 user guide V19
RLMSA4	0x14A4	0xBD	0xC8	Value recommended by GMSL2 user guide V19
not documented	0x140A	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
not documented	0x140B	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
REG5	0x0005	0x00	0x80	Bit 7 : LOCK_EN is set; enables the lock output; required by the VX1161.51 firmware to monitor the link status
CTRL0	0x0010	0x11	0x15	Bit 2: REG_ENAB is set; enables the regulator; required to be compatible with the VX1161.51 hardware; value recommended by GMSL2 user guide V19 Bit 6: RESET_LINK is set and cleared; used to hold back link after power-up while VX1161.51 FW is configuring system
CTRL2	0x0012	0x00	0x10	Bit 4: REG_MNL is set; enables manual regulator control; required to be compatible with the VX1161.51 hardware; value recommended by GMSL2 user guide V19
REG1	0x0001	0x08	0x08	Bit 4: DIS_REM_CC is set and cleared; used to re-establish link to serializer

GPIO Interconnect

The GPIOs of the serializer are connected to the GPIOs of the VX1161.51. The connections to the TAP deserializer are switched off.

For frame synchronization, one GPIO can be used as frame trigger input of the VX1161.51 to output a frame (see section Channel in TX Mode on page 273).

Additionally, the Sensor Trigger Generator can be used to toggle one serializer GPIO per channel. Because one GPIO per channel can be used to trigger the frame transmission, the generator can be used as frame rate generator to the Frame Trigger Detector (see section Channel in TX Mode on page 273).

The GPIOs of the serializer are connected to the GPIOs of the sensor trigger generator as follows:

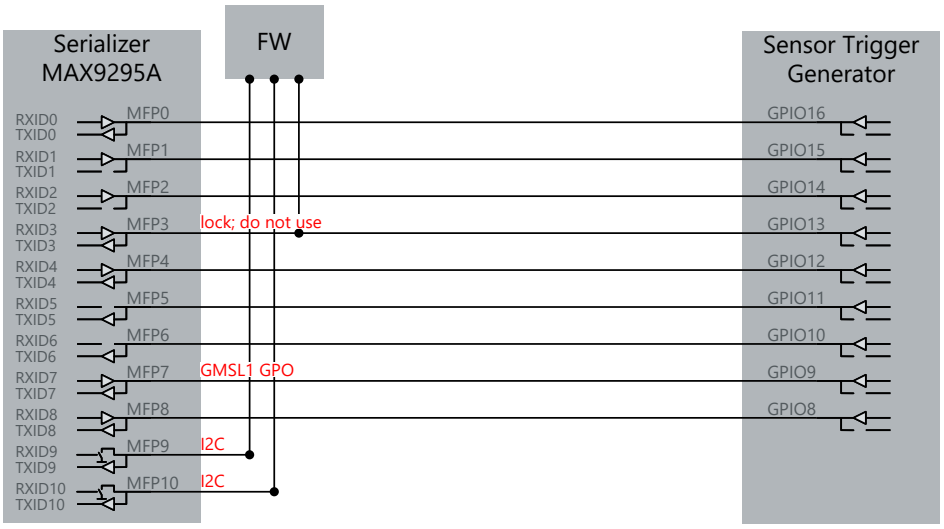


Figure 89: TX MFP architecture; GPIO/MFP connections between TX serializer and Sensor Trigger Generator

Signal of Serializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**
CH2 - MFP0	IO	CH2 - GPIO_16
CH2 - MFP1	O	CH2 - GPIO_15
CH2 - MFP2	O	CH2 - GPIO_14
CH2 - MFP3	IO; LOCK	(CH2 - GPIO_13; do not use)
CH2 - MFP4	IO	CH2 - GPIO_12
CH2 - MFP5	I	CH2 - GPIO_11
CH2 - MFP6	I	CH2 - GPIO_10
CH2 - MFP7	IO	CH2 - GPIO_9
CH2 - MFP8	IO	CH2 -GPIO_8
		-
CH4 - MFP0	IO	CH4 - GPIO_16
CH4 - MFP1	O	CH4 - GPIO_15
CH4 - MFP2	O	CH4 - GPIO_14
CH4 - MFP3	IO; LOCK	(CH4 - GPIO_13; do not use)
CH4 - MFP4	IO	CH4 - GPIO_1
CH4 - MFP5	I	CH4 - GPIO_11

Signal of Serializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**
CH4 - MFP6	I	CH4 - GPIO_10
CH4 - MFP7	IO	CH4 - GPIO_9
CH4 - MFP8	IO	CH4 - GPIO_8

4.5.17.8 Sensor Trigger Generator

The VX1161.51/52 provides a Sensor Trigger Generator for every channel. This generator can be used to generate a trigger signal to one of the GPIO/MFP pins. If the serializer and deserializer are configured correctly via CASL scripts, this trigger signal will be forwarded to the sensor and triggers the acquisition of data.

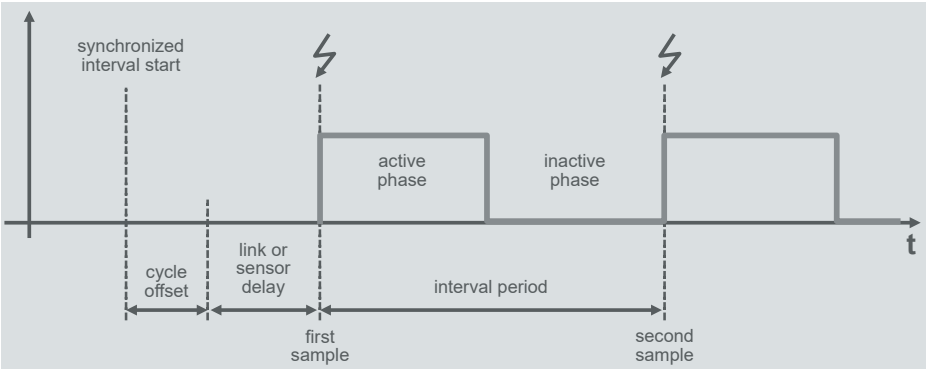


Figure 90: Sensor Trigger Generator with adjustable interval period

The generators of one module use one hardware timer as time base. This timer can be synchronized via PTP between multiple cards. If the PTP service is available, the trigger generation on all channels on all synchronized modules will be synchronous. If the PTP service is not available, only the channels on one module will be synchronous.

The current implementation requires that every VX1161.51/52 can exchange Ethernet broadcast messages with every other VX1161.51/52 during measurement start. These broadcast messages are used to negotiate the precise start of the first Frame Trigger period.

If a measurement setup consists of several VX1161 racks, there must be a path the broadcast messages can take to reach all other cards. Since Windows natively does not forward Ethernet broadcast messages from a receiving port to all other Ethernet ports, another connection is needed. To provide that path, an additional Ethernet cable between all VX1161 racks is required. The Ethernet switch inside the VX1161.22 uplink module will distribute those broadcast messages. Any Ethernet port on the VX1161.22 can be used due to the low bandwidth requirement.

The generators start outputting their trigger signal at start of measurement of the receiving application and it will stop generation at measurement stop. As a result, the sensor might not generate data when the measurement is stopped.
When the measurement is started, the time on all channels starts simultaneously and all generators start with their cycle offset times.

The trigger signal has two phases: the active and the inactive phase. Both can be configured in their duration. Together they define the trigger interval period, which defines the frequency of trigger pulses and the frame rate. Trigger generation always starts with the active phase. The Cycle offset parameter can be used to compensate delays in serial links or sensors. For sensors on fast paths, a delay can be added to trigger at the same time as sensors on slow paths.

For sensors requiring falling edges as trigger, the polarity of the output signal can be selected. If not inverted, the active phase starts with a rising edge which is intended to trigger the sensor.

Note
Care must be taken to avoid signal collisions between serializer or deserializer outputs and Sensor Trigger Generator outputs.

The Sensor Trigger Generator is configured via VXconfig and the following parameters:

Setting	Description
Enable	Activates the Sensor Trigger Generator. If disabled, the GPIOs are high impedance.
Cycle Offset	Defines a static delay from the ideal interval start in nano seconds. Can be used to compensate different delays in between several links or sensors.
Active Phase	Defines the duration of the active phase in nano seconds.
Inactive Phase	Defines the duration of the inactive phase in nano seconds.
GPIO output pin to trigger	GPIO pin of the VX1161.51/52, where the trigger signal shall be output. Check the GPIO/MFP section of your card to see which pins can be used and how they map to deserializer or serializer pins.
GPIO polarity active phase	Can be used to invert the output signal polarity. - checked (default): active phase starts with a rising edge. Active phase is high. Inactive phase is low. - unchecked: active phase starts with a falling edge. Active phase is low. Inactive phase is high.

4.5.17.9 Frame Trigger Detector

The VX1161.51/52 provides a Frame Trigger Detector for every channel in TX mode. This detector can be used to detect the change of one GPIO/MFP pin and trigger the transmission of the next video frame. This feature allows to synchronize frame transmissions to ECU trigger signals and with this to the ECU time domain.

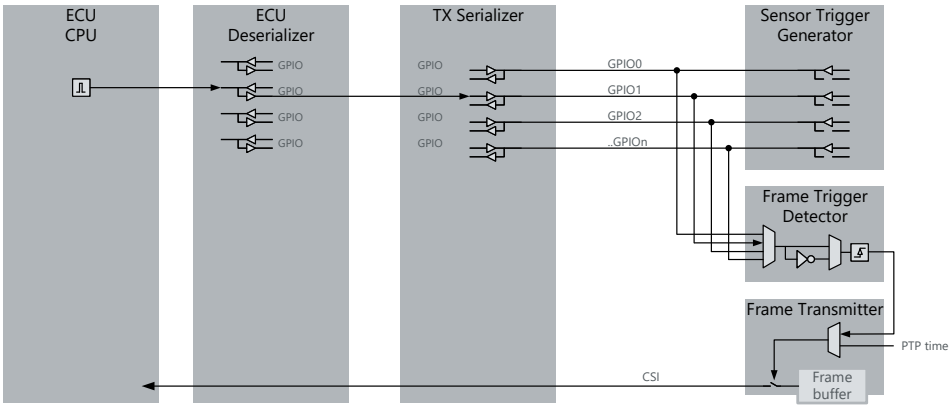


Figure 91: CPU generates frame trigger; Frame Trigger Detector detects edges and triggers the frame transmission

The detector can be fed by one GPIO/MFP of the channel group the channel belongs to. For selecting the active rising or falling edge, chose the polarity of the trigger signal.

Note
Care must be taken to avoid signal collisions between serializer outputs and Sensor Trigger Generator outputs.

The Frame Trigger Detector is configured via VXconfig and the following parameters in the Output Description section.

Setting	Description
GPIO input pin	GPIO pin of the VX1161.51/52, which shall feed the edge detector. Check the GPIO/MFP section of your card to see which pins can be used and how they map to your serializer pins.
GPIO polarity active phase	Specifies whether rising edges (checked) or falling edges (unchecked) on the given GPIO pin trigger the frame transmission.

4.5.17.10 Required Serializer and Deserializer Settings

The following requirements apply to the serializer and deserializer on the VX1161.51. The settings for the connected serializer at the sensor and deserializer in the ECU are allowed to differ according GMSL capabilities.

CSI Continuous Clock

Due to internal technical reasons, the CSI clocks on all channels must run in continuous CSI clock mode only. This mode is initially configured by the VX1161.51 firmware. Changing this setting will interrupt reception of data.

CSI Clock Lane Speed

Due to internal reasons, the CSI clock speed of the serializer must be configured via the Channel Group Mode and not be changed. The VX1161.51 firmware will configure this at start-up. If this setting is changed via CASL scripts, data reception and transmission will no longer be possible.

Link Lock Signals on MFP Pins

For detecting link status changes, the VX1161 firmware configures MFP1 in TAP deserializer MX9296A and MFP3 in TAP serializer MAX9295A to output their Lock signal. If the function of these pins is changed, the TAP will no longer be able to detect unlocks and script execution might be disturbed.

CSI Signal Settings

The CSI lanes of the TAP deserializer MAX9296A and TAP serializer MAX9295A do not require any lane swap, lane inversion or timing adjustments.

4.5.17.11 POC Supply

The VX1161.51C delivers power to connected sensor heads via the coax cable using Power-over-Coax (POC). It has an internal, isolated supply that is adjustable and independent of VX GND. It can source up to 2 W. The voltage can be switched off or adjusted from 5 V ... 12 V using VXconfig.

In TAP mode, the internal supply can be deactivated, allowing the POC supply of the ECU to be forwarded to the sensor head. This gives the ECU control over the sensor's power.

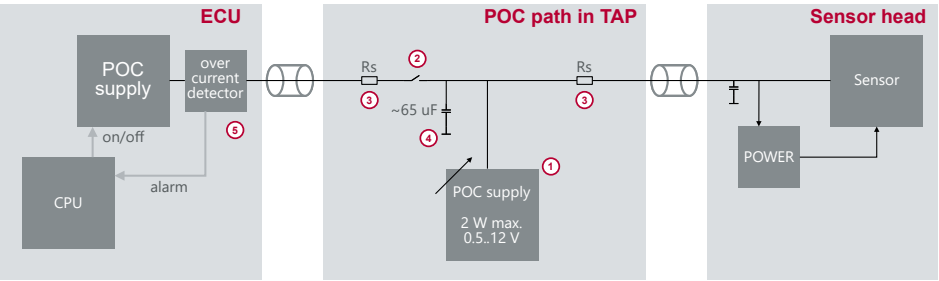


Figure 92: POC structure in VX1161.51C TAP channels

The internal series resistance (Rs) (3) of the POC bypass path is approximately 2 x 2.5 Ohms, and the internal capacitance (4) is approximately 65 µF. This capacitance can cause or increase inrush currents for the ECU that sources power and might trigger overcurrent detection features (5). In such cases, it might be an option to source the sensor from the VX (1) instead of the ECU, but then the ECU would have no control over the sensor power.

Furthermore, the internal VX POC supply can be output to the ECU for use cases in which CSI ingestion PODs require POC supply.

The POC voltage level, generation, and delivery direction are controlled via the POC voltage settings for the TAP_RX and TAP_TX ports in VXconfig as follows:

Note
Sourcing power in TAP mode is supported since VXtools v4.9.

Settings				
POC voltage V_{RX}on TAP RX channel	0V	5...12V	0V	5...12V
POC voltage V_{TX}on TAP TX channel	0V	0V	5...12V	5...12V
Effects				
POC bypass (2)	closed	open	closed	closed
VX POC source (1) voltage	off	V_{RX}	V_{TX}	Min(V_{RX},V_{TX})
Use case	ECU supplies sensor	VX supplies sensor	VX supplies ingest POD	VX supplies sensor and ingest POD

4.5.17.12 Technical Data

Weight	210 g
Power consumption	▶ P_{init}; initialization (~500ms): 500mW ▶ $P_{standby}$; sleep mode: <18 mW ▶ P_{run}; normal operation excl. POC: VX1161.51C:max. 8 W; VX1161.52C:max. 10W
POC (power over cable)	▶ V_{poc}; voltage sourced by VX1161.51: 0 V, 5...12 V configurable ▶ P_{poc}; maximum power delivered per RX channel; sourced by VX1161.51: 2 W ▶ $V_{PocTapMax}$; voltage forwarded from ECU by TAP: 3...18 V ▶ $I_{PocTapMax}$; max. current forwarded from ECU by TAP: 500 mA ▶ Electrical isolation between TAP_CH1/2 and TAP _CH3/4 and chassis max. 50 V ▶ Rpoc; DC resistance of POC forwarding path in TAP: typ. 2.616 Ohm
Minimum power -off time	▶ minimum time between powering-off and re-appying power: >2 s
Start-up time	▶ t_{init}; initialization after power-up: ~ 500 ms ▶ $t_{wakeLink}$; wake up after sleep; GMSL2 link stable: < 50 ms, depends on CASL script execution ▶ $t_{wakeReady}$; wake up after sleep; GMSL2 data to Ethernet: < 10 s
Temperature range	-40 °C ... +60 °C
Maximum payload bandwidths	▶ Card to host via Ethernet: 7.2 Gbps net; total for all channels together ▶ Every TAP/ RX channel GMSL2: 3.2 Gbps net, up to 4.8 Gbps
Latencies/delays	▶ $t_{delayMFP;}$; additional delay of MFPs; forward, backward delay compensated mode: ~6 µs +- 100ns Additional serializer and deserializer delay: ~ one CSI packet.
Connector	(Single signal) FAKRA Type Z; color: cyan; compatible to all other (single signal) FAKRA connector types; 6 GHz max; 25 mating cycles;
Maximum cable length	8 m, in both RX and TX direction at the same time
Part number	▶ VX1161.51C: 180154 ▶ VX1161.52C: 180164

4.5.18 VX1161.51D/.52D 4xRX GMSL2 MX9296

The VX1161.51D/.52.D 4xRX GMSL2 MX9296 card is a four-channel card for GMSL2 data streams. All channels contain one Analog Devices® MAX9296A deserializer for receiving data.

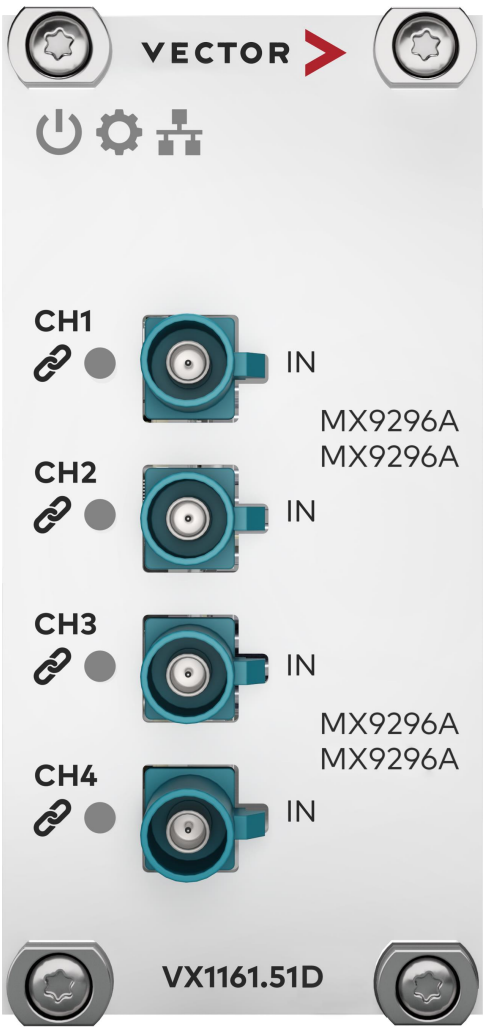


Figure 93: VX1161.51D/.52.D 4xRX GMSL2 MX9296

The VX1161.52 supports binary lossless compression of video streams on hardware (CH1 and CH3 only). This compression method reduces the required bandwidth for video transmission, logging and storage typically by 40 to 60 %, depending on video content. Every channel of the VX1161.52 can be compressed. The VX1161.52 compressor pipelines support RAW8/10/12/14/16 and YUV422 8bit pixel formats only. For other pixel formats, please contact Vector.

The card can be configured with VXconfig and updated to new software versions with VXupdate. Initialization of the transceivers and remote sensors can be done via CASL language scripts within VXconfig.

Note
VX1161.51D is supported by VXtools v4.3 and later.
VX1161.52D is supported by VXtools v4.6 and later.

Note
The full GMSL2 payload bandwidth of 4.8 Gbps net is supported by VXtools V4.6 and later.

Note
GMSL systems require after link lock several CSI packets to activate their internal pipelines. This causes the first packets to get lost inside the deserializer or serializer which will corrupt the first frame.

For frame precise measurements or replays, some dummy sensor data must be flushed through the entire sensor ECU connection in advance.

Note
The VX1161.52D only supports compression on CH1 and CH3.

4.5.18.1 Connectors

- **CH1...4**
  The VX1161.51D/.52.D provides FAKRA coaxial connectors of Type Z for connecting data sources such as video or radar sensors and ECUs. Type Z connectors are compatible to all other coding types of single signal FAKRA connectors. The connector properties can be found in section Technical Data on page 299.

4.5.18.2 LEDs

The status of the card is displayed by system LEDs in the lid top. The status of every channel is displayed by a channel LED next to every channel connector.

Power

Symbol	Sequence	Description
	R R R R R R R R	Device is powered but Firmware not running. The Power LED will light up red at the beginning of the boot sequence of the Interface Card. ► Please contact Vector VX support if the Power LED stays red for more than 10 seconds. In this case a critical error has occurred.
	G G G G G G G G	Firmware loaded successfully.
	G ☐ G ☐ G ☐ G ☐	Response to ”Identify VX device command” (VXconfig).

System

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Firmware not loaded.
	R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Interface Card, its system components are initialized. If a component fails to initialize properly, the System LED stays red.

System

Symbol	Sequence	Description
		► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
	G G G G G G G G	Device is in normal operating mode.
	G ☐ G ☐ G ☐ G ☐	Firmware update in progress.

Ethernet

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Backplane link not established.
	R R R R R R R R	Backplane link error.
	G G G G G G G G	Backplane link established.
	G ☐ G ☐ G ☐ G ☐	Backplane link activity.

Channel

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Channel disabled.
	R R R R ☐ ☐ ☐ ☐	Channel enabled, invalid configuration. See VXconsole for details.
	R R R R R R R R	Channel enabled, no link established.
	G G G G G G G G	Channel enabled, link established, no data traffic.
	G ☐ G ☐ G ☐ G ☐	Channel enabled, link established, data traffic.

4.5.18.3 Boot and Link-up-Time

When the VX1161 is connected to power, the streaming card must prepare itself for a fast wake-up. For about t_{init} the power consumption will be P_{init}. After that time, the card enters standby mode and the power consumption is reduced to its minimum $P_{standby}$.

When the VX1161 is woken up (see section Connectors on page 176), the card is already prepared and can establish both ECU and sensor link within $t_{wakeLink}$. After wakeup, the card can provide data to the recorder after $t_{wakeReady}$.

The power consumption during normal operation consist of the maximum power of P_{run} drawn by the card as well as the power consumed by supplied cameras P_{POC}.

4.5.18.4 Changing the Mode of Operation

The VX1161.51D/.52.D can be used in one of the following use cases:

- ► Video RX (video reception / logging)
- ► µC Measurement/ Radar

In Video RX use case, sensors or other data sources with GMSL2 transceivers can be directly attached to log streams of data.

In µC Measurement/Radar use case, it is required to attach a Vector POD such as the VX1522 (see section VX1522.01 GMSL2 POD LVDS+DAP on page 486). This mode can be used to log radar data streams in parallel to ECU internal algorithm data and parameters. Contact Vector for further information.

The mode of operation can be changed in VXconfig on the Streaming Interface Card dialog.

4.5.18.5 Channel in RX Mode

The VX1161.51D/.52D can be used in Video RX mode only.

In this mode, the incoming GMSL2 data stream is received by the MAX9296A deserializer and forwarded to the recording data processing.

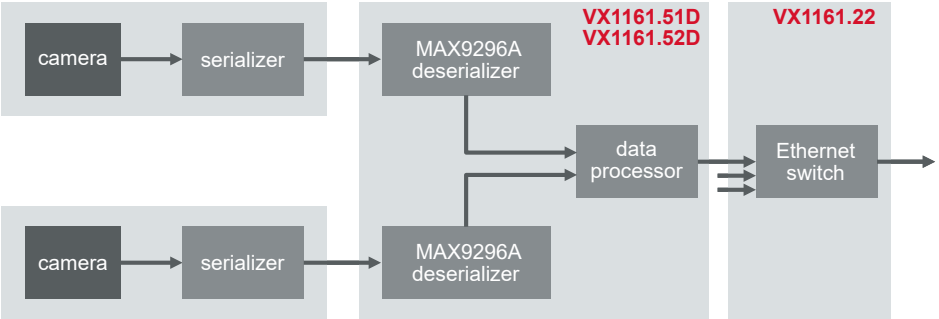


Figure 94: Channel in RX mode; payload reception

Initialization

The initialization sequence of the sensor link follows this chart:

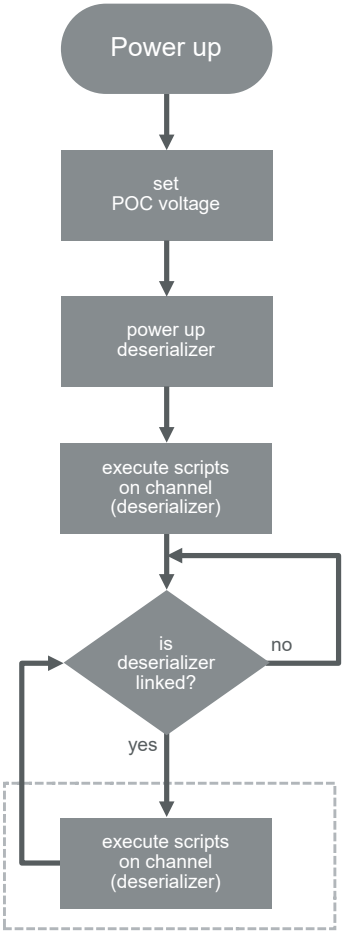


Figure 95: RX initialization sequence

After power-up, the POC voltage is applied as configured and the deserializer is activated.

Next, the enabled scripts for the RX channel are executed immediately (script 1, 2, 3) to configure the deserializer, sensor serializer and sensor as early as possible.

After the initial run, the script execution engine waits for a deserializer lock.

When the deserializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the follwing conditions apply:

- Script is not configured as "run once" (CASL SetConfig(7,1))
- OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
- OR script terminated with a return value ≠ 0 in previous run

If the deserializer link unlocks, the script sequence repeates from the inital deserializer run on.

I2C Architecture

In RX mode, the deserializer, sensor serializer and sensor need to be configured via CASL scripts. A dynamic sensor reconfiguration is currently not supported.

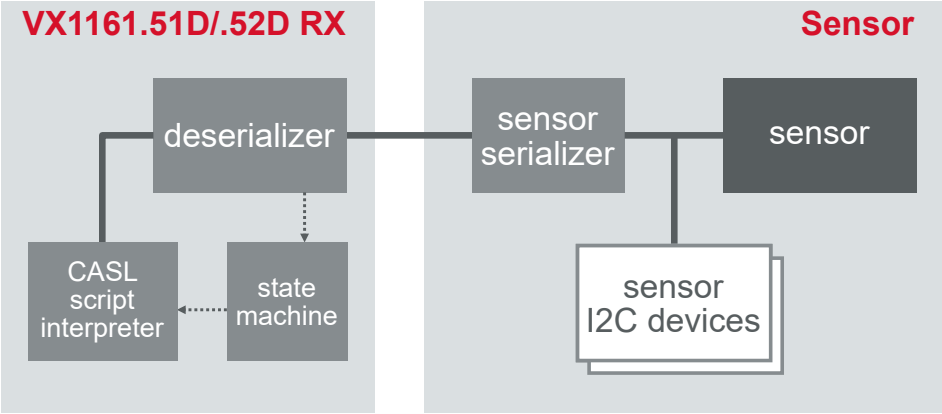


Figure 96: RX mode; I2C architecture; CASL scripts configure deserializer, serializer and sensor

The I2C address of the TAP deserializer can be configured in VXconfig. The chip starts with its hardware strapped address and is then overwritten by the VX1161 firmware with the address configured in VXconfig. Additionally, the SetConfig(8,x) CASL command can be used to change the I2C address dynamically.

The sensor serializer I2C address is defined by its hardware strapped pins and could be changed via CASL scripts.

The sensor I2C address is defined by the sensor itself. CASL scripts can be used to configure the sensor via I2C transactions. To enable I2C accesses to the sensor, deserializer and serializer registers might have to be set properly in advance.

Modified Registers

Register Name	Register Address	Default Value	Modified Value	Description
Deserializer				
I2C_DEVICE_ID	0x0000	-	configured value	Device ID is set according configuration
RLMS58	0x1458	0x18	0x28	Value recommended by GMSL2 user guide V19
RLMS59	0x1459	0x58	0x68	Value recommended by GMSL2 user guide V19
RLMS4	0x1404	0x4B	0x29	Value recommended by GMSL2 user guide V19
RLMSA4	0x14A4	0xBD	0xC8	Value recommended by GMSL2 user guide V19
not documented	0x140A	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
not documented	0x140B	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
RLMS4	0x1504	0x4B	0x29	Value recommended by GMSL2 user guide V19
RLMSA4	0x15A4	0xBD	0xC8	value recommended by GMSL2 user guide V19
not documented	0x150A	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
not documented	0x150B	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
CTRL0	0x0010	0x11	0x15	Bit 2 : REG_ENABLE set; enables the LDO; required by VX1161.51 hardware
CTRL2	0x0012	0x00	0x10	Bit 4: REG_MNL set; enables manual LDO control; required by VX1161.51 hardware
BACKTOP25	0x0320	0x3F	configured value	Bit 4:0 : PHY1_CSI_TX_DPLL_ PREDEF_FREQ set to the CSI speed, selected in VXconfig; required to be compatible with the VX1161.51 hardware
MIPI_PHY0	0x0330	0x0x	0x8x	Bit 7: FORCE_CSI_OUT_EN set; enables continuous CSI clock; required by VX1161.51 hardware

MFP Interconnect

The VX1161.51D/.52.D has a Sensor Trigger Generator which can generate synchronized trigger pulses on one MFP pin per channel.

For forwarding MFPs (multi function pins) from the Sensor Trigger Generator to the sensor, the TAP deserializer and the sensor serializer registers must be configured properly. This can be done via CASL scripts. Static MFP settings can also be configured via CASL scripts.

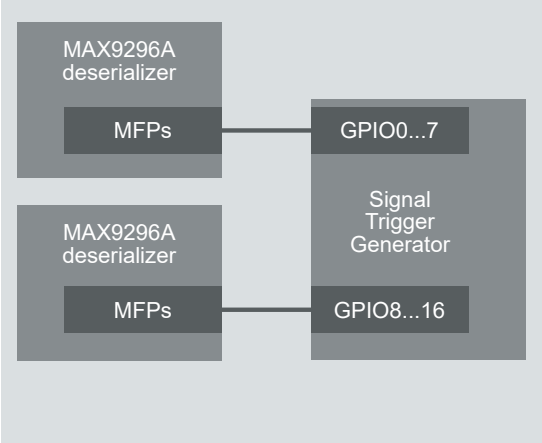


Figure 97: Channel in RX mode; payload reception

Note
Care must be taken to avoid signal collisions between MFP outputs and Sensor Trigger Generator outputs.

Note
MFP1 of deserializer must stay configured as LOCK outputs. This function will be configured by the card firmware at boot time. If the function of these pins is changed, the VX1161 firmware will no longer be able to detect unlocks and script execution might be disturbed.

On GMSL2, the transmission of MFPs can be configured either in standard or delay compensation mode. For frame trigger signals the delay compensated mode should be used to reduce jitter. Both modes can be configured via CASL scripts. The transmission of MFP signals takes about $t_{delayMFP}$ for a GMSL2 link.

The MFPs of the TAP deserializer connected to the MFPs of the Sensor Trigger Generator GPIOs as follows:

Signal of Deserializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**
CH1 - MFP0	IO	CH1 - GPIO_0
CH1 - MFP1	IO; LOCK	(CH1 - GPIO_1; do not use)
CH1 - MFP2	O	CH1 - GPIO_2
CH1 - MFP3	O	CH1 - GPIO_3
CH1 - MFP4	IO	CH1 - GPIO_4
CH1 - MFP5	IO	CH1 - GPIO_5
CH1 - MFP6	IO	CH1 - GPIO_6

Signal of Deserializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**
CH1 - MFP7	I	CH1 - GPIO_7
CH1 - MFP8	I	-
CH1 - MFP9	I	-
CH1 - MFP10	I	-
CH1 - MFP11	IO_DD	-
CH1 - MFP12	IO_DD	-
CH2 - MFP0	IO	CH2 - GPIO_8
CH2 - MFP1	IO; LOCK	(CH2 - GPIO_9; do not use)
CH2 - MFP2	O	CH2 - GPIO_10
CH2 - MFP3	O	CH2 - GPIO_11
CH2 - MFP4	IO	CH2 - GPIO_12
CH2 - MFP5	IO	CH2 - GPIO_13
CH2 - MFP6	IO	CH2 - GPIO_14
CH2 - MFP7	I	CH2 - GPIO_15
CH2 - MFP8	I	-
CH2 - MFP9	I	-
CH2 - MFP10	I	-
CH2 - MFP11	IO_DD	-
CH2 - MFP12	IO_DD	-
CH3 - MFP0	IO	CH3 - GPIO_0
CH3 - MFP1	IO; LOCK	(CH3 - GPIO_1; do not use)
CH3 - MFP2	O	CH3 - GPIO_2
CH3 - MFP3	O	CH3 - GPIO_3
CH3 - MFP4	IO	CH3 - GPIO_4
CH3 - MFP5	IO	CH3 - GPIO_5
CH3 - MFP6	IO	CH3 - GPIO_6
CH3 - MFP7	I	CH3 - GPIO_7
CH3 - MFP8	I	-
CH3 - MFP9	I	-
CH3 - MFP10	I	-
CH3 - MFP11	IO_DD	-
CH3 - MFP12	IO_DD	-
CH4 - MFP0	IO	CH4 - GPIO_8
CH4 - MFP1	IO; LOCK	(CH4 - GPIO_9; do not use)
CH4 - MFP2	O	CH4 - GPIO_10
CH4 - MFP3	O	CH4 - GPIO_11
CH4 - MFP4	IO	CH4 - GPIO_12
CH4 - MFP5	IO	CH4 - GPIO_13
CH4 - MFP6	IO	CH4 - GPIO_14
CH4 - MFP7	I	CH4 - GPIO_15
CH4 - MFP8	I	-
CH4 - MFP9	I	-
CH4 - MFP10	I	-
CH4 - MFP11	IO_DD	-
CH4 - MFP12	IO_DD	-

Signal mapping of TAP deserializer MFPs to trigger generator GPIOs; directions

with view of deserializer (IO_OD: Input+open drain output); LOCK: Do not use. Signals required internally.

4.5.18.6 Sensor Trigger Generator

The VX1161.51/52 provides a Sensor Trigger Generator for every channel. This generator can be used to generate a trigger signal to one of the GPIO/MFP pins. If the serializer and deserializer are configured correctly via CASL scripts, this trigger signal will be forwarded to the sensor and triggers the acquisition of data.

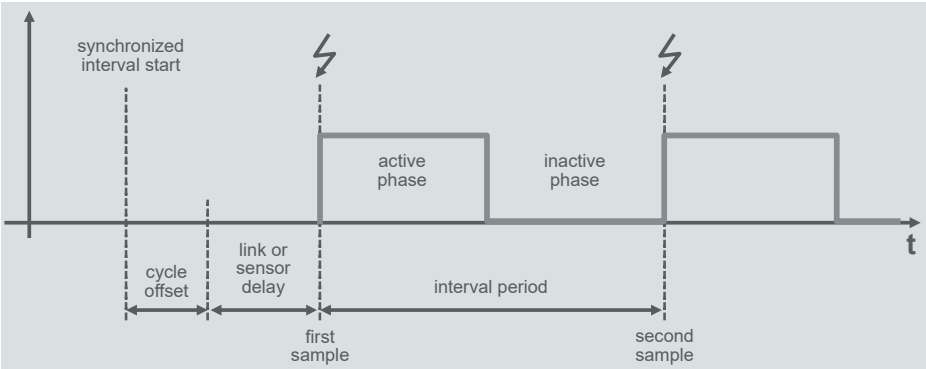


Figure 98: Sensor Trigger Generator with adjustable interval period

The generators of one module use one hardware timer as time base. This timer can be synchronized via PTP between multiple cards. If the PTP service is available, the trigger generation on all channels on all synchronized modules will be synchronous. If the PTP service is not available, only the channels on one module will be synchronous.

The current implementation requires that every VX1161.51/52 can exchange Ethernet broadcast messages with every other VX1161.51/52 during measurement start. These broadcast messages are used to negotiate the precise start of the first Frame Trigger period.

If a measurement setup consists of several VX1161 racks, there must be a path the broadcast messages can take to reach all other cards. Since Windows natively does not forward Ethernet broadcast messages from a receiving port to all other Ethernet ports, another connection is needed. To provide that path, an additional Ethernet cable between all VX1161 racks is required. The Ethernet switch inside the VX1161.22 uplink module will distribute those broadcast messages. Any Ethernet port on the VX1161.22 can be used due to the low bandwidth requirement.

The generators start outputting their trigger signal at start of measurement of the receiving application and it will stop generation at measurement stop. As a result, the sensor might not generate data when the measurement is stopped.
When the measurement is started, the time on all channels starts simultaneously and all generators start with their cycle offset times.

The trigger signal has two phases: the active and the inactive phase. Both can be configured in their duration. Together they define the trigger interval period, which defines the frequency of trigger pulses and the frame rate. Trigger generation always starts with the active phase. The Cycle offset parameter can be used to compensate delays in serial links or sensors. For sensors on fast paths, a delay can be added to trigger at the same time as sensors on slow paths.

For sensors requiring falling edges as trigger, the polarity of the output signal can be selected. If not inverted, the active phase starts with a rising edge which is intended to trigger the sensor.

Note
Care must be taken to avoid signal collisions between serializer or deserializer outputs and Sensor Trigger Generator outputs.

The Sensor Trigger Generator is configured via VXconfig and the following parameters:

Setting	Description
Enable	Activates the Sensor Trigger Generator. If disabled, the GPIOs are high impedance.
Cycle Offset	Defines a static delay from the ideal interval start in nano seconds. Can be used to compensate different delays in between several links or sensors.
Active Phase	Defines the duration of the active phase in nano seconds.
Inactive Phase	Defines the duration of the inactive phase in nano seconds.
GPIO output pin to trigger	GPIO pin of the VX1161.51/52, where the trigger signal shall be output. Check the GPIO/MFP section of your card to see which pins can be used and how they map to deserializer or serializer pins.
GPIO polarity active phase	Can be used to invert the output signal polarity. - checked (default): active phase starts with a rising edge. Active phase is high. Inactive phase is low. - unchecked: active phase starts with a falling edge. Active phase is low. Inactive phase is high.

4.5.18.7 Required Serializer and Deserializer Settings

The following requirements apply to the deserializer on the VX1161.51. The settings for the connected serializer at the sensor are allowed to differ according GMSL capabilities.

CSI Continuous Clock

Due to internal technical reasons, the CSI clocks on all channels must run in continuous CSI clock mode only. This mode is initially configured by the VX1161.51 firmware. Changing this setting will interrupt reception of data.

CSI Clock Lane Speed

Due to internal reasons, the CSI clock speed of the serializer must be configured via the Channel Group Mode and not be changed. The VX1161.51 firmware will configure this at start-up. If this setting is changed via CASL scripts, data reception and transmission will no longer be possible.

Link Lock Signals on MFP Pins

For detecting link status changes, the VX1161 firmware configures MFP1 of the deserializers MX9296A to output its Lock signal. If the function of those pins is changed, the VX1161.51 will no longer be able to detect unlocks and script execution might be disturbed.

CSI Signal Settings

The CSI lanes of the deserializer MAX9296A do not require any lane swap, lane inversion or timing adjustments.

4.5.18.8 POC Supply

The VX1161.51 can supply a sensor with power via Power-over-Coax (POC) of up to P_{POC}. The voltage can be adjusted to V_{poc} via VXconfig.

4.5.18.9 Technical Data

Weight	210 g
Power consumption	► P_{init}; initialization (~500ms): 500mW ► $P_{standby}$; sleep mode: <18 mW ► P_{run}; normal operation excl. POC: VX1161.51D:max. 8 W; VX1161.52D:10W
POC (power over cable)	► V_{poc}; voltage sourced by VX1161.51: 0 V, 5...12 V configurable ► P_{poc}; maximum power delivered per RX channel; sourced by VX1161.51/.52: 2 W ► Electrical isolation beween CH1, CH2, CH3 and CH4 max. 50 V ► Electrical isolation beween channels and chassis max. 50 V
Minimum power -off time	► minimum time between powering-off and re-appying power: >2 s
Start-up time	► t_{init}; initialization after power-up: ~ 500 ms ► $t_{wakeLink}$; wake up after sleep; GMSL2 link stable: < 50 ms, depends on CASL script execution ► $t_{wakeReady}$; wake up after sleep; GMSL2 data to Ethernet: < 10 s
Temperature range	-40 °C ... +60 °C
Maximum payload bandwidths	► Card to host via Ethernet: 7.2 Gbps net; total for all channels together ► Every channel GMSL2: 3.2 Gbps net, up to 5.2 Gbps net with later firmware
Latencies/delays	► $t_{delayMFP;}$; additional delay of MFPs; forward, backward delay compensated mode: ~6 µs +- 100ns Additional serializer and deserializer delay: ~ one CSI packet.
Connector	(Single signal) FAKRA Type Z; color: cyan; compatible to all other (single signal) FAKRA connector types; 6 GHz max; 25 mating cycles;
Maximum cable length	8 m, in both RX and TX direction at the same time
Part number	► VX1161.51D: 180155 ► VX1161.52D:180165

4.5.19 VX1161.52G 2xTAP FPDLINK4 TI9702/TI971

The VX1161.52G 2xTAP TI9702/TI971 card is a two-channel TAP card for FPDLINK4 data streams. TAP means that a receiver (deserializer) and a transmitter (serializer) are connected back-to-back and the received data is forwarded to the transmitter and duplicated towards the logging system. Both channels contain one Texas Instruments DS90UB9702 deserializer and one Texas Instruments DS90UB971 serializer each.

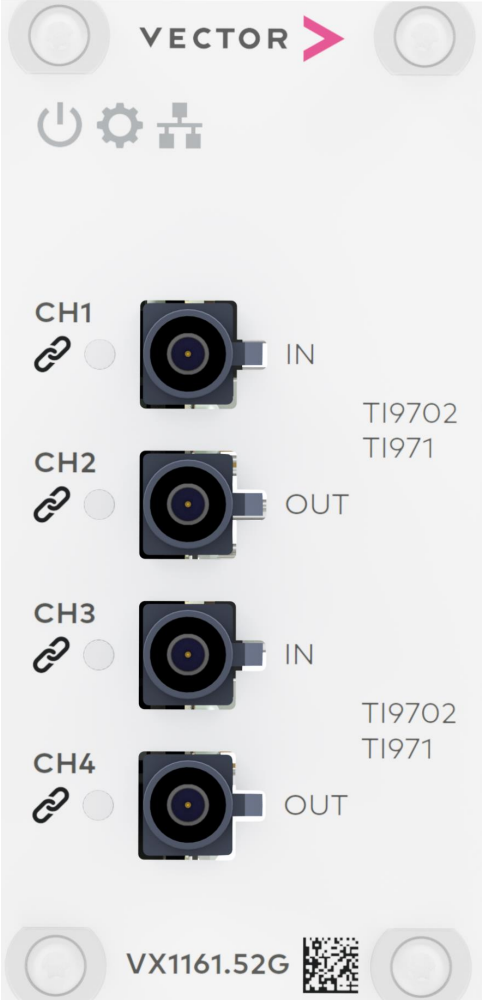


Figure 99: VX1161.52G 2xTAP TI9702/TI971

The operating mode of the card can be configured either in TAP, RX or TX mode. The VX1161.52G supports binary lossless compression (see section Lossless Compression on page 301).

The card can be configured with VXconfig and updated to new software versions with VXupdate. Initialization of the transceivers and remote sensors can be done via CASL language scripts within VXconfig.

Note
VX1161.52G is supported by VXtools v4.7 SP1 and later.

4.5.19.1 Lossless Compression

The video interface supports the hardware-based binary lossless compression of video streams. This compression method typically reduces the required bandwidth for video transmission, logging, and storage by 40 % ... 60 %, depending on the video content. Each channel of the interface can be compressed. The compression pipelines support the following pixel formats: RAW8, RAW10, RAW12, RAW16, and YUV422 8-bit. For other pixel formats, please contact Vector.

Four different compression modes are supported:

- **MobilEye on PC**
  This mode compresses the recording on the computer and requires a MobilEye license.

- **MobilEye on VX**
  This mode enables lossless compression on VX hardware and requires a MobilEye license. It reduces the bandwidth from the VX to the recording computer and offloads the necessary computing power.

- **Vector on VX**
  This mode enables lossless compression on the VX hardware and does not require a license. Only the payload of all received CSI long packets (all data types) is fed into the compression pipelines. The CSI headers and footers are stripped off beforehand. The embedded line payload is also fed into the compressor and reconstructed when decompressed. In this mode, all lines (embedded data, visible, and statistic lines) must be the same length as set in VXconfig. Enabling line length padding in the sensor may be necessary. To ensure good compression efficiency and enable later decoding, it is essential that all lines be the same length as the length set in VXconfig. The sensor may need to pad the embedded lines to the full length. Starting with VXtools V5.1, the interface enforces this.

- **Vector on VX V2**
  This mode has been supported since VXtools V5.1 and enables lossless compression on the VX hardware and does not require a license.

  If selected:

  - The CSI packets specified via CSI leading embedded packets and CSI trailing embedded packets of a frame are stored uncompressed.
  - The CSI headers and footers of all packets are also stored uncompressed.
  - The pixels are losslessly compressed on the interface hardware.
  - All three stored parts are secured via individual CRCs to detect data changes during transport or storage.

Each of the two compression pipelines is capable of compression:

- one stream with up to 350 MPx/s RAW8 ... RAW16
- one stream with up to 170 MPx/s YUV422

Vector provides decompression libraries and tools for Windows and Linux.

4.5.19.2 CRC Algorithm

```
std::vector<uint32_t> VCrc32::sCrcTable;

uint32_t VCrc32::Calculate(const uint8_t* rawData, const uint64_t rawDataS-
ize, const uint8_t bitsPerSample)
{
  uint32_t crc = 0xFFFFFFFF;
  const uint32_t specialWidth = 32 - bitsPerSample;
  const int andMask = (1 << bitsPerSample) - 1;
  uint32_t usedBytes = 0;
  const uint16_t* pData = reinterpret_cast<const uint16_t*>(rawData);

  while (usedBytes < rawDataSize)
  {
    /* XOR-in next input byte into MSB of crc and get this MSB, that's our
new intermediate dividend */
const uint16_t pos = static_cast<uint16_t>(((crc ^ (*pData << spe-
cialWidth)) >> specialWidth) & andMask);
    /* Shift out the MSB used for division per look up table and XOR with
the remainder */

    crc = (crc << bitsPerSample) ^ sCrcTable[pos];
    usedBytes += 2;
    pData++;

    if (rawDataSize % 2 != 0 && usedBytes == rawDataSize - 1)
    {
      uint16_t paddedValue = *pData;
      paddedValue = paddedValue & 0xFF;
      pData = &paddedValue;
    }
  }
  return crc;
}



void VCrc32::GenerateCrc32Table(const uint8_t bitsPerSample)
{
  if (!sCrcTable.empty())
  {
    return ;
  }

  const uint32_t specialWidth = 32 - bitsPerSample;
  const uint32_t elementCount = 1 << bitsPerSample;
  sCrcTable.resize(elementCount);

  for (uint32_t dividend = 0; dividend < elementCount; dividend++)
  {
    uint32_t currentByte = dividend << specialWidth;
    for (uint8_t bit = 0; bit < bitsPerSample; bit++)
    {
       if ((currentByte & 0x80000000) != 0)
      {
        constexpr uint32_t polynomial = 0x04C11DB7;
        currentByte <<= 1;
        currentByte ^= polynomial;
       }
       else
      {
         currentByte <<= 1;
       }
     }
    sCrcTable[dividend] = currentByte;
  }
}
```

4.5.19.3 Connectors

- **CH1...4**
  The VX1161.52G provides FAKRA coaxial connectors of Type Z for connecting data sources and data sinks such as video or radar sensors and ECUs. Type Z connectors are compatible to all other coding types of single signal FAKRA connectors. The connector properties can be found in section Technical Data on page 330.

4.5.19.4 LEDs

The status of the card is displayed by system LEDs in the lid top. The status of every channel is displayed by a channel LED next to every channel connector.

Power

Symbol	Sequence	Description
	R R R R R R R R	Device is powered but Firmware not running. The Power LED will light up red at the beginning of the boot sequence of the Interface Card. ► Please contact Vector VX support if the Power LED stays red for more than 10 seconds. In this case a critical error has occurred.
	G G G G G G G G	Firmware loaded successfully.
	G ☐ G ☐ G ☐ G ☐	Response to ”Identify VX device command” (VXconfig).

System

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Firmware not loaded.
	R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Interface Card, its system components are initialized. If a component fails to initialize properly, the System LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
	G G G G G G G G	Device is in normal operating mode.
	G ☐ G ☐ G ☐ G ☐	Firmware update in progress.

Ethernet

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Backplane link not established.
	R R R R R R R R	Backplane link error.
	G G G G G G G G	Backplane link established.
	G ☐ G ☐ G ☐ G ☐	Backplane link activity.

Channel

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Channel disabled.
	R R R R ☐ ☐ ☐ ☐	Channel enabled, invalid configuration. See VXconsole for details.
	R R R R R R R R	Channel enabled, no link established.
	G G G G G G G G	Channel enabled, link established, no data traffic.
	G ☐ G ☐ G ☐ G ☐	Channel enabled, link established, data traffic.

4.5.19.5 Boot and Link-up-Time

When the VX1161 is connected to power, the streaming card must prepare itself for a fast wake-up. For about t_{init} the power consumption will be P_{init}. After that time, the card enters standby mode and the power consumption is reduced to its minimum $P_{standby}$.

When the VX1161 is woken up (see section Connectors on page 176), the card is already prepared and can establish both ECU and sensor link within $t_{wakeLink}$. After wakeup, the card can provide data to the recorder after $t_{wakeReady}$.

The power consumption during normal operation consist of the maximum power of P_{run} drawn by the card as well as the power consumed by supplied cameras P_{POC}.

4.5.19.6 Changing the Mode of Operation

The VX1161.52G can be used in one of the following use cases:

- Video RX/TAP (video reception / logging)
- Video TX (video transmission / replay)

In these modes, sensors or ECUs with FPDLINK4 transceivers can be directly attached to log or replay streams of data.

The mode of operation can be changed in VXconfig on the Streaming Interface Card dialog.

4.5.19.7 Channel in TAP Mode

The TAP is designed to be as transparent as possible to the ECU.

If a channel group is configured forTAP mode , the incoming FPDLINK4 data stream is received by the DS90UB9702 TAP deserializer. The data payload (e. g. video or radar frames) is then duplicated. One copy is forwarded to the DS90UB971 TAP serializer and one copy is forwarded to the recording data processor.

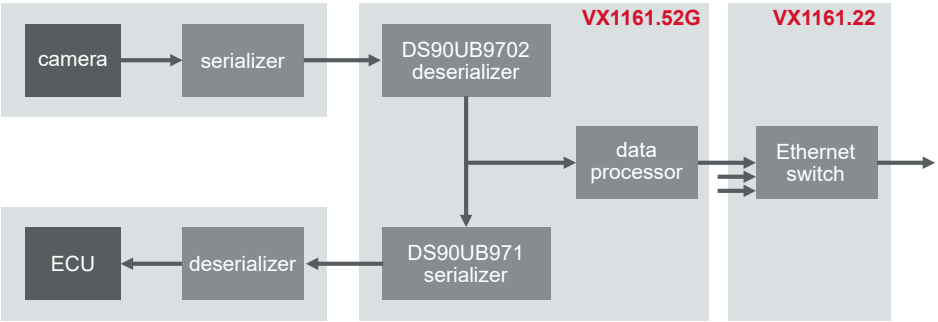


Figure 100: Channel in TAP mode; payload duplication

Control accesses from the ECU to the sensor via the I2C control channel or GPIOs are forwarded from the TAP serializer to the TAP deserializer (see section GPIO Interconnect on page 312).

TAP Initialization

The initialization sequence of the sensor and the ECU links follows this chart:

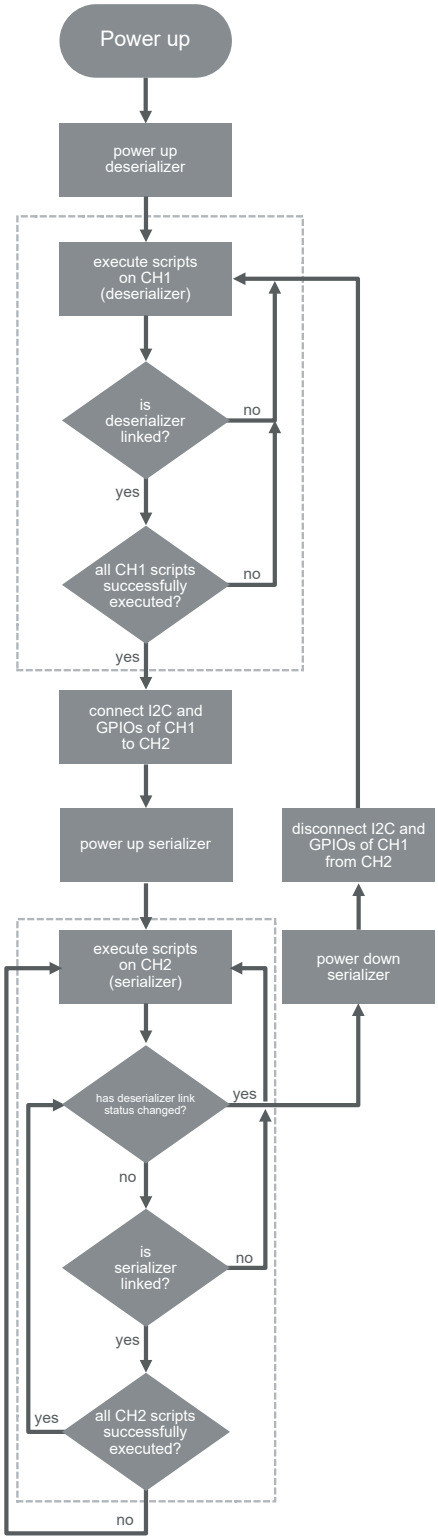


Figure 101: TAP initialization sequence; for details about dashed boxes see text below

After power-up, the POC voltage is applied as configured and the deserializer is activated.

Next, the enabled scripts for the TAP RX channel (CH1/CH3) are immediately executed in an initial run (script 1, 2, 3) to configure the TAP deserializer, the sensor serializer and the sensor imager as early as possible.

After the initial script run on the TAP RX channel, the script execution engine waits for a TAP deserializer lock.

When the TAP deserializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the following conditions apply:

1. Script is not configured as "run once" (CASL SetConfig(7,1))
2. OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
3. OR script terminated with a return value ≠ 0 in previous run

If the TAP deserializer link unlocks, the script sequence repeats from the initial TAP deserializer run on, the I2C and GPIO bridges will be opened and the TAP serializer will be deactivated.

When the TAP deserializer locks, the I2C bus and GPIOs of the TAP serializer and TAP deserializer are connected. This allows the ECU to configure and control the sensor (see section GPIO Interconnect).

When the TAP deserializer locks, the TAP serializer is activated and the script execution on the TAP TX channel (CH2/CH4) are executed in an initial run (script 1, 2, 3).

After the initial script run on the TAP TX channel, the script execution engine waits for a TAP serializer lock.

When the TAP serializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the following conditions apply:

1. Script is not configured as "run once" (CASL SetConfig(7,1))
2. OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
3. OR script terminated with a return value ≠ 0 in previous run

If the TAP serializer link unlocks, the sequence repeats from the initial TAP serializer run on.

If the TAP deserializer link unlocks now, the I2C and GPIO bridges will be opened, the TAP serializer will be deactivated and the script sequence repeats from the initial TAP deserializer run on.

I2C Architecture

Without the TAP, the ECU would configure its ECU deserializer, the sensor serializer and the sensor. The TAP structure induces a second set of TAP deserializer and TAP serializer which also needs to be configured via CASL scripts.

FPDLINK4 deserializers are automatically bound to their connected serializers and transfer the serializer I2C slave address as I2C aliasing destination address into the deserializer. This leads to the situation where the ECU would try to configure the TAP serializer as needed for the sensor. Since this configuration might violate the requirements of section Channel in TAP Mode on page 306, remote accesses to the TAP serializer are disabled via the LOCAL_WRITE_DISABLE bit in the TAP serializer. ECU write attempts to the TAP serializer will appear successful to the ECU but will not change any registers in the serializer. As a result, the configuration of the TAP serializer must also be done via CASL scripts.

The configuration of the sensor is left to the ECU which allows dynamic sensor reconfiguration and status polling.

Note
ECU write accesses to the TAP serializer are disabled via the LOCAL_WRITE_DISABLE bit in the TAP serializer. This must be considered regarding its impact to functional safety implementations.

Note
This behaviour differs from GMSL cards, where the ECU controller can still configure the sensor serializer and the sensor (see section Channel in TAP Mode on page 306).

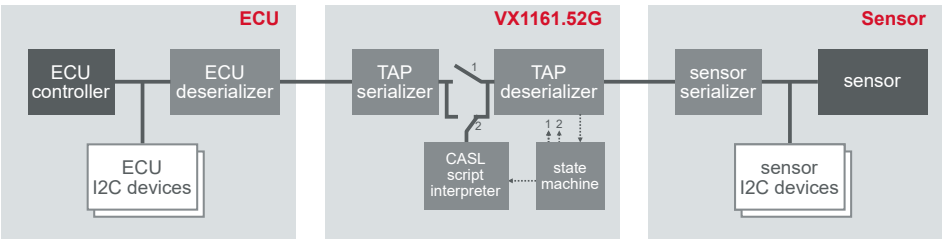


Figure 102: TAP I2C architecture; ECU configures ECU deserializer + sensor

The I2C address of the ECU deserializer does not change by inducing the TAP.

The I2C address of the TAP deserializer and the TAP serializer can be configured in VXconfig. Both chips start with their hardware strapped address and are then overwritten by the VX1161 firmware with the address configured in VXconfig. Additionally, the SetConfig(8,x) CASL command can be used to change the I2C address dynamically.

The sensor serializer I2C address is defined by its hardware straps. It could be changed via CASL scripts.

The sensor I2C address is defined by the sensor itself. To allow I2C accesses from the ECU to the sensor the I2C alias registers in the TAP deserializer need to be set via CASL scripts.

The additional TAP link will cause additional transmission delays (via clock stretch-

ing) to any sensor registers, which will reduce the effective control channel bandwidth and could affect system timings.

Modified Registers

The following register values are modified by the VX1161 firmware during initialization:

Register Name	Register Address	Default Value	Modified Value	Description
Deserializer				
I2C_DEVICE_ID	0x00	-	configured value	Device ID is set according configuration
SCL_HIGH_TIME	0x0A	0x7F	0x06	set I2C baudrate to 1Mbps
SCL_LOW_TIME	0x0C	0xF0	0x0B	set I2C baudrate to 1Mbps
CSI_PORT_SEL	0x32	0x00	0x02	Bit 1:TX_WRITE_PORT_1 is set: enable writes to TX port 1
BCC_CONFIG	0x58	0x1E	0x1E	Bit 2..0:set to 6; back channel runs at 50Mbps
FPD3_PORT_SEL	0x4C	0x00	0x02	Bit 1:RX_WRITE_PORT_1 is set and enables connected hardware port
CSI_PLL_CTL	0x1F	0x10	configured value	Bit 3..0 : CSI_TX_SPEED[3:0] set to the CSI speed, selected in VXconfig; required to be compatible with the VX1161.52 hardware
CSI_PLL_DIV	0xC9	0x32	configured value	Bit 5..0 : CSI_PLL_DIVIDER[5:0] set to the CSI speed, selected in VXconfig; required to be compatible with the VX1161.52 hardware
CSIPLL_REG_1	0x92	0x50	configured value	PLL_FB_DIV and PLL_OUT_DIV are set is set to the CSI speed, selected in VXconfig; required to be compatible with the VX1161.52 hardware
CSI0_TRIM_TCK_TRAIL	0x4B	0x00	0x2A	set to CSI timing required by VX1161.52 hardware
CSI0_TRIM_THS_TRAIL	0x4F	0x00	0x2A	set to CSI timing required by VX1161.52 hardware
and DS90UB971 compatibility values according DS90UB9701 datasheet				

Serializer				
I2C_DEVICE_ID	0x00	-	configured value	Device ID is set according configuration
SCL_HIGH_TIME	0x0B	0x7F	0x06	set I2C baudrate to 1Mbps
SCL_LOW_TIME	0x0C	0xF0	0x0B	set I2C baudrate to 1Mbps

Detection of TAP by ECU Firmware for Measurement Mode

In cases where the ECU needs to know whether a TAP is present, CASL scripts can be used to manipulate default register values in the TAP serializer (e. g. PGEN_LINE_SIZE1 (indirect address 0x04) in DS90UB971). Care must be taken about the risk of racing conditions between script execution and ECU register accesses.

GPIO Interconnect

For forwarding GPIOs from the ECU to the sensor, the TAP deserializer and the sensor serializer registers must be configured properly. This can be done via CASL scripts.

Forwarding of analog serializer features is not supported.

On FPDLINK4, GPIOs are sampled and transferred via control channel packets. Those packets are 30 bit in length and transmitted via the selected control channel baud rate of 52 Mbps, 50 Mbps, 47 Mbps or 9.5 Mbps. This results in an additional delay of $t_{delayGPIO}$.

In cases where camera sampling triggers are transferred via GPIOs, it must be considered that the control channels for two TAP links are not synchronized. Therefore, a skew between the two independent VX1161.52 TAP channels of twice $t_{delayGPIO}$ must be assumed.

The GPIOs of the TAP serializer are bidirectionally connected to the GPIOs of the deserializer and the Sensor Trigger Generator GPIOs as follows:

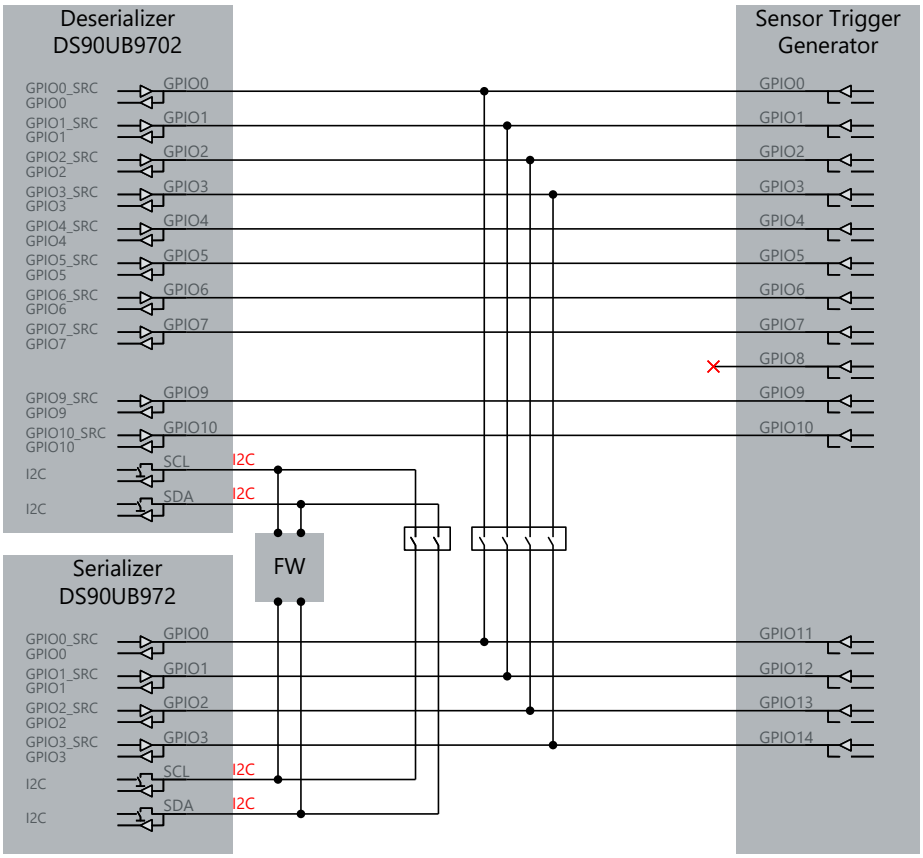


Figure 103: TAP GPIO architecture; GPIO connections between TAP deserializer, TAP serializer and Sensor Trigger Generator; switches are closed when TAP deserializer link is stable

Signal of Serializer		Connected to Signal of Deserializer		Signal of Sensor Trigger Generator
Name	**Dir**	**Name**	**Dir**	**Name**
CH2 - GPIO_0	IO	CH1 - GPIO_0	IO	CH1 - GPIO_0 = CH2 - GPIO_11
CH2 - GPIO_1	IO	CH1 - GPIO_1	IO	CH1 - GPIO_1 =

Signal of Serializer		Connected to Signal of Deserializer		Signal of Sensor Trigger Generator
Name	Dir	Name	Dir	Name
				CH2 - GPIO_12
CH2 - GPIO_2	IO	CH1 - GPIO_2	IO	CH1 - GPIO_2 = CH2 - GPIO_13
CH2 - GPIO_3	IO	CH1 - GPIO_3	IO	CH1 - GPIO_3= CH2 - GPIO_14
-	-	CH1 - GPIO_4	IO	CH1 - GPIO_4
-	-	CH1 - GPIO_5	IO	CH1 - GPIO_5
-	-	CH1 - GPIO_6	IO	CH1 - GPIO_6
-	-	CH1 - GPIO_7	IO	CH1 - GPIO_7
-	-	-	IO	CH1 - GPIO_8
-	-	CH1 - GPIO_9	IO	CH1 - GPIO_9
-	-	CH1 - GPIO_10	IO	CH1 - GPIO_10
CH4 - GPIO_0	IO	CH3 - GPIO_0	IO	CH3 - GPIO_0 = CH4 - GPIO_11
CH4 - GPIO_1	IO	CH3 - GPIO_1	IO	CH3 - GPIO_1 = CH4 - GPIO_12
CH4 - GPIO_2	IO	CH3 - GPIO_2	IO	CH3 - GPIO_2 = CH4 - GPIO_13
CH4 - GPIO_3	IO	CH3 - GPIO_3	-	CH3 - GPIO_3
-	-	CH3 - GPIO_4	IO	CH3 - GPIO_4
-	-	CH3 - GPIO_5	IO	CH3 - GPIO_5
-	-	CH3 - GPIO_6	IO	CH3 - GPIO_6
-	-	CH3 - GPIO_7	IO	CH3 - GPIO_7
-	-	-	IO	CH3 - GPIO_8
-	-	CH3 - GPIO_9	IO	CH3 - GPIO_9
-	-	CH 3 - GPIO_10	IO	CH3 - GPIO_10

4.5.19.8 Channel in RX Mode

If a channel group is configured in RX mode, the incoming FPDLINK4 data stream on channel 1/3 is received by the DS90UB9702 deserializer and forwarded to the recording data processing. The serializers DS90UB971 on channel 2/4 are powered-off.

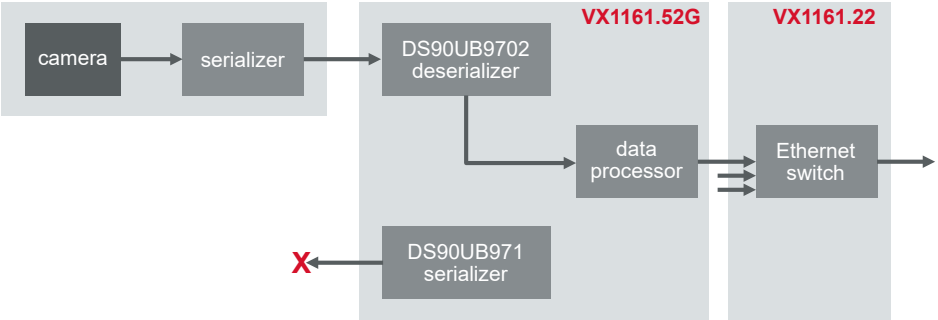


Figure 104: TAP channel in RX mode; payload reception

Initialization

The initialization sequence of the sensor link follows this chart:

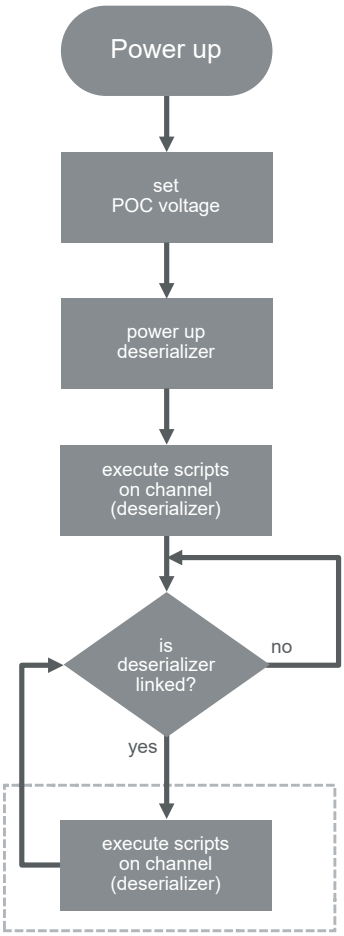


Figure 105: RX initialization sequence; for details about dashed box see text below

After power-up, the POC voltage is applied as configured and the deserializer is activated.

Next, the enabled scripts for the RX channel are executed immediately (script 1, 2, 3) to configure the deserializer, sensor serializer and sensor as early as possible.

After the initial run, the script execution engine waits for a deserializer lock.

When the deserializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the follwing conditions apply:

- Script is not configured as "run once" (CASL SetConfig(7,1))
- OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
- OR script terminated with a return value ≠ 0 in previous run

If the deserializer link unlocks, the script sequence repeates from the inital deserializer run on.

I2C Architecture

In RX mode, the VX1161.52 deserializer, sensor serializer and sensor need to be configured via CASL scripts. Dynamic sensor reconfiguration is currently only possible via CASL scripts.

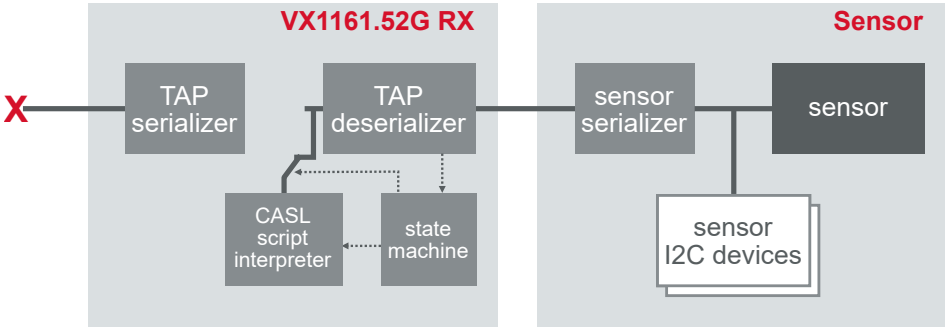


Figure 106: RX mode; I2C architecture; CASL scripts configure deserializer, serializer and sensor

The I2C address of the TAP deserializer can be configured in VXconfig. The chip starts with its hardware strapped address and is then overwritten by the VX1161 firmware with the address configured in VXconfig. Additionally, the SetConfig(8,x) CASL command can be used to change the I2C address dynamically.

The sensor serializer I2C address is defined by its hardware straps and could be changed via CASL scripts. The deserializer will bind automatically if configured properly. Therefore, no I2C address adjustments should be required.

The sensor I2C address is defined by the sensor itself. CASL scripts can be used to configure the sensor via I2C control channel accesses. To enable I2C control channel accesses via CASL scripts, the I2C alias registers in the deserializer must be set properly.

The VX1161.52 does not emulate any I2C devices (besides the TAP serializer). In order to avoid error states in the attached ECU, the ECU firmware must be modified to omit or ignore erroneous I2C accesses.

Modified Registers

The following register values are modified by the VX1161 firmware during initialization:

Register Name	Register Address	Default Value	Modified Value	Description
Deserializer				
I2C_DEVICE_ID	0x00	-	configured value	Device ID is set according configuration
SCL_HIGH_TIME	0x0A	0x7F	0x06	set I2C baudrate to 1Mbps
SCL_LOW_TIME	0x0C	0xF0	0x0B	set I2C baudrate to 1Mbps
CSI_PORT_SEL	0x32	0x00	0x02	Bit 1:TX_WRITE_PORT_1 is set: enable writes to TX port 1
BCC_CONFIG	0x58	0x1E	0x1E	Bit 2..0:set to 6; back channel runs at 50Mbps
FPD3_PORT_SEL	0x4C	0x00	0x02	Bit 1:RX_WRITE_PORT_1 is set and enables connected hardware port
CSI_PLL_CTL	0x1F	0x10	configured value	Bit 3..0 : CSI_TX_SPEED[3:0] set to the CSI speed, selected in VXconfig; required to be compatible with the VX1161.52 hardware
CSI_PLL_DIV	0xC9	0x32	configured value	Bit 5..0 : CSI_PLL_DIVIDER[5:0] set to the CSI speed, selected in VXconfig; required to be compatible with the VX1161.52 hardware
CSIPLL_REG_1	0x92	0x50	configured value	PLL_FB_DIV and PLL_OUT_DIV are set is set to the CSI speed, selected in VXconfig; required to be compatible with the VX1161.52 hardware
CSI0_TRIM_TCK_TRAIL	0x4B	0x00	0x2A	set to CSI timing required by VX1161.52 hardware
CSI0_TRIM_THS_TRAIL	0x4F	0x00	0x2A	set to CSI timing required by VX1161.52 hardware
and DS90UB971 compatibility values according DS90UB9701 datasheet				

GPIO Interconnect

The Sensor Trigger Generator can be used to toggle GPIOs of the deserializer. These signals can be forwarded to the connected serializer at the sensor and to the sensor to trigger sampling a frame simultaneously on several channels. To use this function, the deserializer, the sensor serializer and the sensor registers must be configured properly. This can be done via CASL scripts.

On FPDLINK4, GPIOs are sampled and transferred via control channel packets. Those packets are 30 bit in length and transmitted via the selected control channel baud rate of 52 Mbps, 50 Mbps, 47 Mbps or 9.5 Mbps. This results in an additional delay of $t_{delayGPIO}$.

In cases where sensor triggers are transferred via GPIOs it must be considered that the control channels for two links are not synchronized. Therefore, a skew between the two independent VX1161.52 channels of twice $t_{delayGPIO}$ must be assumed.

The GPIOs of the deserializer are connected to the GPIOs of the sensor trigger generator as follows:

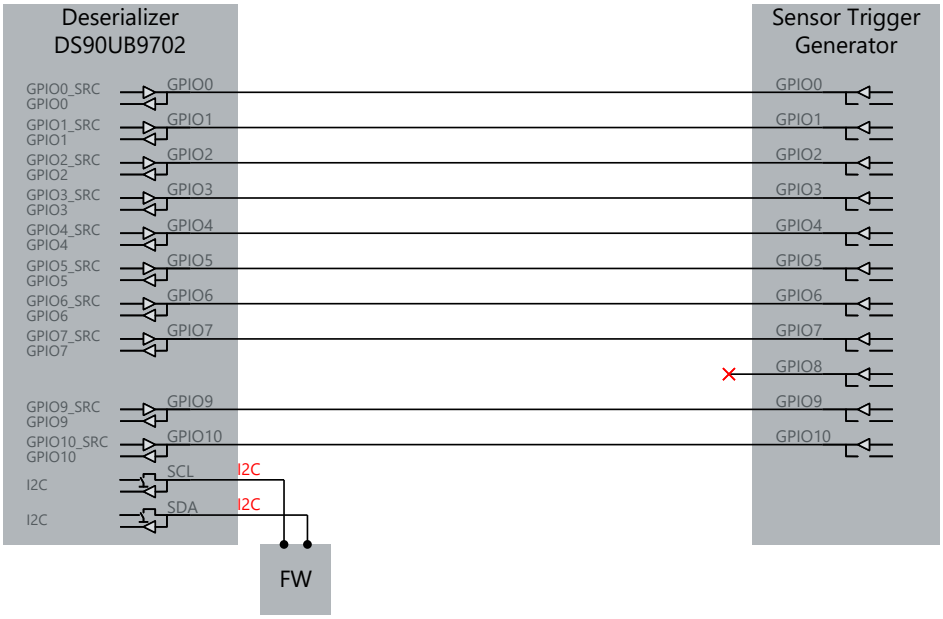


Figure 107: RX GPIO architecture; GPIO connections between RX deserializer and Sensor Trigger Generator

Signal of Deserializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**
CH1- GPIO_0	IO	CH1 - GPIO_0
CH1- GPIO_1	IO	CH1 - GPIO_1
CH1- GPIO_2	IO	CH1 - GPIO_2
CH1- GPIO_3	IO	CH1 - GPIO_3
CH1- GPIO_4	IO	CH1 - GPIO_4
CH1- GPIO_5	IO	CH1 - GPIO_5
CH1- GPIO_6	IO	CH1 - GPIO_6
CH1- GPIO_7	IO	CH1 - GPIO_7
-	IO	CH1 - GPIO_8

Signal of Deserializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**
CH1- GPIO_9	IO	CH1 - GPIO_9
CH1- GPIO_10	IO	CH1 - GPIO_10
CH3 - GPIO_0	IO	CH3 - GPIO_0
CH3 - GPIO_1	IO	CH3 - GPIO_1
CH3 - GPIO_2	IO	CH3 - GPIO_2
CH3 - GPIO_3	IO	CH3 - GPIO_3
CH3 - GPIO_4	IO	CH3 - GPIO_4
CH3 - GPIO_5	IO	CH3 - GPIO_5
CH3 - GPIO_6	IO	CH3 - GPIO_6
CH3 - GPIO_7	IO	CH3 - GPIO_7
-	IO	CH3 - GPIO_8
CH3 - GPIO_9	IO	CH3 - GPIO_9
CH3 - GPIO_10	IO	CH3 - GPIO_10

4.5.19.9 Channel in TX Mode

If a channel group is configured in TX mode, an incoming data stream from Ethernet is output via the DS90UB971 serializer towards an ECU for example. The deserializers on channel 1/3 are disabled.

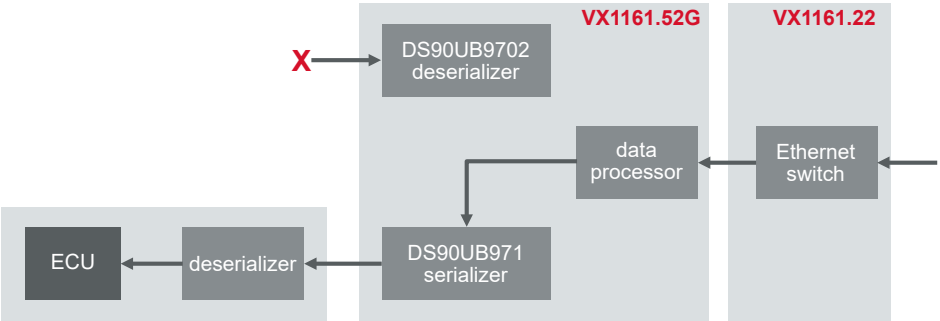


Figure 108: TAP channel in TX mode; payload transmission

Initialization

The initialization sequence of the sensor link follows this chart:

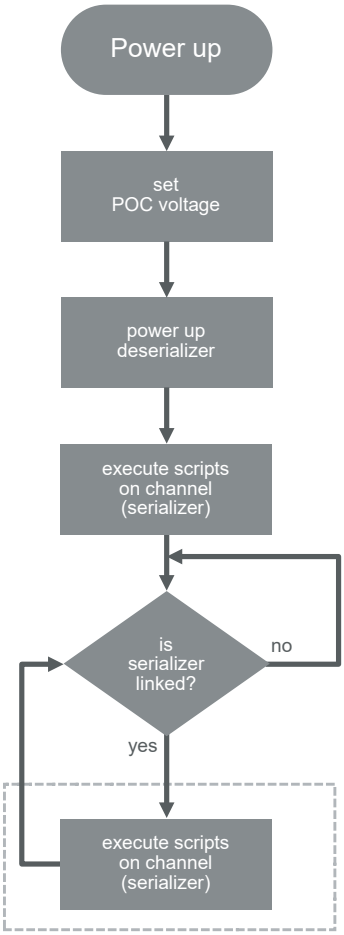


Figure 109: TX initialization sequence; for details about dashed box see text below

After power-up, the serializer is immediately activated and the enabled scripts for the TX channel are immediately executed (script 1, 2, 3) to configure the serializer

and the ECU deserializer as early as possible.

After the initial run, the script execution engine waits for a serializer lock.

When the serializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the following conditions apply:

1. Script is not configured as "run once" (CASL SetConfig(7,1))
2. OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
3. OR script terminated with a return value ≠ 0 in previous run

If the serializer link unlocks, the sequence repeats from the initial serializer run on.

I2C Architecture

In TX mode, the serializer needs to be configured via CASL scripts. Since its configuration must match the hardware architecture, remote configuration accesses are disabled.

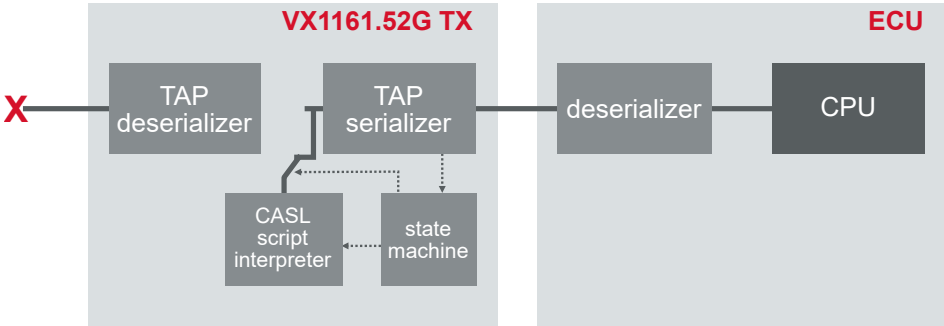


Figure 110: TX mode; I2C architecture; CASL scripts configure the serializer.

The I2C address of the TAP serializer can be configured in VXconfig. The chip starts with its hardware strapped address and is then overwritten by the VX1161 firmware with the address configured in VXconfig. Additionally, the SetConfig(8,x) CASL command can be used to change the I2C address dynamically.

Since the ECU deserializer will automatically bind to the TAP serializer, no address adjustments in the ECU firmware are required.

FPDLINK4 deserializers are automatically bound to their connected serializers and transfer the serializer I2C slave address as I2C aliasing destination address into the deserializer. This leads to the situation where the ECU would try to configure the TAP serializer as needed for the sensor. Since this configuration might violate the requirements of section Channel in TX Mode on page 320, remote accesses to the TAP serializer are disabled via the LOCAL_WRITE_DISABLE bit in the TAP serializer. ECU write attempts to the TAP serializer will appear successful to the ECU but will not change any registers in the serializer. As a result, the configuration of the TAP serializer must also be done via CASL scripts.

The VX1161.52 does not emulate any I2C devices (besides the serializer). In order to avoid error states in the attached ECU, either the Channel Group configuration 1TX with sensor can be used or the ECU firmware must be modified to omit or ignore erroneous I2C accesses. The 1TX with sensor mode acts logically like the TAP mode but video data from the sensor is ignored and data from Ethernet is sent to the ECU .

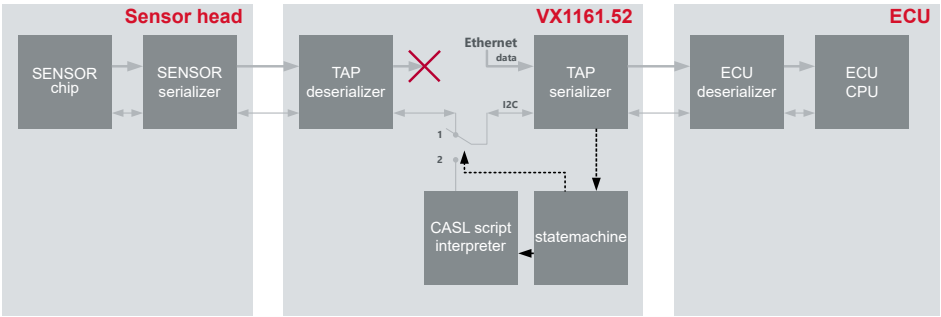


Figure 111: 1TX with sensor mode; ECU I2C accesses forwarded to sensor head; stream data for ECU supplied via Ethernet

The TX mode can be used to replace a sensor head which usually consists of the sensor serializer and the sensor chip. In typical applications both chips are configured by the ECU firmware via I2C. Since the VX1161.52 only provides the serializer and does not emulate any other I2C devices, the attached ECU firmware will fail accessing the sensor chip and enter an error state. To overcome this issue, either the ECU firmware must be enabled to cope with the missing sensor (code instrumentation, HIL mode) or the "1TX with sensor" mode in the Channel Group configuration can be used. The 1TX with sensor mode uses a real sensor head. I2C and GPIOs are forwarded to the sensor head like in TAP mode but data from the sensor is ignored and data from Ethernet is sent to the ECU.

Modified Registers

The following register values are modified by the VX1161 firmware during initialization:

Register Name	Register Address	Default Value	Modified Value	Description
Serializer				
I2C_DEVICE_ID	0x00	-	configured value	Device ID is set according configuration
SCL_HIGH_TIME	0x0B	0x7F	0x06	set I2C baudrate to 1Mbps
SCL_LOW_TIME	0x0C	0xF0	0x0B	set I2C baudrate to 1Mbps

GPIO Interconnect

The GPIOs of the serializer are connected to the GPIOs of the VX1161.52. The connections to the TAP deserializer are switched off.

For frame synchronization, one GPIO can be used as frame trigger input of the VX1161.52 to output a frame (see section Frame Trigger Detector on page 327). Additionally, the Sensor Trigger Generator can be used to toggle one serializer GPIO per channel. Because one GPIO per channel can be used to trigger the frame transmission, the generator can be used as frame rate generator to the Frame Trigger Detector (see section Frame Trigger Detector on page 327).

The GPIOs of the serializer are connected to the GPIOs of the sensor trigger generator as follows:

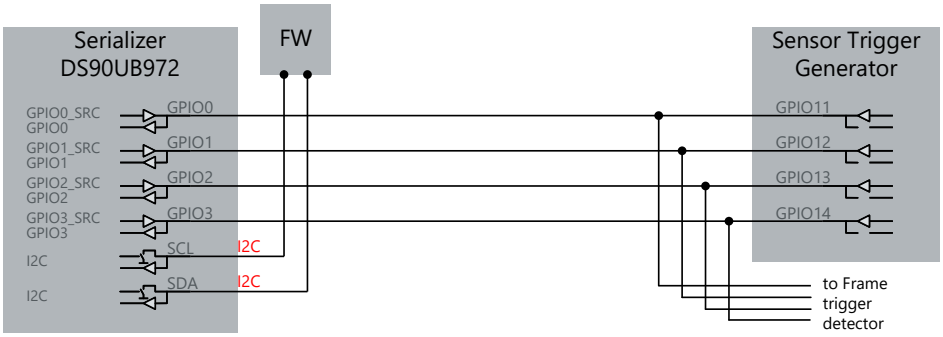


Figure 112: TX GPIO architecture; GPIO connections between the TX serializer, the Sensor Trigger Generator and the Frame Trigger Detector

Signal of Serializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**
CH2- GPIO_0	IO	CH2 - GPIO_11
CH2- GPIO_1	IO	CH2 - GPIO_12
CH2- GPIO_2	IO	CH2 - GPIO_13
CH2- GPIO_3	IO	CH2- GPIO_14
CH4 - GPIO_0	IO	CH4 - GPIO_11
CH4 - GPIO_1	IO	CH4 - GPIO_12
CH4 - GPIO_2	IO	CH4 - GPIO_13
CH4 - GPIO_3	IO	CH4 - GPIO_14

4.5.19.10 Sensor Trigger Generator

The VX1161.52 provides a Sensor Trigger Generator for every channel. This generator can be used to generate a trigger signal to one of the GPIO/MFP pins. If the serializer and deserializer are configured correctly via CASL scripts, this trigger signal will be forwarded to the sensor and triggers the acquisition of data.

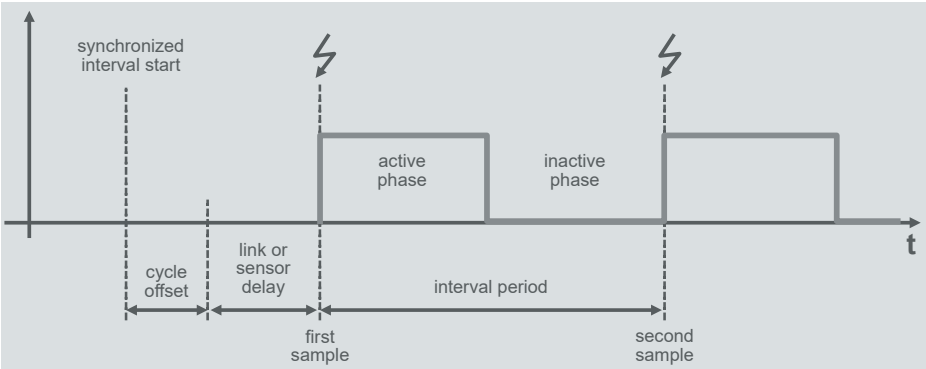


Figure 113: Sensor Trigger Generator with adjustable interval period

The generators of one module use one hardware timer as time base. This timer can be synchronized via PTP between multiple cards. If the PTP service is available, the trigger generation on all channels on all synchronized modules will be synchronous. If the PTP service is not available, only the channels on one module will be synchronous.

The current implementation requires that every VX1161.52 can exchange Ethernet broadcast messages with every other VX1161.52 during measurement start. These broadcast messages are used to negotiate the precise start of the first Frame Trigger period.

If a measurement setup consists of several VX1161 racks, there must be a path the broadcast messages can take to reach all other cards. Since Windows natively does not forward Ethernet broadcast messages from a receiving port to all other Ethernet ports, another connection is needed. To provide that path, an additional Ethernet cable between all VX1161 racks is required. The Ethernet switch inside the VX1161.22 uplink module will distribute those broadcast messages. Any Ethernet port on the VX1161.22 can be used due to the low bandwidth requirement.

The generators start outputting their trigger signal at start of measurement of the receiving application and it will stop generation at measurement stop. As a result, the sensor might not generate data when the measurement is stopped.
When the measurement is started, the time on all channels starts simultaneously and all generators start with their cycle offset times.

The trigger signal has two phases: the active and the inactive phase. Both can be configured in their duration. Together they define the trigger interval period, which defines the frequency of trigger pulses and the frame rate. Trigger generation always starts with the active phase. The Cycle offset parameter can be used to compensate delays in serial links or sensors. For sensors on fast paths, a delay can be added to trigger at the same time as sensors on slow paths.

For sensors requiring falling edges as trigger, the polarity of the output signal can be selected. If not inverted, the active phase starts with a rising edge which is intended to trigger the sensor.

Note
Care must be taken to avoid signal collisions between serializer or deserializer outputs and Sensor Trigger Generator outputs.

The Sensor Trigger Generator is configured via VXconfig and the following parameters:

Setting	Description
Enable	Activates the Sensor Trigger Generator. If disabled, the GPIOs are high impedance.
Cycle Offset	Defines a static delay from the ideal interval start in nano seconds. Can be used to compensate different delays in between several links or sensors.
Active Phase	Defines the duration of the active phase in nano seconds.
Inactive Phase	Defines the duration of the inactive phase in nano seconds.
GPIO output pin to trigger	GPIO pin of the VX1161.52, where the trigger signal shall be output. Check the GPIO/MFP section of your card to see which pins can be used and how they map to deserializer or serializer pins.
GPIO polarity active phase	Can be used to invert the output signal polarity. - checked (default): active phase starts with a rising edge. Active phase is high. Inactive phase is low. - unchecked: active phase starts with a falling edge. Active phase is low. Inactive phase is high.

4.5.19.11 Frame Trigger Detector

The VX1161.52 provides a Frame Trigger Detector for every channel in TX mode. This detector can be used to detect the change of one GPIO/MFP pin and trigger the transmission of the next video frame. This feature allows to synchronize frame transmissions to ECU trigger signals and with this to the ECU time domain.

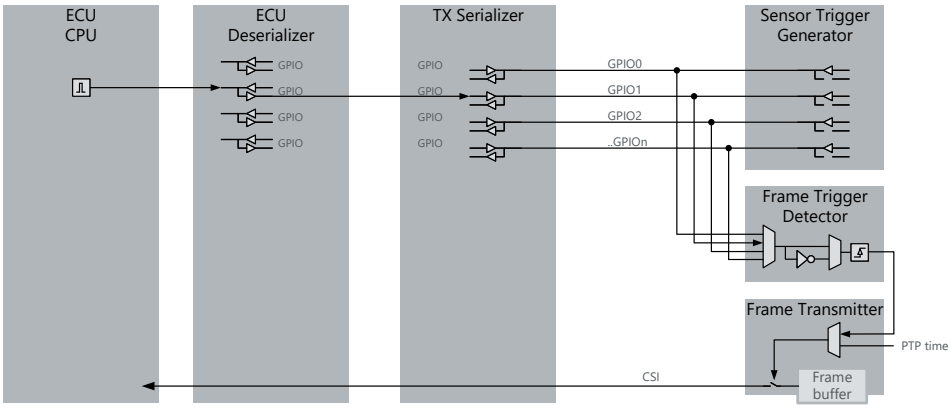


Figure 114: CPU generates frame trigger; Frame Trigger Detector detects edges and triggers the frame transmission

The detector can be fed by one GPIO/MFP of the channel group the channel belongs to. For selecting the active rising or falling edge, chose the polarity of the trigger signal.

Note
Care must be taken to avoid signal collisions between serializer outputs and Sensor Trigger Generator outputs.

The Frame Trigger Detector is configured via VXconfig and the following parameters in the Output Description section.

Setting	Description
GPIO input pin	GPIO pin of the VX1161.52, which shall feed the edge detector. Check the GPIO/MFP section of your card to see which pins can be used and how they map to your serializer pins.
GPIO polarity active phase	Specifies whether rising edges (checked) or falling edges (unchecked) on the given GPIO pin trigger the frame transmission.

4.5.19.12 Required Serializer and Deserializer Settings

The following requirements apply to the serializer and deserializer on the VX1161.52. The settings for the connected serializer at the sensor and deserializer in the ECU are allowed to differ according FPDLINK4 capabilities.

FPDLINK4 serializers are sensitive to incompatible CSI clocking schemes. If a serializer and its feeding data source are configured differently, the serializer will not accept any valid data.

CSI Continuous Clock

Due to internal technical reasons, the CSI clocks on all channels must run in continuous CSI clock mode only. This mode is initially configured by the VX1161.52 firmware. Changing this setting will interrupt reception of data.

Note
ECU write accesses to the TAP serializer are disabled via the LOCAL_WRITE_DISABLE bit in the TAP serializer. This must be considered regarding its impact to functional safety implementations.

CSI Clock Lane Speed

Due to internal reasons, the CSI clock speed of the serializer must be configured via the Channel Group Mode and not be changed. The VX1161.52 firmware will configure this at start-up. If this setting is changed via CASL scripts, data transmission will no longer be possible.

CSI Signal Settings

The CSI lanes of the TAP serializer DS90UB971 do not require any lane swap, lane inversion or timing adjustments.

4.5.19.13 POC Supply

The VX1161.52G delivers power to connected sensor heads via the coax cable using Power-over-Coax (POC). It has an internal, isolated supply that is adjustable and independent of VX GND. It can source up to 2 W. The voltage can be switched off or adjusted from 5 V ... 12 V using VXconfig.

In TAP mode, the internal supply can be deactivated, allowing the POC supply of the ECU to be forwarded to the sensor head. This gives the ECU control over the sensor's power.

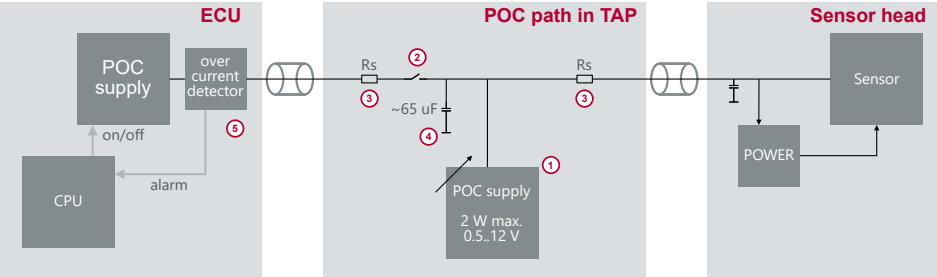


Figure 115: POC structure in VX1161.52G TAP channels

The internal series resistance (Rs) (3) of the POC bypass path is approximately 2 x 2.5 Ohms, and the internal capacitance (4) is approximately 65 µF. This capacitance can cause or increase inrush currents for the ECU that sources power and might trigger overcurrent detection features (5). In such cases, it might be an option to source the sensor from the VX (1) instead of the ECU, but then the ECU would have no control over the sensor power.

Furthermore, the internal VX POC supply can be output to the ECU for use cases in which CSI ingestion PODs require POC supply.

The POC voltage level, generation, and delivery direction are controlled via the POC voltage settings for the TAP_RX and TAP_TX ports in VXconfig as follows:

Note
Sourcing power in TAP mode is supported since VXtools v4.9.

Settings				
POC voltage V_{RX}on TAP RX channel	0V	5...12V	0V	5...12V
POC voltage V_{TX}on TAP TX channel	0V	0V	5...12V	5...12V
Effects				
POC bypass (2)	closed	open	closed	closed
VX POC source (1) voltage	off	V_{RX}	V_{TX}	Min(V_{RX},V_{TX})
Use case	ECU supplies sensor	VX supplies sensor	VX supplies ingest POD	VX supplies sensor and ingest POD

4.5.19.14 Technical Data

Weight	210 g
Power consumption	► P_{init}; initialization (~500ms): 500mW ► $P_{standby}$; sleep mode: <18 mW ► P_{run}; normal operation excl. POC: VX1161.52 max. 10W
POC (power over cable)	► V_{poc}; voltage sourced by VX1161.52: 0 V, 5...12 V configurable ► P_{poc}; maximum power delivered per RX channel; sourced by VX1161.52: 2 W ► $V_{PocTapMax}$; voltage forwarded from ECU by TAP: 3...18 V ► $I_{PocTapMax}$; max. current forwarded from ECU by TAP: 500 mA ► Electrical isolation between TAP_CH1/2 and TAP _CH3/4 and chassis max. 50 V ► Rpoc; DC resistance of POC forwarding path in TAP: typ. 3.952 Ohm
Minimum power -off time	► minimum time between powering-off and re-appying power: >2 s
Start-up time	► t_{init}; initialization after power up: ~ 500 ms ► $t_{wakeLink}$; wake up after sleep; FPDLINK4 link stable: < 50 ms, depends on CASL script execution ► $t_{wakeReady}$; wake up after sleep; FPDLINK4 data to Ethernet: < 10 s
Temperature range	-40 °C ... +60 °C
Maximum bandwidths	► Card to host via Ethernet: 7.2 Gbps net; total for all channels together ► Every TAP / RX / TX channel FPDLINK4: 7.2 Gbps gross; 6.3 Gbps net
Latencies/delays	► t_{delay}; FPDLINK4 TAP; TAP_RX to TAP_TX; forward, backward, I2C and GPIO channels: ~ 1 µs. Additional serializer and deserializer delay: ~ one CSI packet. ► $t_{delayGPIO;}$ additional delay of GPIOs; forward, backward: dependent on control channel baud rate; 600 ns @ 50 Mbps, 3 µs @ 10 Mbps or 12.5µs @ 2.5 Mbps
Connector	(Single signal) FAKRA Type Z; Rosenberger 59S2AQ-40MT5-Z; color: cyan; compatible to all other (single signal) FAKRA connector types; 6 GHz max; >=25 mating cycles
Maximum cable length	8 m, in both RX and TX direction at the same time
Part number	180166

4.5.20 VX1161.52H 2xTAP GMSL2 MX96718A/MX96717

The VX1161.52H 2xTAP GMSL2 MX96718A/MX96717 is a two-channel card (CH1...CH4) for GMSL2 data streams with tunneling mode.

TAP means that a receiver (deserializer) and a transmitter (serializer) are connected back-to-back and the received data is forwarded to the transmitter and duplicated towards the logging system. Both channels contain one Analog Devices® MAX96718A deserializer and one Analog Devices® MAX96717 serializer each.

The MAX96718A/MAX96717 transceivers support the Analog Devices Tunneling Mode and Reference Over Reverse (RoR).

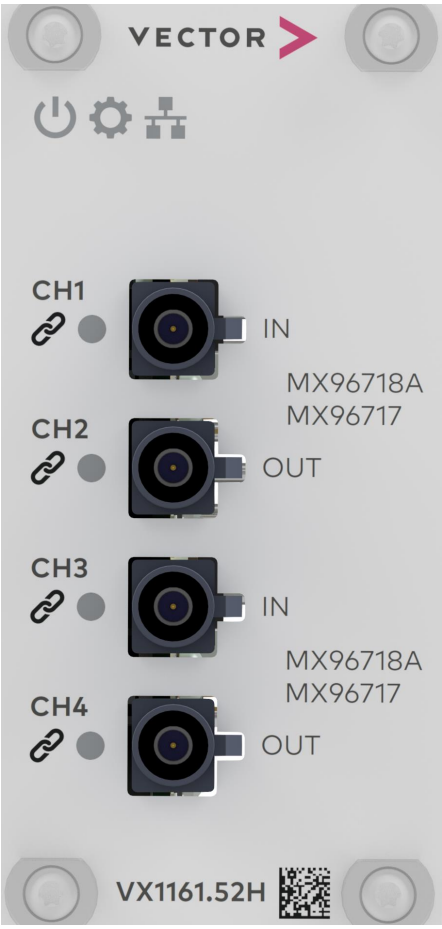


Figure 116: VX1161.52H 2xTAP GMSL2 MX96718A/MX96717

The operating mode of the card can be configured either in TAP, RX or TX mode. The VX1161.52H supports binary lossless compression (see section Lossless Compression on page 332).

The card can be configured with VXconfig and updated to new software versions with VXupdate. Initialization of the transceivers and remote sensors can be done via CASL language scripts within VXconfig.

Note
VX1161.52H is supported by VXtools v4.6 and later.

Note
GMSL systems require after link lock several CSI packets to activate their internal pipelines. This causes the first packets to get lost inside the deserializer or serializer which will corrupt the first frame.

For frame precise measurements or replays, some dummy sensor data must be flushed through the entire sensor ECU connection.

4.5.20.1 Lossless Compression

The video interface supports the hardware-based binary lossless compression of video streams. This compression method typically reduces the required bandwidth for video transmission, logging, and storage by 40 % ... 60 %, depending on the video content. Each channel of the interface can be compressed. The compression pipelines support the following pixel formats: RAW8, RAW10, RAW12, RAW16, and YUV422 8-bit. For other pixel formats, please contact Vector.

Four different compression modes are supported:

- **MobilEye on PC**
  This mode compresses the recording on the computer and requires a MobilEye license.

- **MobilEye on VX**
  This mode enables lossless compression on VX hardware and requires a MobilEye license. It reduces the bandwidth from the VX to the recording computer and offloads the necessary computing power.

- **Vector on VX**
  This mode enables lossless compression on the VX hardware and does not require a license. Only the payload of all received CSI long packets (all data types) is fed into the compression pipelines. The CSI headers and footers are stripped off beforehand. The embedded line payload is also fed into the compressor and reconstructed when decompressed. In this mode, all lines (embedded data, visible, and statistic lines) must be the same length as set in VXconfig. Enabling line length padding in the sensor may be necessary. To ensure good compression efficiency and enable later decoding, it is essential that all lines be the same length as the length set in VXconfig. The sensor may need to pad the embedded lines to the full length. Starting with VXtools V5.1, the interface enforces this.

- **Vector on VX V2**
  This mode has been supported since VXtools V5.1 and enables lossless compression on the VX hardware and does not require a license.

  If selected:

  - The CSI packets specified via CSI leading embedded packets and CSI trailing embedded packets of a frame are stored uncompressed.
  - The CSI headers and footers of all packets are also stored uncompressed.

- The pixels are losslessly compressed on the interface hardware.
- All three stored parts are secured via individual CRCs to detect data changes during transport or storage.

Each of the two compression pipelines is capable of compression:

- one stream with up to 350 MPx/s RAW8 ... RAW16
- one stream with up to 170 MPx/s YUV422

Vector provides decompression libraries and tools for Windows and Linux.

4.5.20.2 CRC Algorithm

```
std::vector<uint32_t> VCrc32::sCrcTable;

uint32_t VCrc32::Calculate(const uint8_t* rawData, const uint64_t rawDataS-
ize, const uint8_t bitsPerSample)
{
  uint32_t crc = 0xFFFFFFFF;
  const uint32_t specialWidth = 32 - bitsPerSample;
  const int andMask = (1 << bitsPerSample) - 1;
  uint32_t usedBytes = 0;
  const uint16_t* pData = reinterpret_cast<const uint16_t*>(rawData);

  while (usedBytes < rawDataSize)
  {
    /* XOR-in next input byte into MSB of crc and get this MSB, that's our
new intermediate dividend */
const uint16_t pos = static_cast<uint16_t>(((crc ^ (*pData << spe-
cialWidth)) >> specialWidth) & andMask);
    /* Shift out the MSB used for division per look up table and XOR with
the remainder */

    crc = (crc << bitsPerSample) ^ sCrcTable[pos];
    usedBytes += 2;
    pData++;

    if (rawDataSize % 2 != 0 && usedBytes == rawDataSize - 1)
    {
      uint16_t paddedValue = *pData;
      paddedValue = paddedValue & 0xFF;
      pData = &paddedValue;
    }
  }
  return crc;
}



void VCrc32::GenerateCrc32Table(const uint8_t bitsPerSample)
{
  if (!sCrcTable.empty())
  {
    return ;
  }

  const uint32_t specialWidth = 32 - bitsPerSample;
  const uint32_t elementCount = 1 << bitsPerSample;
  sCrcTable.resize(elementCount);

  for (uint32_t dividend = 0; dividend < elementCount; dividend++)
  {
    uint32_t currentByte = dividend << specialWidth;
    for (uint8_t bit = 0; bit < bitsPerSample; bit++)
    {
       if ((currentByte & 0x80000000) != 0)
      {
        constexpr uint32_t polynomial = 0x04C11DB7;
        currentByte <<= 1;
        currentByte ^= polynomial;
       }
       else
      {
         currentByte <<= 1;
       }
     }
    sCrcTable[dividend] = currentByte;
  }
}
```

4.5.20.3 Connectors

- **CH1...4**
  The VX1161.52H provides FAKRA coaxial connectors of Type Z for connecting data sources and data sinks such as video or radar sensors and ECUs. Type Z connectors are compatible to all other coding types of single signal FAKRA connectors. The connector properties can be found in section Technical Data on page 362.

4.5.20.4 LEDs

The status of the card is displayed by system LEDs in the lid top. The status of every channel is displayed by a channel LED next to every channel connector.

Power

Symbol	Sequence	Description
	R R R R R R R R	Device is powered but Firmware not running. The Power LED will light up red at the beginning of the boot sequence of the Interface Card. ► Please contact Vector VX support if the Power LED stays red for more than 10 seconds. In this case a critical error has occurred.
	G G G G G G G G	Firmware loaded successfully.
	G ☐ G ☐ G ☐ G ☐	Response to ”Identify VX device command” (VXconfig).

System

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Firmware not loaded.
	R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Interface Card, its system components are initialized. If a component fails to initialize properly, the System LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
	G G G G G G G G	Device is in normal operating mode.
	G ☐ G ☐ G ☐ G ☐	Firmware update in progress.

Ethernet

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Backplane link not established.
	R R R R R R R R	Backplane link error.
	G G G G G G G G	Backplane link established.
	G ☐ G ☐ G ☐ G ☐	Backplane link activity.

Channel

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Channel disabled.
	R R R R ☐ ☐ ☐ ☐	Channel enabled, invalid configuration. See VXconsole for details.
	R R R R R R R R	Channel enabled, no link established.
	G G G G G G G G	Channel enabled, link established, no data traffic.
	G ☐ G ☐ G ☐ G ☐	Channel enabled, link established, data traffic.

4.5.20.5 Boot and Link-up-Time

When the VX1161 is connected to power, the streaming card must prepare itself for a fast wake-up. For about t_{init} the power consumption will be P_{init}. After that time, the card enters standby mode and the power consumption is reduced to its minimum $P_{standby}$.

When the VX1161 is woken up (see section Connectors on page 176), the card is already prepared and can establish both ECU and sensor link within $t_{wakeLink}$. After wakeup, the card can provide data to the recorder after $t_{wakeReady}$.

The power consumption during normal operation consist of the maximum power of P_{run} drawn by the card as well as the power consumed by supplied cameras P_{POC}.

4.5.20.6 Changing the Mode of Operation

The VX1161.52H can be used in one of the following use cases:

- Video RX/TAP (video reception / logging)
- Video TX (video transmission / replay)

In these modes, sensors or ECUs with GMSL2 transceivers can be directly attached to log or reply streams of data.

The mode of operation can be changed in VXconfig on the Streaming Interface Card dialog.

4.5.20.7 Channel in TAP Mode

The TAP is designed to be as transparent as possible to the ECU.

If a channel is configured in TAP mode, the incoming GMSL2 data stream is received by the MX96718A TAP deserializer. The data payload (e. g. video or radar frames) is then duplicated. One copy is forwarded to the MX96717 TAP serializer and one copy is forwarded to the recording data processor.

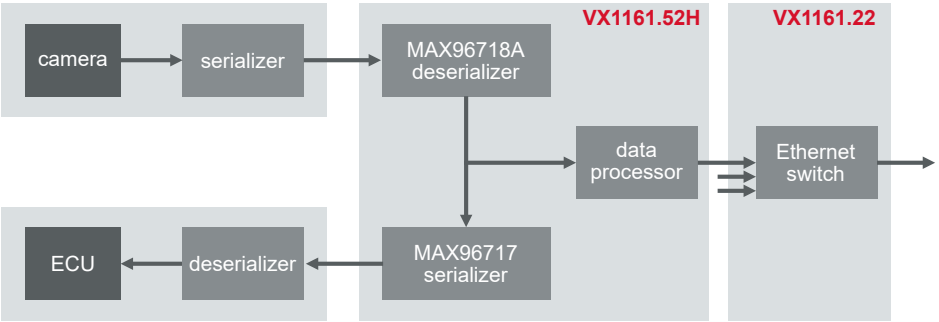


Figure 117: Channel in TAP mode; payload duplication

Control accesses from the ECU to the sensor via the I2C control channel or MFPs/GPIOs are forwarded from the TAP serializer to the TAP deserializer.

TAP Initialization

The initialization sequence of the sensor and the ECU links follows this chart:

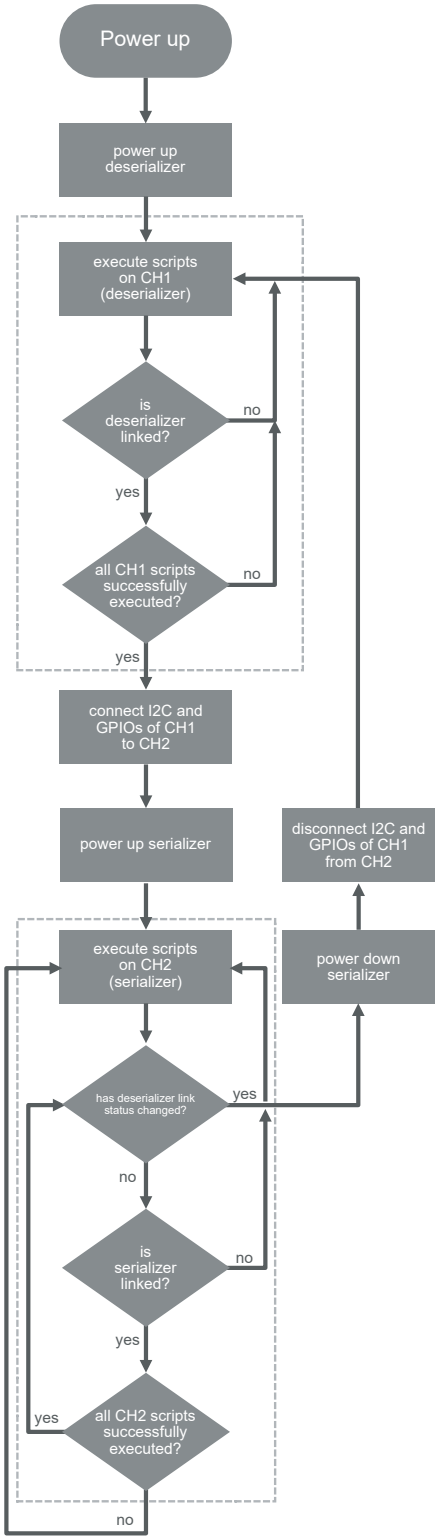


Figure 118: TAP initialization sequence

After power-up, the POC voltage is applied as configured and the deserializer is activated.

Next, the enabled scripts for the TAP RX channel (CH1/CH3) are immediately executed in an initial run (script 1, 2, 3) to configure the TAP deserializer, the sensor serializer and the sensor imager as early as possible.

After the initial script run on the TAP RX channel, the script execution engine waits for a TAP deserializer lock.

When the TAP deserializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the following conditions apply:

1. Script is not configured as "run once" (CASL SetConfig(7,1))
2. OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
3. OR script terminated with a return value ≠ 0 in previous run

If the TAP deserializer link unlocks, the script sequence repeats from the initial TAP deserializer run on, the I2C and GPIO bridges will be opened and the TAP serializer will be deactivated.

When the TAP deserializer locks, the I2C bus and GPIOs of the TAP serializer and TAP deserializer are connected. This allows the ECU to configure and control the sensor (see section GPIO Interconnect).

When the TAP deserializer locks, the TAP serializer is activated and the script execution on the TAP TX channel (CH2/CH4) are executed in an initial run (script 1, 2, 3).

After the initial script run on the TAP TX channel, the script execution engine waits for a TAP serializer lock.

When the TAP serializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the following conditions apply:

1. Script is not configured as "run once" (CASL SetConfig(7,1))
2. OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
3. OR script terminated with a return value ≠ 0 in previous run

If the TAP serializer link unlocks, the sequence repeats from the initial TAP serializer run on.

If the TAP deserializer link unlocks now, the I2C and GPIO bridges will be opened, the TAP serializer will be deactivated and the script sequence repeats from the initial TAP deserializer run on.

I2C Architecture

Without the TAP, the ECU is directly connected to the sensor head. Via the ECU deserializer and the sensor serializer it can configure and control all devices in that path. By inserting the TAP, the ECU deserializer is now connected to the TAP serializer and the sensor serializer is now connected to the TAP deserializer.

To allow the ECU controller to configure and control the sensor serializer and the sensor also in TAP mode, it might be necessary to configure the TAP serializer and the TAP deserializer via CASL scripts. Forwarding of pass through I2C channels must be configured, whereas the configuration I2C channel is enabled by default.

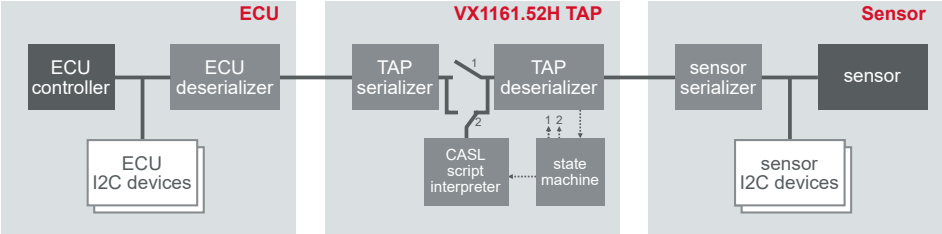


Figure 119: TAP I2C architecture; ECU configures ECU deserializer + TAP serializer + sensor

The I2C address of the TAP deserializer and the TAP serializer can be configured in VXconfig. Both chips start with their hardware strapped address and are then overwritten by the VX1161 firmware with the address configured in VXconfig. Additionally, the SetConfig(8,x) CASL command can be used to change the I2C address dynamically.

The sensor serializer I2C address is defined by its hardware straps. It can be changed via CASL scripts and must not be the same as the TAP serializer address.

The sensor I2C address is defined by the sensor itself. Accesses by the ECU will transparently be forwarded through the sensor link.

The additional TAP link will cause additional transmission delays (via clock stretching) to any sensor register access. These delays will reduce the effective control channel bandwidth and could affect system timings.

Modified Registers

Register Name	Register Address	Default Value	Modified Value	Description
Deserializer				
I2C_DEVICE_ID	0x0000	-	configured value	Device ID is set according configuration
RLMS58	0x1458	0x18	0x28	Value recommended by GMSL2 user guide V19
RLMS59	0x1459	0x58	0x68	Value recommended by GMSL2 user guide V19
RLMS4	0x1404	0x4B	0x29	Value recommended by GMSL2 user guide V19
RLMSA4	0x14A4	0xBD	0xC8	Value recommended by GMSL2 user guide V19
not documented	0x140A	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
not documented	0x140B	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
RLMS4	0x1504	0x4B	0x29	Value recommended by GMSL2 user guide V19
RLMSA4	0x15A4	0xBD	0xC8	Value recommended by GMSL2 user guide V19
not documented	0x150A	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
not documented	0x150B	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
CTRL0	0x0010	0x11	0x15	Bit 2 : REG_ENABLE set; enables the LDO; required by VX1161.52 hardware
CTRL2	0x0012	0x00	0x10	Bit 4: REG_MNL set; enables manual LDO control; required by VX1161.52 hardware
BACKTOP25	0x0320	0x3F	0xX8	Bit 4:0 : PHY1_CSI_TX_DPLL_ PREDEF_FREQ set to 8; sets CSI clock rate to 800 Mbps; required to be compatible with the VX1161.52 hardware
MIPI_PHY0	0x0330	0x0x	0x8x	Bit 7: FORCE_CSI_OUT_EN set; enables continuous CSI clock; required by VX1161.52 hardware
Serializer				
I2C_DEVICE_ID	0x0000	-	configured value	Device ID is set according configuration
RLMS4	0x1404	0x4B	0x29	Value recommended by GMSL2 user guide V19
RLMSA4	0x14A4	0xBD	0xC8	Value recommended by GMSL2 user guide V19
not documented	0x140A	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
not documented	0x140B	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
REG5	0x0005	0x00	0x80	Bit 7 : LOCK_EN is set; enables the lock output; required by the VX1161.52 firmware to monitor the link status
CTRL0	0x0010	0x11	0x15	Bit 2: REG_ENAB is set; enables the regulator; required to be compatible with the VX1161.52 hardware; value recommended by GMSL2 user guide V19

Register Name	Register Address	Default Value	Modified Value	Description
				Bit 6: RESET_LINK is set and cleared; used to hold back link after power up while VX1161.52 FW is configuring system
CTRL2	0x0012	0x00	0x10	Bit 4: REG_MNL is set; enables manual regulator control; required to be compatible with the VX1161.52 hardware; value recommended by GMSL2 user guide V19
REG1	0x0001	0x08	0x08	Bit 4: DIS_REM_CC is set and cleared; used to re-establish link to serializer

Detection of TAP by ECU Firmware for Measurement Mode

In cases where the ECU needs to know whether a TAP is present, CASL scripts can be used to manipulate unused spare default register values in the CAM serializer. The ECU can then read the register and check whether it contains the default value (no TAP) or the CASL script value (TAP node).

MFP Interconnect

For forwarding MFPs (multi function pins) from the ECU to the sensor the TAP deserializer and the sensor serializer registers must be configured properly. This can be done via CASL scripts. Static MFP settings can also be configured via CASL scripts.

Additionally, the VX1161.52H has a Sensor Trigger Generator which can generate synchronized trigger pulses on one MFP pin per channel (see section Sensor Trigger Generator on page 357).

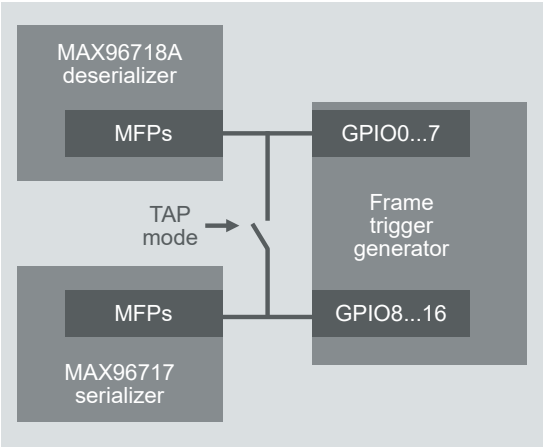


Figure 120: In TAP mode, some MFPs of serializer and deserializer are connected as long as the deserializer is GMSL locked to the remote serializer

Note
Care must be taken to avoid signal collisions between MFP outputs and Sensor Trigger Generator outputs.

Note
MFP1 of deserializer and MFP3 of serializer must stay configured as LOCK outputs. This function will be configured by the card firmware at boot time. If the function of these pins is changed, the TAP will no longer be able to detect unlocks and script execution might be disturbed.

Forwarding of analog MFP features is not supported.

On GMSL2, the transmission of MFPs can be configured either in standard or delay compensation mode. For frame trigger signals the delay compensated mode should be used to reduce jitter. Both modes can be configured via CASL scripts. The transmission of MFP signals takes about $t_{delayMFP}$ per GMSL2 link. In TAP mode, the ECU link and the sensor link add both this delay.

In cases where sensor triggers are transferred via MFPs, it must be considered that the sensor links of two TAP links are not synchronized. Therefore, a skew between the two independent VX1161.52H TAP channels of $t_{delayMFP}$ must be assumed.

The MFPs of the TAP serializer are bidirectionally connected to the MFPs of the deserializer and the Sensor Trigger Generator GPIOs as follows:

Signal of Serializer		Connected to Signal of Deserializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**	**Direction**	**Name**
CH2 - MFP0	IO	CH1 - MFP0	IO	CH2 - GPIO_16 = CH1 - GPIO_0
CH2 - MFP1	O	CH1 - MFP8	I	CH2 - GPIO_15
CH2 - MFP2	O	CH1 - MFP7	I	CH2 - GPIO_14 = CH1 - GPIO_7
CH2 - MFP3	IO; LOCK	-	-	(CH2 - GPIO_13; do not use)
CH2 - MFP4	IO	CH1 - MFP4	IO	CH2 - GPIO_12 = CH1 - GPIO_4
CH2 - MFP5	I	CH1 - MFP3	O	CH2 - GPIO_11 = CH1 - GPIO_3
CH2 - MFP6	I	CH1 - MFP2	O	CH2 - GPIO_10 = CH1 - GPIO_2
CH2 - MFP7	IO	-	-	CH2 -GPIO_9
CH2 - MFP8	IO	-	-	CH2 -GPIO_8
CH2 - MFP9	IO_OD	CH1 - MFP9	I	-
CH2 - MFP10	IO_OD	CH1 - MFP10	I	-
-	-	CH1 - MFP1	IO; LOCK	(CH1 - GPIO_1; do not use)
-	-	CH1 - MFP5	IO	CH1 -GPIO_5
-	-	CH1 - MFP6	IO	CH1 -GPIO_6
-	-	CH1 - MFP11	IO_DD	-
-	-	CH1 - MFP12	ID_DD	-
CH4 - MFP0	IO	CH3 - MFP0	IO	CH4 - GPIO_16 = CH3 - GPIO_0
CH4 - MFP1	O	CH3 - MFP8	I	CH4 - GPIO_15
CH4 - MFP2	O	CH3 - MFP7	I	CH4 - GPIO_14 = CH3 - GPIO_7
CH4 - MFP3	IO; LOCK	-	-	(CH2 - GPIO_13; do not use)
CH4 - MFP4	IO	CH3 - MFP4	IO	CH4 - GPIO_12 = CH1 - GPIO_4
CH4 - MFP5	I	CH3 - MFP3	O	CH4 - GPIO_11 =

Signal of Serializer		Connected to Signal of Deserializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**	**Direction**	**Name**
				CH3 - GPIO_3
CH4 - MFP6	I	CH3 - MFP2	O	CH4 - GPIO_10 = CH3 - GPIO_2
CH4 - MFP7	IO	-	-	CH4 -GPIO_9
CH4 - MFP8	IO	-	-	CH4 -GPIO_8
CH4 - MFP9	IO_OD	CH3 - MFP9	I	-
CH4 - MFP10	IO_OD	CH3 - MFP10	I	-
	-	CH3 - MFP1	IO; LOCK	(CH3 - GPIO_1; do not use)
	-	CH3 - MFP5	IO	CH3 -GPIO_5
	-	CH3 - MFP6	IO	CH3 -GPIO_6
	-	CH3 - MFP11	IO_DD	-
		CH3 - MFP12	ID_DD	-

Signal mapping of TAP deserializer MFPs to trigger generator GPIOs; directions with view of deserializer (IO_OD: Input+open drain output); LOCK: Do not use. Signals required internally.

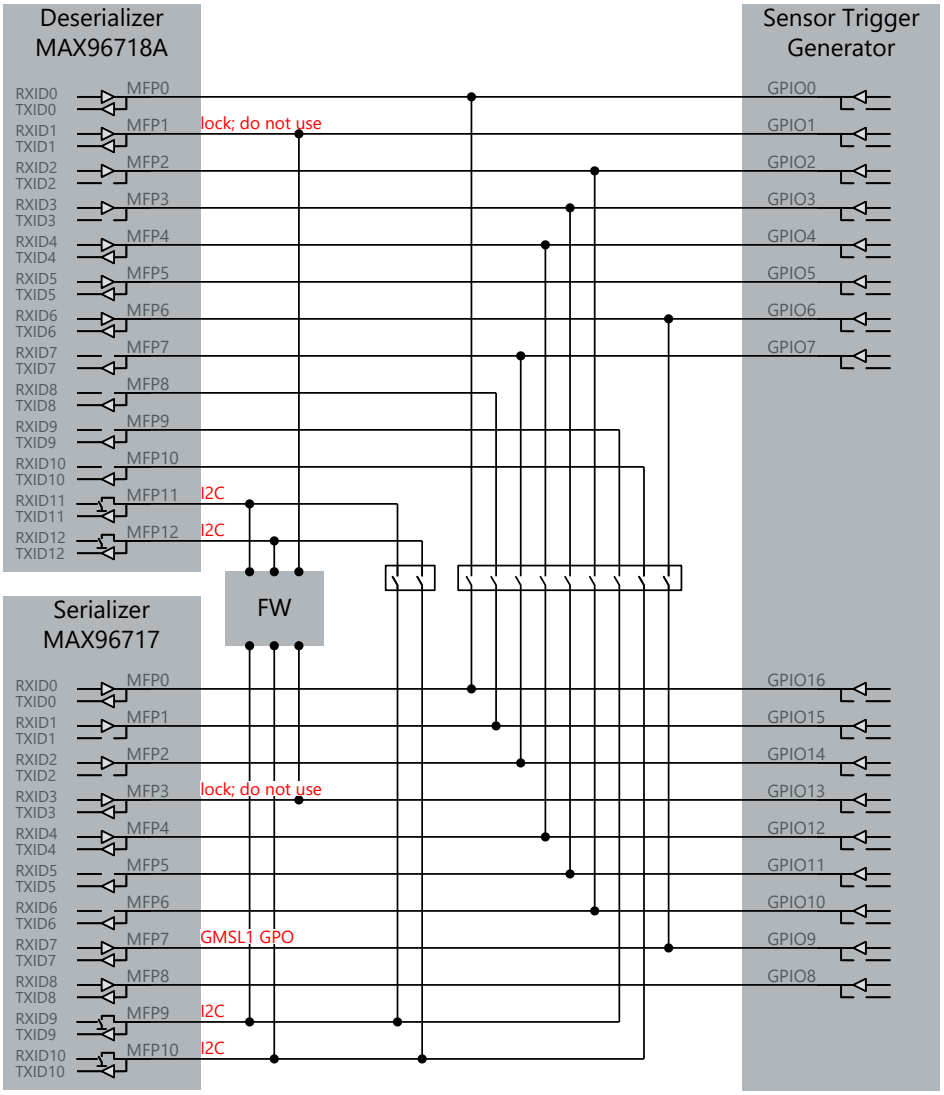


Figure 121: TAP GPIO architecture; GPIO connections between TAP deserializer, TAP serializer and Sensor Trigger Generator; switches are closed when TAP deserializer link is stable

4.5.20.8 Channel in RX Mode

If a channel is configured in RX mode, the incoming GMSL2 data stream is received by the MAX96718A deserializer and forwarded to the recording data processing.

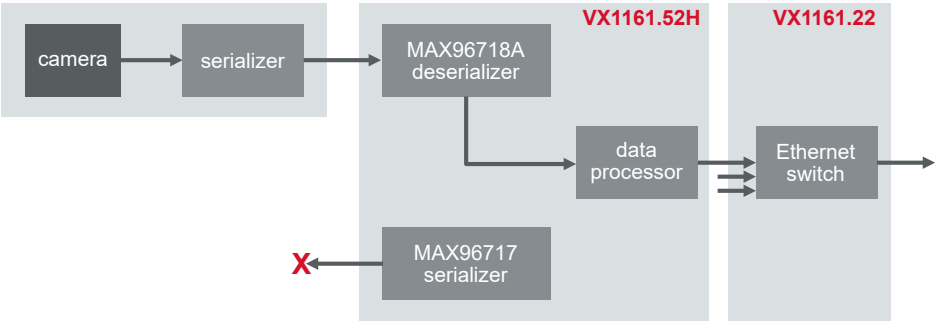


Figure 122: TAP channel in RX mode; payload reception

Initialization

The initialization sequence of the sensor link follows this chart:

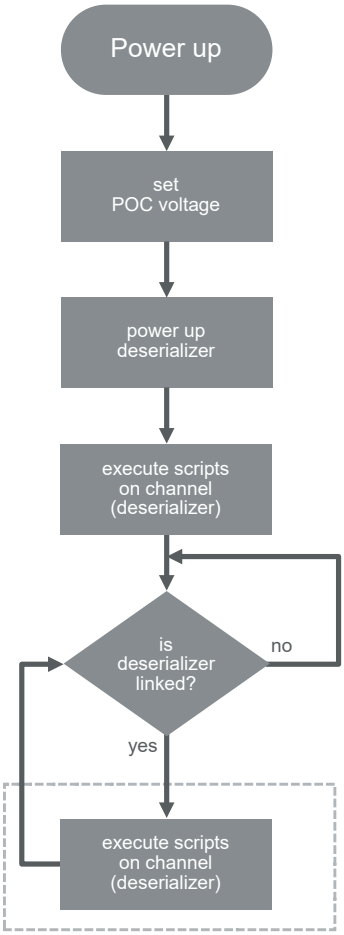


Figure 123: RX initialization sequence

After power-up, the POC voltage is applied as configured and the deserializer is activated.

Next, the enabled scripts for the RX channel are executed immediately (script 1, 2, 3) to configure the deserializer, sensor serializer and sensor as early as possible.

After the initial run, the script execution engine waits for a deserializer lock.

When the deserializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the follwing conditions apply:

- Script is not configured as "run once" (CASL SetConfig(7,1))
- OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
- OR script terminated with a return value ≠ 0 in previous run

If the deserializer link unlocks, the script sequence repeates from the inital deserializer run on.

I2C Architecture

In RX mode, the TAP deserializer, sensor serializer and sensor need to be configured via CASL scripts. A dynamic sensor reconfiguration is currently not supported.

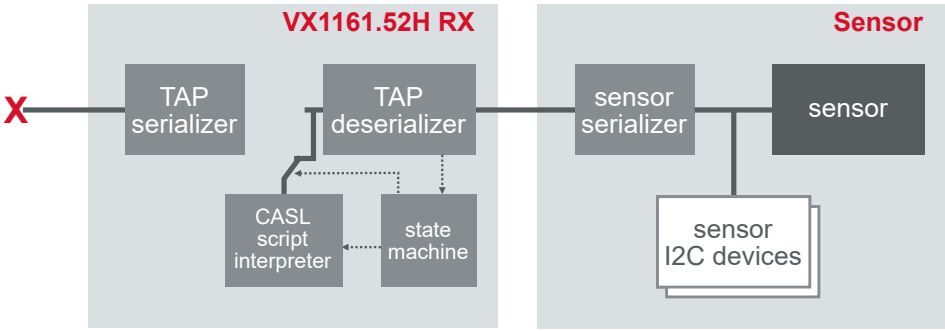


Figure 124: RX mode; I2C architecture; CASL scripts configure deserializer, serializer and sensor

The I2C address of the TAP deserializer can be configured in VXconfig. The chip starts with its hardware strapped address and is then overwritten by the VX1161 firmware with the address configured in VXconfig. Additionally, the SetConfig(8,x) CASL command can be used to change the I2C address dynamically.

The sensor serializer I2C address is defined by its hardware strapped pins and could be changed via CASL scripts.

The sensor I2C address is defined by the sensor itself. CASL scripts can be used to configure the sensor via I2C transactions. To enable I2C accesses to the sensor, deserializer and serializer registers might have to be set properly in advance.

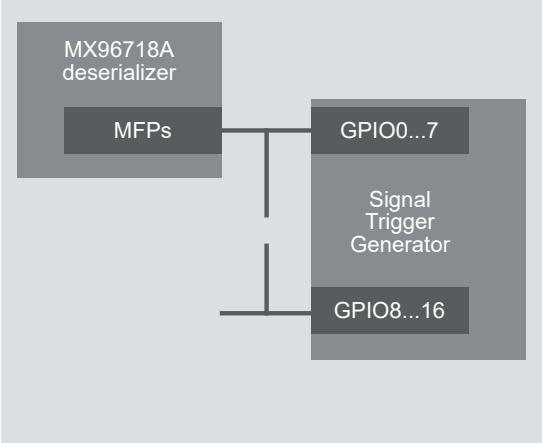
Modified Registers

Register Name	Register Address	Default Value	Modified Value	Description
Deserializer				
I2C_DEVICE_ID	0x0000	-	configured value	Device ID is set according configuration
RLMS58	0x1458	0x18	0x28	Value recommended by GMSL2 user guide V19
RLMS59	0x1459	0x58	0x68	Value recommended by GMSL2 user guide V19
RLMS4	0x1404	0x4B	0x29	Value recommended by GMSL2 user guide V19
RLMSA4	0x14A4	0xBD	0xC8	Value recommended by GMSL2 user guide V19
not documented	0x140A	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
not documented	0x140B	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
RLMS4	0x1504	0x4B	0x29	value recommended by GMSL2 user guide V19
RLMSA4	0x15A4	0xBD	0xC8	value recommended by GMSL2 user guide V19
not documented	0x150A	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
not documented	0x150B	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
CTRL0	0x0010	0x11	0x15	Bit 2 : REG_ENABLE set; enables the LDO; required by VX1161.52 hardware
CTRL2	0x0012	0x00	0x10	Bit 4: REG_MNL set; enables manual LDO control; required by VX1161.52 hardware
BACKTOP25	0x0320	0x3F	configured value	Bit 4:0 : PHY1_CSI_TX_DPLL_ PREDEF_FREQ set to the CSI speed, selected in VXconfig; required to be compatible with the VX1161.52 hardware
MIPI_PHY0	0x0330	0x0x	0x8x	Bit 7: FORCE_CSI_OUT_EN set; enables continuous CSI clock; required by VX1161.52 hardware

MFP Interconnect

The VX1161.52H has a Sensor Trigger Generator which can generate synchronized trigger pulses on one MFP pin per channel.

For forwarding MFPs (multi function pins) from the Sensor Trigger Generator to the sensor, the TAP deserializer and the sensor serializer registers must be configured properly. This can be done via CASL scripts.This also enables the configuration of static MFP levels inside the deserializer.



Note
Care must be taken to avoid signal collisions between MFP outputs and Sensor Trigger Generator outputs.

Note
MFP1 of the deserializer must stay configured as LOCK outputs. This function will be configured by the card firmware at boot time. If the function of this pin is changed, the VX1161 firmware will no longer be able to detect unlocks and script execution might be disturbed.

On GMSL2, the transmission of MFPs can be configured either in standard or delay compensation mode. For frame trigger signals, the delay compensated mode should be used to reduce jitter. Both modes can be configured via CASL scripts. The transmission of MFP signals takes about $t_{delayMFP}$ for a GMSL2 link.

The MFPs of the TAP deserializer connected to the MFPs of the Sensor Trigger Generator GPIOs as follows:

Signal of Deserializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**
CH1 - MFP0	IO	CH1 - GPIO_0
CH1 - MFP8	I	-
CH1 - MFP7	I	CH1 - GPIO_7
CH1 - MFP4	IO	CH1 - GPIO_4
CH1 - MFP3	O	CH1 - GPIO_3
CH1 - MFP2	O	CH1 - GPIO_2
CH1 - MFP9	I	-
CH1 - MFP10	I	-

Signal of Deserializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**
CH1 - MFP1	IO; LOCK	(CH1 - GPIO_1; do not use)
CH1 - MFP5	IO	CH1 - GPIO_5
CH1 - MFP6	IO	CH1 - GPIO_6
CH1 - MFP11	IO_DD	-
CH1 - MFP12	IO_DD	-
CH3 - MFP0	IO	CH3 - GPIO_0
CH3 - MFP8	I	-
CH3 - MFP7	I	CH3 - GPIO_7
CH3 - MFP4	IO	CH3 - GPIO_4
CH3 - MFP3	O	CH3 - GPIO_3
CH3 - MFP2	O	CH3 - GPIO_2
CH3 - MFP9	I	-
CH3 - MFP10	I	-
CH3 - MFP1	IO; LOCK	(CH3 - GPIO_1; do not use)
CH3 - MFP5	IO	CH3 - GPIO_5
CH3 - MFP6	IO	CH3 - GPIO_6
CH3 - MFP11	IO_DD	-
CH3 - MFP12	IO_DD	-

Signal mapping of deserializer MFPs to trigger generator GPIOs; directions with view of deserializer (IO_OD: Input+open drain output); LOCK: Do not use. Signals required internally.

4.5.20.9 Channel in TX Mode

If a channel is configured in TX mode, an incoming data stream from Ethernet is output via the MAX96718A serializer towards an ECU for example. The deserializer is kept powered-up for internal purposes, but cannot be used. Thus CH1 and CH3 are not available.

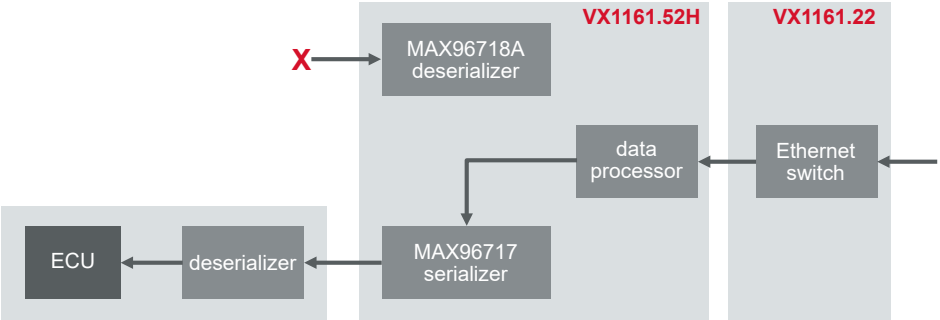


Figure 125: TAP channel in TX mode; payload transmission

Initialization

The initialization sequence of the sensor link follows this chart:

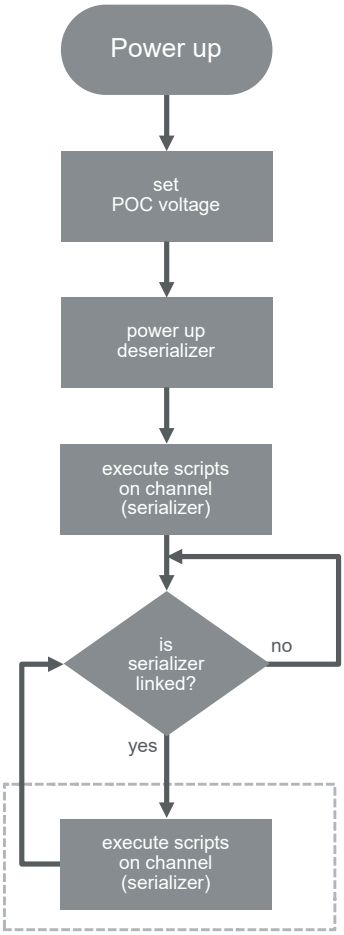


Figure 126: TX initialization sequence

After power-up, the serializer is immediately activated and the enabled scripts for the TX channel are immediately executed (script 1, 2, 3) to configure the serializer

and the ECU deserializer as early as possible.

After the initial run, the script execution engine waits for a serializer lock.

When the serializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the following conditions apply:

1. Script is not configured as "run once" (CASL SetConfig(7,1))
2. OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
3. OR script terminated with a return value ≠ 0 in previous run

If the serializer link unlocks, the sequence repeats from the initial serializer run on.

I2C Architecture

In TX mode, the serializer needs to be configured via CASL scripts. Since its configuration must match the hardware architecture, remote configuration accesses are disabled.

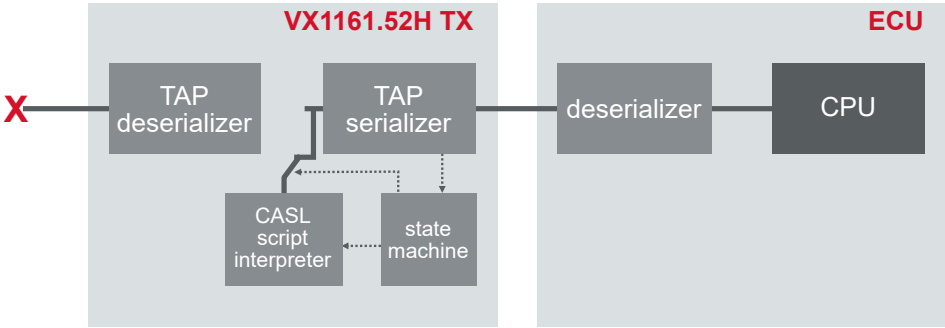


Figure 127: TX mode; I2C architecture; CASL scripts configure the serializer.

The I2C address of the TAP serializer can be configured in VXconfig. The chip starts with its hardware strapped address and is then overwritten by the VX1161 firmware with the address configured in VXconfig. Additionally, the SetConfig(8,x) CASL command can be used to change the I2C address dynamically.

The VX1161.52 does not emulate any I2C devices (besides the serializer). In order to avoid error states in the attached ECU, either the Channel Group configuration 1TX with sensor can be used or the ECU firmware must be modified to omit or ignore erroneous I2C accesses.The 1TX with sensor mode acts logically like the TAP mode but video data from the sensor is ignored and data from Ethernet is sent to the ECU .

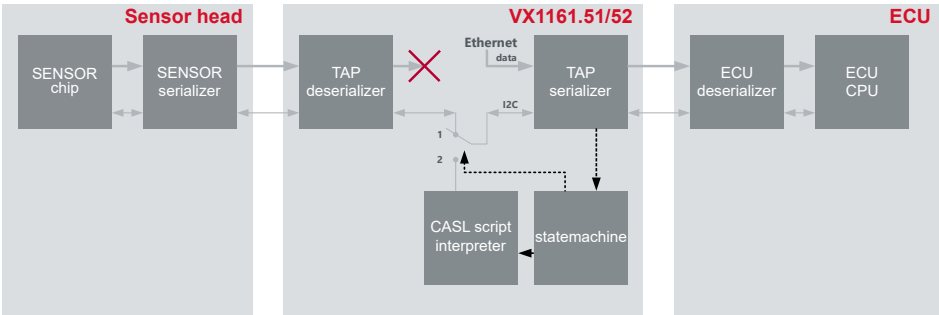


Figure 128: 1TX with sensor mode; ECU I2C accesses forwarded to sensor head; stream data for ECU supplied via Ethernet

The TX mode can be used to replace a sensor head which usually consists of the sensor serializer and the sensor chip. In typical applications both chips are configured by the ECU firmware via I2C. Since the VX1161.51 only provides the serializer and does not emulate any other I2C devices, the attached ECU firmware will fail accessing the sensor chip and enter an error state. To overcome this issue, either the ECU firmware must be enabled to cope with the missing sensor (code instrumentation, HIL mode) or the "1TX with sensor" mode in the Channel Group configuration can be used. The 1TX with sensor mode uses a real sensor head. I2C and MFPs/GPIOs are forwarded to the sensor head like in TAP mode but data from the sensor is ignored and data from Ethernet is sent to the ECU.

Modified Registers

The following register values are modified by the VX1161 firmware during initialization:

Register Name	Register Address	Default Value	Modified Value	Description
Serializer				
I2C_DEVICE_ID	0x0000	-	configured value	Device ID is set according configuration
RLMS4	0x1404	0x4B	0x29	Value recommended by GMSL2 user guide V19
RLMSA4	0x14A4	0xBD	0xC8	Value recommended by GMSL2 user guide V19
not documented	0x140A	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
not documented	0x140B	not documented	0x00	Value recommended by GMSL2 user guide V19; undocumented register
REG5	0x0005	0x00	0x80	Bit 7 : LOCK_EN is set; enables the lock output; required by the VX1161.52 firmware to monitor the link status
CTRL0	0x0010	0x11	0x15	Bit 2: REG_ENAB is set; enables the regulator; required to be compatible with the VX1161.52 hardware; value recommended by GMSL2 user guide V19 Bit 6: RESET_LINK is set and cleared; used to hold back link after power up while VX1161.52 FW is configuring system
CTRL2	0x0012	0x00	0x10	Bit 4: REG_MNL is set; enables manual regulator control; required to be compatible with the VX1161.52 hardware; value recommended by GMSL2 user guide V19
REG1	0x0001	0x08	0x08	Bit 4: DIS_REM_CC is set and cleared; used to re-establish link to serializer

GPIO Interconnect

The GPIOs of the serializer are connected to the GPIOs of the VX1161.52. The connections to the TAP deserializer are switched off.

For frame synchronization, one GPIO can be used as frame trigger input of the VX1161.52 to output of a frame (see section Frame Trigger Detector on page 359).

Additionally, the Sensor Trigger Generator can be used to toggle one serializer GPIO per channel. Because one GPIO per channel can be used to trigger the frame transmission, the generator can be used as frame rate generator to the Frame Trigger Detector (see section Frame Trigger Detector on page 359).

The GPIOs of the serializer are connected to the GPIOs of the sensor trigger generator as follows:

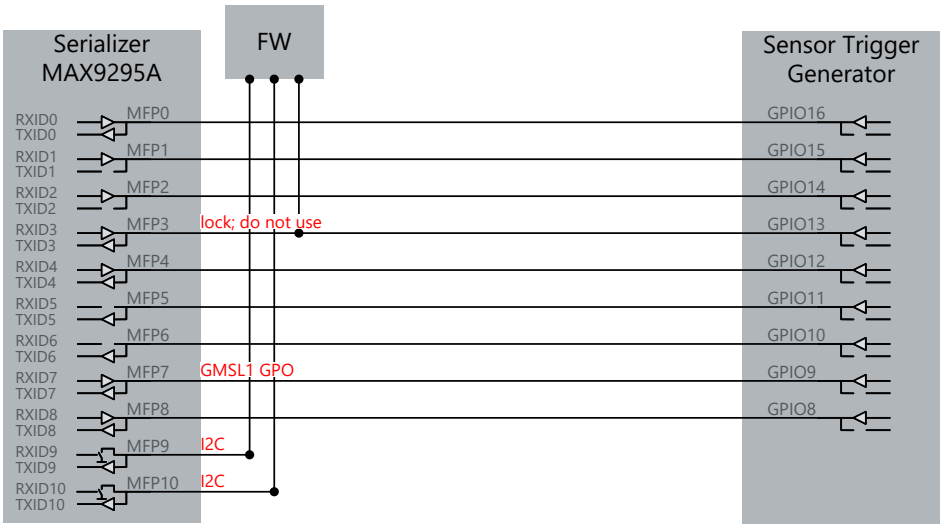


Figure 129: TX MFP architecture; GPIO/MFP connections between TX serializer and Sensor Trigger Generator

Signal of Serializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**
CH2 - MFP0	IO	CH2 - GPIO_16
CH2 - MFP1	O	CH2 - GPIO_15
CH2 - MFP2	O	CH2 - GPIO_14
CH2 - MFP3	IO; LOCK	(CH2 - GPIO_13; do not use)
CH2 - MFP4	IO	CH2 - GPIO_12
CH2 - MFP5	I	CH2 - GPIO_11
CH2 - MFP6	I	CH2 - GPIO_10
CH2 - MFP7	IO	CH2 - GPIO_9
CH2 - MFP8	IO	CH2 -GPIO_8
		-
CH4 - MFP0	IO	CH4 - GPIO_16
CH4 - MFP1	O	CH4 - GPIO_15
CH4 - MFP2	O	CH4 - GPIO_14
CH4 - MFP3	IO; LOCK	(CH4 - GPIO_13; do not use)
CH4 - MFP4	IO	CH4 - GPIO_1
CH4 - MFP5	I	CH4 - GPIO_11

Signal of Serializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**
CH4 - MFP6	I	CH4 - GPIO_10
CH4 - MFP7	IO	CH4 - GPIO_9
CH4 - MFP8	IO	CH4 - GPIO_8

4.5.20.10 Sensor Trigger Generator

The VX1161.52 provides a Sensor Trigger Generator for every channel. This generator can be used to generate a trigger signal to one of the GPIO/MFP pins. If the serializer and deserializer are configured correctly via CASL scripts, this trigger signal will be forwarded to the sensor and triggers the acquisition of data.

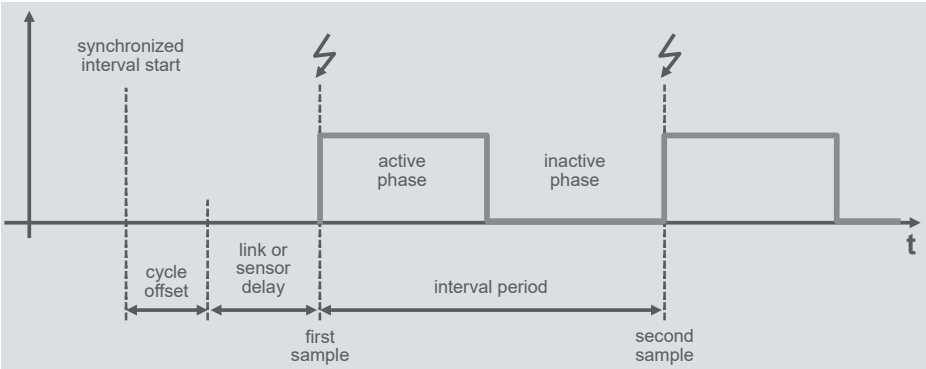


Figure 130: Sensor Trigger Generator with adjustable interval period

The generators of one module use one hardware timer as time base. This timer can be synchronized via PTP between multiple cards. If the PTP service is available, the trigger generation on all channels on all synchronized modules will be synchronous. If the PTP service is not available, only the channels on one module will be synchronous.

The current implementation requires that every VX1161.52 can exchange Ethernet broadcast messages with every other VX1161.52 during measurement start. These broadcast messages are used to negotiate the precise start of the first Frame Trigger period.

If a measurement setup consists of several VX1161 racks, there must be a path the broadcast messages can take to reach all other cards. Since Windows natively does not forward Ethernet broadcast messages from a receiving port to all other Ethernet ports, another connection is needed. To provide that path, an additional Ethernet cable between all VX1161 racks is required. The Ethernet switch inside the VX1161.22 uplink module will distribute those broadcast messages. Any Ethernet port on the VX1161.22 can be used due to the low bandwidth requirement.

The generators start outputting their trigger signal at start of measurement of the receiving application and it will stop generation at measurement stop. As a result, the sensor might not generate data when the measurement is stopped.
When the measurement is started, the time on all channels starts simultaneously and all generators start with their cycle offset times.

The trigger signal has two phases: the active and the inactive phase. Both can be configured in their duration. Together they define the trigger interval period, which defines the frequency of trigger pulses and the frame rate. Trigger generation always starts with the active phase. The Cycle offset parameter can be used to compensate delays in serial links or sensors. For sensors on fast paths, a delay can be added to trigger at the same time as sensors on slow paths.

For sensors requiring falling edges as trigger, the polarity of the output signal can be selected. If not inverted, the active phase starts with a rising edge which is intended to trigger the sensor.

Note
Care must be taken to avoid signal collisions between serializer or deserializer outputs and Sensor Trigger Generator outputs.

The Sensor Trigger Generator is configured via VXconfig and the following parameters:

Setting	Description
Enable	Activates the Sensor Trigger Generator. If disabled, the GPIOs are high impedance.
Cycle Offset	Defines a static delay from the ideal interval start in nano seconds. Can be used to compensate different delays in between several links or sensors.
Active Phase	Defines the duration of the active phase in nano seconds.
Inactive Phase	Defines the duration of the inactive phase in nano seconds.
GPIO output pin to trigger	GPIO pin of the VX1161.52, where the trigger signal shall be output. Check the GPIO/MFP section of your card to see which pins can be used and how they map to deserializer or serializer pins.
GPIO polarity active phase	Can be used to invert the output signal polarity. - checked (default): active phase starts with a rising edge. Active phase is high. Inactive phase is low. - unchecked: active phase starts with a falling edge. Active phase is low. Inactive phase is high.

4.5.20.11 Frame Trigger Detector

The VX1161.52 provides a Frame Trigger Detector for every channel in TX mode. This detector can be used to detect the change of one GPIO/MFP pin and trigger the transmission of the next video frame. This feature allows to synchronize frame transmissions to ECU trigger signals and with this to the ECU time domain.

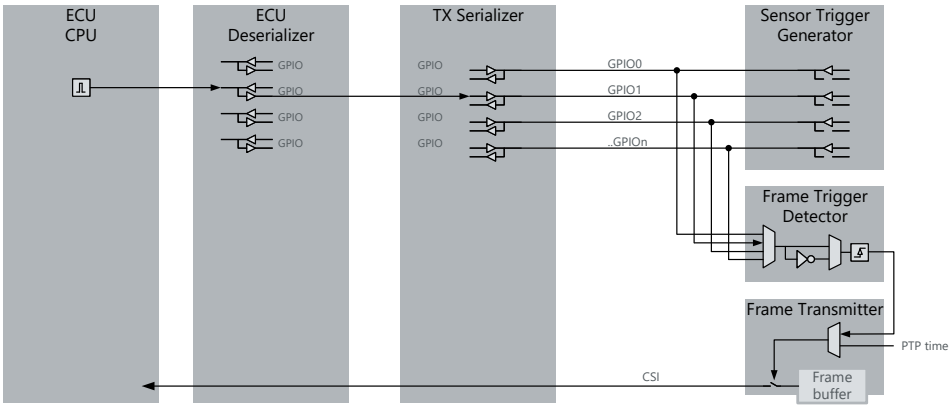


Figure 131: CPU generates frame trigger; Frame Trigger Detector detects edges and triggers the frame transmission

The detector can be fed by one GPIO/MFP of the channel group the channel belongs to. For selecting the active rising or falling edge, chose the polarity of the trigger signal.

Note
Care must be taken to avoid signal collisions between serializer outputs and Sensor Trigger Generator outputs.

The Frame Trigger Detector is configured via VXconfig and the following parameters in the Output Description section.

Setting	Description
GPIO input pin	GPIO pin of the VX1161.52, which shall feed the edge detector. Check the GPIO/MFP section of your card to see which pins can be used and how they map to your serializer pins.
GPIO polarity active phase	Specifies whether rising edges (checked) or falling edges (unchecked) on the given GPIO pin trigger the frame transmission.

4.5.20.12 Required Serializer and Deserializer Settings

The following requirements apply to the serializer and deserializer on the VX1161.52H. The settings for the connected serializer at the sensor and deserializer in the ECU are allowed to differ according GMSL capabilities.

CSI Continuous Clock

Due to internal technical reasons, the CSI clocks on all channels must run in continuous CSI clock mode only. This mode is initially configured by the VX1161.52 firmware. Changing this setting will interrupt reception of data.

CSI Clock Lane Speed

Due to internal reasons, the CSI clock speed of the serializer must be configured via the Channel Group Mode and not be changed. The VX1161.52H firmware will configure this at start-up. If this setting is changed via CASL scripts, data transmission will no longer be possible.

Link Lock Signals on MFP Pins

For detecting link status changes the VX1161 firmware configures MFP1 in TAP deserializer MX96718A and MFP3 in TAP serializer MX96717 to output their Lock signal. If the function of these pins is changed, the TAP will no longer be able to detect unlocks and script execution might be disturbed.

CSI Signal Settings

The CSI lanes of the TAP deserializer MAX96718A and TAP serializer MAX96717 do not require any lane swap, lane inversion or timing adjustments.

4.5.20.13 POC Supply

The VX1161.52H delivers power to connected sensor heads via the coax cable using Power-over-Coax (POC). It has an internal, isolated supply that is adjustable and independent of VX GND. It can source up to 2 W. The voltage can be switched off or adjusted from 5 V ... 12 V using VXconfig.

In TAP mode, the internal supply can be deactivated, allowing the POC supply of the ECU to be forwarded to the sensor head. This gives the ECU control over the sensor's power.

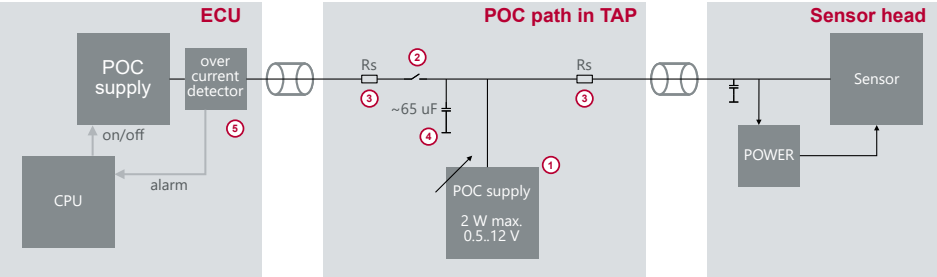


Figure 132: POC structure in VX1161.52H TAP channels

The internal series resistance (Rs) (3) of the POC bypass path is approximately 2 x 2.5 Ohms, and the internal capacitance (4) is approximately 65 µF. This capacitance can cause or increase inrush currents for the ECU that sources power and might trigger overcurrent detection features (5). In such cases, it might be an option to source the sensor from the VX (1) instead of the ECU, but then the ECU would have no control over the sensor power.

Furthermore, the internal VX POC supply can be output to the ECU for use cases in which CSI ingestion PODs require POC supply.

The POC voltage level, generation, and delivery direction are controlled via the POC voltage settings for the TAP_RX and TAP_TX ports in VXconfig as follows:

Note
Sourcing power in TAP mode is supported since VXtools v4.9.

Settings				
POC voltage V_{RX}on TAP RX channel	0V	5...12V	0V	5...12V
POC voltage V_{TX}on TAP TX channel	0V	0V	5...12V	5...12V
Effects				
POC bypass (2)	closed	open	closed	closed
VX POC source (1) voltage	off	V_{RX}	V_{TX}	Min(V_{RX},V_{TX})
Use case	ECU supplies sensor	VX supplies sensor	VX supplies ingest POD	VX supplies sensor and ingest POD

4.5.20.14 Technical Data

Weight	210 g
Power consumption	▶ P_{init}; initialization (~500ms): 500mW ▶ $P_{standby}$; sleep mode: <18 mW ▶ P_{run}; normal operation excl. POC: max.10 W
POC (power over cable)	▶ V_{poc}; voltage sourced by VX1161.52: 0 V, 5...12 V configurable ▶ P_{poc}; maximum power delivered per RX channel; sourced by VX1161.52: 2 W ▶ $V_{PocTapMax}$; voltage forwarded from ECU by TAP: 3...18 V ▶ $I_{PocTapMax}$; max. current forwarded from ECU by TAP: 500 mA ▶ Electrical isolation between TAP_CH1/2, TAP _CH3/4 and chassis max. 50 V ▶ Rpoc; DC resistance of POC forwarding path in TAP: typ. 2.616 Ohm
Minimum power -off time	▶ minimum time between powering-off and re-appying power: >2 s
Start-up time	▶ t_{init}; initialization after power up: ~ 500 ms ▶ $t_{wakeLink}$; wake up after sleep; GMSL2 link stable: < 50 ms, depends on CASL script execution ▶ $t_{wakeReady}$; wake up after sleep; GMSL2 data to Ethernet: < 15 s
Temperature range	-40 °C ... +60 °C
Maximum payload bandwidths	▶ Card to host via Ethernet: 7.2 Gbps net; total for all channels together ▶ Every TAP/ RX channel GMSL2: up to 5.2 Gbps net
Latencies/delays	▶ $t_{delayMFP}$;; additional delay of MFPs; forward, backward delay compensated mode: ~6 µs +- 100ns Additional serializer and deserializer delay: ~ one CSI packet.
Connector	(Single signal) FAKRA Type Z; color: cyan; compatible to all other (single signal) FAKRA connector types; 6 GHz max; 25 mating cycles;
Maximum cable length	8 m, in both RX and TX direction at the same time
Part number	180173

4.5.21 VX1161.52K 2xTAP APHY VA7044/VA7031

The VX1161.52K 2xTAP APHY VA7044/VA7031 is a two-channel card for MIPI A-PHY data streams.

TAP means that a receiver (deserializer) and a transmitter (serializer) are connected back-to-back and the received data is forwarded to the transmitter and duplicated towards the logging system. Each channel contains one Valens VA7044 deserializer and one Valens VA7031 serializer.

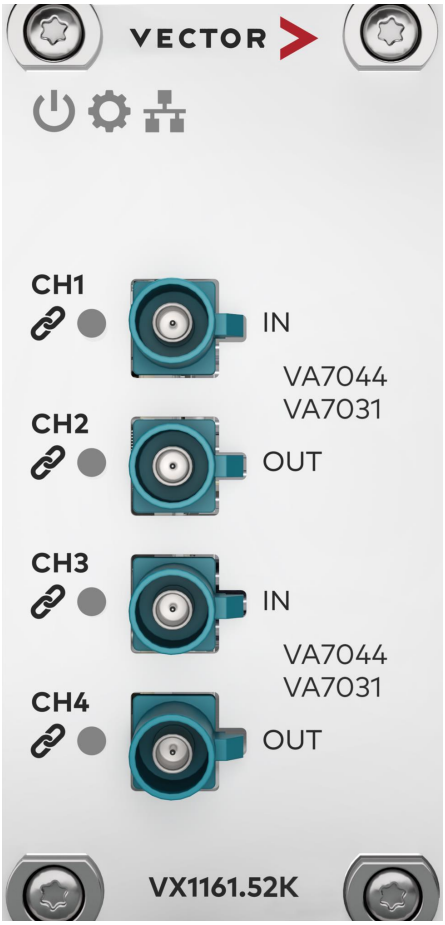


Figure 133: VX1161.52K 2xTAP APHY VA7044/VA7031

The operating mode of the card can be configured either in TAP, RX or TX mode. The VX1161.52K supports binary lossless compression (see section Lossless Compression on page 364).

The card can be configured with VXconfig and updated to new software versions with VXupdate. Initialization of the transceivers and remote sensors can be done via CASL language scripts within VXconfig.

Note
VX1161.52K is supported by VXtools v5.0 and later.

4.5.21.1 Lossless Compression

The video interface supports the hardware-based binary lossless compression of video streams. This compression method typically reduces the required bandwidth for video transmission, logging, and storage by 40 % ... 60 %, depending on the video content. Each channel of the interface can be compressed. The compression pipelines support the following pixel formats: RAW8, RAW10, RAW12, RAW16, and YUV422 8-bit. For other pixel formats, please contact Vector.

Four different compression modes are supported:

- **MobilEye on PC**
  This mode compresses the recording on the computer and requires a MobilEye license.

- **MobilEye on VX**
  This mode enables lossless compression on VX hardware and requires a MobilEye license. It reduces the bandwidth from the VX to the recording computer and offloads the necessary computing power.

- **Vector on VX**
  This mode enables lossless compression on the VX hardware and does not require a license. Only the payload of all received CSI long packets (all data types) is fed into the compression pipelines. The CSI headers and footers are stripped off beforehand. The embedded line payload is also fed into the compressor and reconstructed when decompressed. In this mode, all lines (embedded data, visible, and statistic lines) must be the same length as set in VXconfig. Enabling line length padding in the sensor may be necessary. To ensure good compression efficiency and enable later decoding, it is essential that all lines be the same length as the length set in VXconfig. The sensor may need to pad the embedded lines to the full length. Starting with VXtools V5.1, the interface enforces this.

- **Vector on VX V2**
  This mode has been supported since VXtools V5.1 and enables lossless compression on the VX hardware and does not require a license.

  If selected:

  - The CSI packets specified via CSI leading embedded packets and CSI trailing embedded packets of a frame are stored uncompressed.
  - The CSI headers and footers of all packets are also stored uncompressed.
  - The pixels are losslessly compressed on the interface hardware.
  - All three stored parts are secured via individual CRCs to detect data changes during transport or storage.

Each of the two compression pipelines is capable of compression:

- one stream with up to 350 MPx/s RAW8 ... RAW16
- one stream with up to 170 MPx/s YUV422

Vector provides decompression libraries and tools for Windows and Linux.

4.5.21.2 CRC Algorithm

```
std::vector<uint32_t> VCrc32::sCrcTable;

uint32_t VCrc32::Calculate(const uint8_t* rawData, const uint64_t rawDataS-
ize, const uint8_t bitsPerSample)
{
  uint32_t crc = 0xFFFFFFFF;
  const uint32_t specialWidth = 32 - bitsPerSample;
  const int andMask = (1 << bitsPerSample) - 1;
  uint32_t usedBytes = 0;
  const uint16_t* pData = reinterpret_cast<const uint16_t*>(rawData);

  while (usedBytes < rawDataSize)
  {
    /* XOR-in next input byte into MSB of crc and get this MSB, that's our
new intermediate dividend */
const uint16_t pos = static_cast<uint16_t>(((crc ^ (*pData << spe-
cialWidth)) >> specialWidth) & andMask);
    /* Shift out the MSB used for division per look up table and XOR with
the remainder */

    crc = (crc << bitsPerSample) ^ sCrcTable[pos];
    usedBytes += 2;
    pData++;

    if (rawDataSize % 2 != 0 && usedBytes == rawDataSize - 1)
    {
      uint16_t paddedValue = *pData;
      paddedValue = paddedValue & 0xFF;
      pData = &paddedValue;
    }
  }
  return crc;
}



void VCrc32::GenerateCrc32Table(const uint8_t bitsPerSample)
{
  if (!sCrcTable.empty())
  {
    return ;
  }

  const uint32_t specialWidth = 32 - bitsPerSample;
  const uint32_t elementCount = 1 << bitsPerSample;
  sCrcTable.resize(elementCount);

  for (uint32_t dividend = 0; dividend < elementCount; dividend++)
  {
    uint32_t currentByte = dividend << specialWidth;
    for (uint8_t bit = 0; bit < bitsPerSample; bit++)
    {
       if ((currentByte & 0x80000000) != 0)
      {
        constexpr uint32_t polynomial = 0x04C11DB7;
        currentByte <<= 1;
        currentByte ^= polynomial;
       }
       else
      {
         currentByte <<= 1;
       }
     }
    sCrcTable[dividend] = currentByte;
  }
}
```

4.5.21.3 Connectors

- **CH1...4**
  The VX1161.52K provides FAKRA coaxial connectors of Type Z for connecting data sources and data sinks such as video or radar sensors and ECUs. Type Z connectors are compatible to all other coding types of single signal FAKRA connectors. The connector properties can be found in section Technical Data on page 362.

4.5.21.4 LEDs

The status of the card is displayed by system LEDs in the lid top. The status of every channel is displayed by a channel LED next to every channel connector.

Power

Symbol	Sequence	Description
	R R R R R R R R	Device is powered but Firmware not running. The Power LED will light up red at the beginning of the boot sequence of the Interface Card. ► Please contact Vector VX support if the Power LED stays red for more than 10 seconds. In this case a critical error has occurred.
	G G G G G G G G	Firmware loaded successfully.
	G ☐ G ☐ G ☐ G ☐	Response to ”Identify VX device command” (VXconfig).

System

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Firmware not loaded.
	R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Interface Card, its system components are initialized. If a component fails to initialize properly, the System LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
	G G G G G G G G	Device is in normal operating mode.
	G ☐ G ☐ G ☐ G ☐	Firmware update in progress.

Ethernet

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Backplane link not established.
	R R R R R R R R	Backplane link error.
	G G G G G G G G	Backplane link established.
	G ☐ G ☐ G ☐ G ☐	Backplane link activity.

Channel

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Channel disabled.
	R R R R ☐ ☐ ☐ ☐	Channel enabled, invalid configuration. See VXconsole for details.
	R R R R R R R R	Channel enabled, no link established.
	G G G G G G G G	Channel enabled, link established, no data traffic.
	G ☐ G ☐ G ☐ G ☐	Channel enabled, link established, data traffic.

4.5.21.5 Boot and Link-up-Time

When the VX1161 is connected to power, the streaming card must prepare itself for a fast wake-up. For about t_{init} the power consumption will be P_{init}. After that time, the card enters standby mode and the power consumption is reduced to its minimum $P_{standby}$.

When the VX1161 is woken up (see section Connectors on page 176), the card is already prepared and can establish both ECU and sensor link within $t_{wakeLink}$. After wakeup, the card can provide data to the recorder after $t_{wakeReady}$.

The power consumption during normal operation consist of the maximum power of P_{run} drawn by the card as well as the power consumed by supplied cameras P_{POC}.

4.5.21.6 Channel in TAP Mode

The TAP is designed to be as transparent as possible to the ECU.

If a channel is configured in TAP mode, the incoming APHY data stream is received by the VA7044 TAP deserializer. The data payload (e. g. video or radar frames) is then duplicated. One copy is forwarded to the VA7031 TAP serializer and one copy is forwarded to the recording data processor.

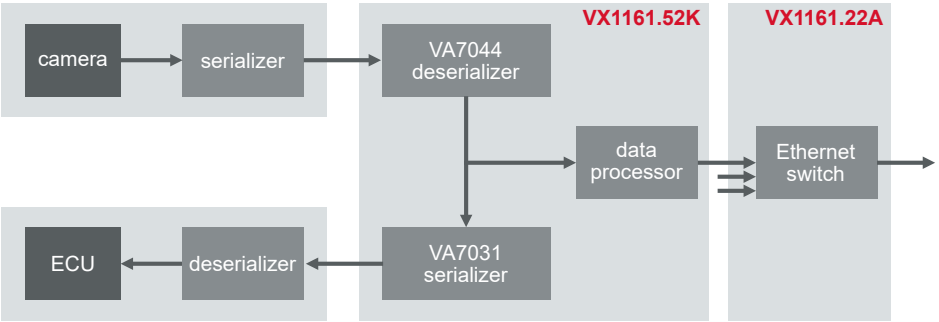


Figure 134: Channel in TAP mode; payload duplication

Control accesses from the ECU to the sensor via the I2C control channel or GPIOs are forwarded from the TAP serializer to the TAP deserializer.

TAP Initialization

The initialization sequence of the sensor and the ECU links follows this chart:

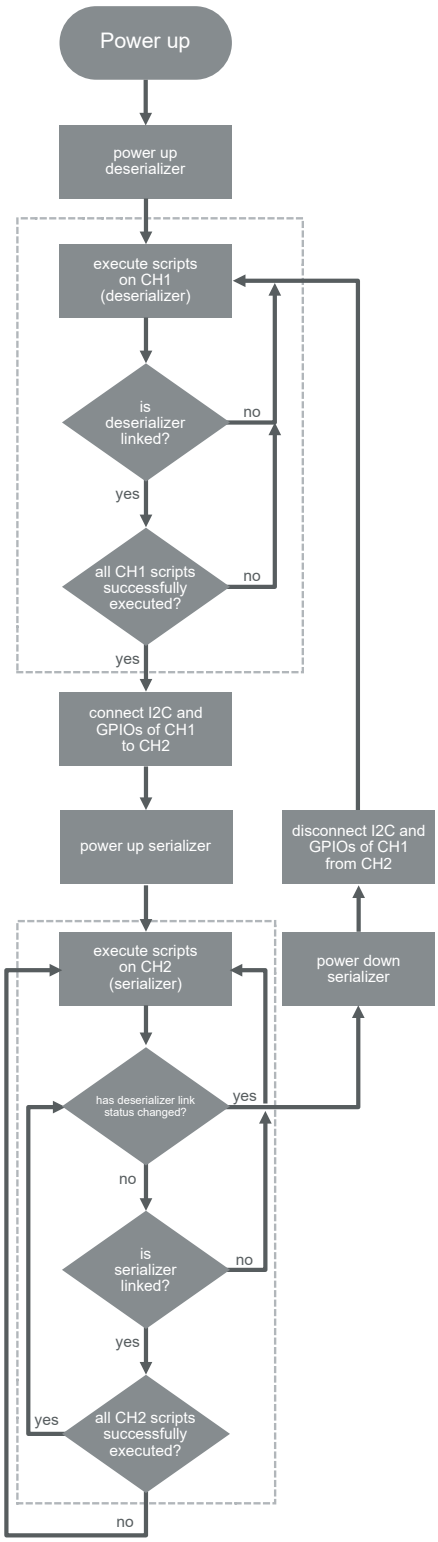


Figure 135: TAP initialization sequence

After power-up, the POC voltage is applied as configured and the deserializer is activated.

Next, the enabled scripts for the TAP RX channel (CH1/CH3) are immediately executed in an initial run (script 1, 2, 3) to configure the TAP deserializer, the sensor serializer and the sensor imager as early as possible.

After the initial script run on the TAP RX channel, the script execution engine waits for a TAP deserializer lock.

When the TAP deserializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the following conditions apply:

1. Script is not configured as "run once" (CASL SetConfig(7,1))
2. OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
3. OR script terminated with a return value ≠ 0 in previous run

If the TAP deserializer link unlocks, the script sequence repeats from the initial TAP deserializer run on, the I2C and GPIO bridges will be opened and the TAP serializer will be deactivated.

When the TAP deserializer locks, the I2C bus and GPIOs of the TAP serializer and TAP deserializer are connected. This allows the ECU to configure and control the sensor (see section GPIO Interconnect).

When the TAP deserializer locks, the TAP serializer is activated and the script execution on the TAP TX channel (CH2/CH4) are executed in an initial run (script 1, 2, 3).

After the initial script run on the TAP TX channel, the script execution engine waits for a TAP serializer lock.

When the TAP serializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the following conditions apply:

1. Script is not configured as "run once" (CASL SetConfig(7,1))
2. OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
3. OR script terminated with a return value ≠ 0 in previous run

If the TAP serializer link unlocks, the sequence repeats from the initial TAP serializer run on.

If the TAP deserializer link unlocks now, the I2C and GPIO bridges will be opened, the TAP serializer will be deactivated and the script sequence repeats from the initial TAP deserializer run on.

I2C Architecture

Without the TAP, the ECU is directly connected to the sensor head. Via the ECU deserializer and the sensor serializer it can configure and control all devices in that path. By inserting the TAP, the ECU deserializer is now connected to the TAP serializer and the sensor serializer is now connected to the TAP deserializer.

To allow the ECU controller to configure and control the sensor serializer and the sensor also in TAP mode, it might be necessary to configure the TAP serializer and the TAP deserializer via CASL scripts.

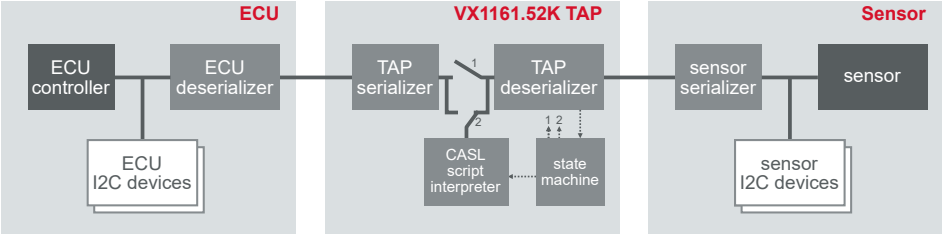


Figure 136: TAP I2C architecture; ECU configures ECU deserializer + TAP serializer + sensor

The I2C address of the TAP deserializer and the TAP serializer can be configured in VXconfig. Both chips start with their hardware strapped address and are then overwritten by the VX1161 firmware with the address configured in VXconfig. Additionally, the SetConfig(8,x) CASL command can be used to change the I2C address dynamically.

The sensor serializer I2C address is defined by its hardware straps. It can be changed via CASL scripts and must not be the same as the TAP serializer address.

The sensor I2C address is defined by the sensor itself. Accesses by the ECU will transparently be forwarded through the sensor link.

The additional TAP link will cause additional transmission delays (via clock stretching) to any sensor register access. These delays will reduce the effective control channel bandwidth and could affect system timings.

Detection of TAP by ECU Firmware for Measurement Mode

In cases where the ECU needs to know whether a TAP is present, CASL scripts can be used to manipulate unused spare default register values in the CAM serializer. The ECU can then read the register and check whether it contains the default value (no TAP) or the CASL script value (TAP node).

GPIO Interconnect

For forwarding GPIOs (general purpose input output pins) from the ECU to the sensor the TAP deserializer and the sensor serializer registers must be configured properly. This can be done via CASL scripts. This also allows static GPIO levels to be configured inside the deserializer..

Additionally, the VX1161.52K has a Sensor Trigger Generator which can generate synchronized trigger pulses on one GPIO pin per channel (see section Sensor Trigger Generator on page 357).

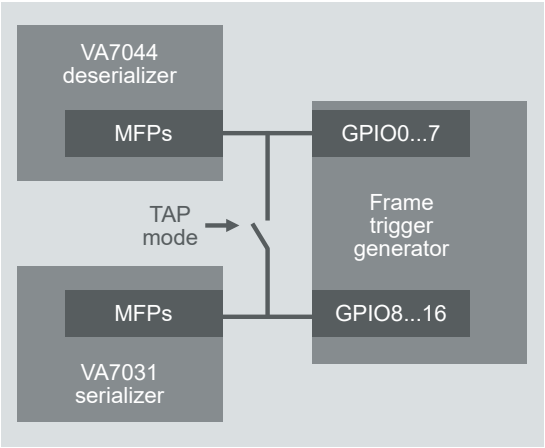


Figure 137: In TAP mode, some GPIOs of serializer and deserializer are connected as long as the deserializer is APHY locked to the remote serializer

Note
Care must be taken to avoid signal collisions between GPIO outputs and Sensor Trigger Generator outputs.

Note
GPIO_0 of the serializer must stay configured as INT output. It will be connected to GPO_0 of the deserializer in TAP mode. This function will be configured by the card firmware at boot time. If the function of these pins is changed, the TAP will no longer be able to detect unlocks and script execution might be disturbed.

Forwarding of analog GPIO features is not supported.

In cases where sensor triggers are transferred via GPIOs, it must be considered that the sensor links of two TAP links are not synchronized. Therefore, a skew between the two independent VX1161.52K TAP channelsmust be assumed.

The GPIOs of the TAP serializer are bidirectionally connected to the GPIOs of the deserializer and the Sensor Trigger Generator GPIOs as follows:

Signal of Serializer		Connected to Signal of Deserializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**	**Direction**	**Name**
CH2 - GPO_0	O; INT=LOCK	CH1 - GPIO_0	IO	(CH2 - GPIO_16 = CH1 - GPIO_0 do not use)
CH2 - GPO_1	O	CH1 - GPIO_2	IO	CH2 - GPIO_15 = CH1 - GPIO_1
CH2 - GPO_2	O	CH1 - GPIO_3	IO	CH2 - GPIO_14 = CH1 - GPIO_2
CH2 - GPIO_0	IO	CH1 - GPIO_4	IO	CH2 - GPIO_13 = CH1 - GPIO_3
CH2 - GPIO_1	IO	CH1 - GPIO_6	IO	CH2 - GPIO_12 = CH1 - GPIO_4
CH2 - GPIO_3	IO	CH1 - GPIO_8	IO	CH2 - GPIO_11 = CH1 - GPIO_5
CH2 - GPIO_6	IO	CH1 - GPIO_9	IO	CH2 - GPIO_10 = CH1 - GPIO_6
CH2 - GPIO_7	IO	CH1 - GPIO_11	IO	CH2 -GPIO_9 = CH1 -GPIO_7
CH2 - GPIO_8	IO	CH1 - GPIO_1	IO	CH2 -GPIO_8
CH4 - GPO_0	O; INT=LOCK	CH3 - GPIO_0	IO	(CH4 - GPIO_16 = CH3 - GPIO_0 do not use)
CH4 - GPO_1	O	CH3 - GPIO_2	IO	CH4 - GPIO_15 = CH3 - GPIO_1
CH4 - GPO_2	O	CH3 - GPIO_3	IO	CH4 - GPIO_14 = CH3 - GPIO_2
CH4 - GPIO_0	IO	CH3 - GPIO_4	IO	CH4 - GPIO_13 = CH3 - GPIO_3
CH4 - GPIO_1	IO	CH3 - GPIO_6	IO	CH4 - GPIO_12 = CH3 - GPIO_4
CH4 - GPIO_3	IO	CH3 - GPIO_8	IO	CH4 - GPIO_11 = CH3 - GPIO_3
CH4 - GPIO_6	IO	CH3 - GPIO_9	IO	CH4 - GPIO_10 = CH3 - GPIO_6
CH4 - GPIO_7	IO	CH3 - GPIO_11	IO	CH4 - GPIO_9 = CH3 - GPIO_7
CH4 - GPIO_8	IO	CH3 - GPIO_1	IO	CH4 -GPIO_8

Signal mapping of TAP deserializer GPIOs to trigger generator GPIOs; directions with view of deserializer (IO_OD: Input+open drain output); INT=LOCK: Do not use. Signals required internally.

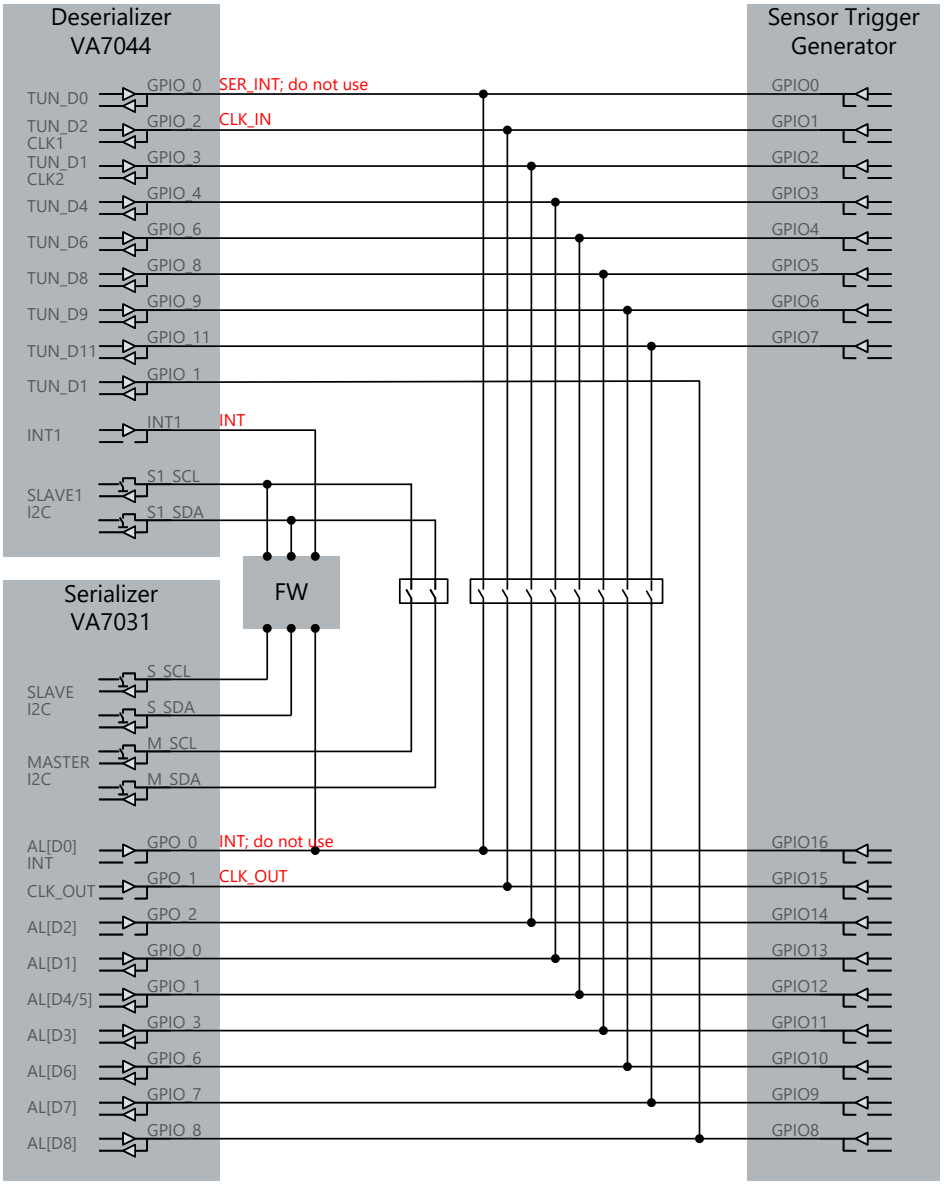


Figure 138: TAP GPIO architecture; GPIO connections between TAP deserializer, TAP serializer and Sensor Trigger Generator; switches are closed when TAP deserializer link is stable

4.5.21.7 Channel in RX Mode

If a channel is configured in RX mode, the incoming A-PHY data stream is received by the VA7044 deserializer and forwarded to the recording data processing.

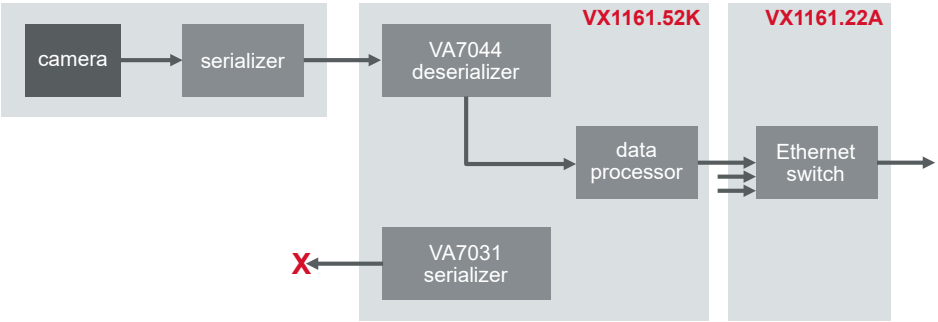


Figure 139: TAP channel in RX mode; payload reception

Initialization

The initialization sequence of the sensor link follows this chart:

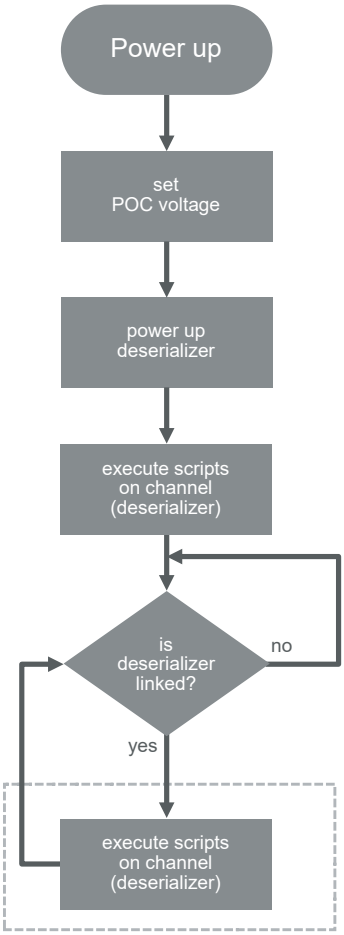


Figure 140: RX initialization sequence

After power-up, the POC voltage is applied as configured and the deserializer is activated.

Next, the enabled scripts for the RX channel are executed immediately (script 1, 2, 3) to configure the deserializer, sensor serializer and sensor as early as possible.

After the initial run, the script execution engine waits for a deserializer lock.

When the deserializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the follwing conditions apply:

- Script is not configured as "run once" (CASL SetConfig(7,1))
- OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
- OR script terminated with a return value ≠ 0 in previous run

If the deserializer link unlocks, the script sequence repeates from the inital deserializer run on.

I2C Architecture

In RX mode, the TAP deserializer, sensor serializer and sensor need to be configured via CASL scripts. A dynamic sensor reconfiguration is currently not supported.

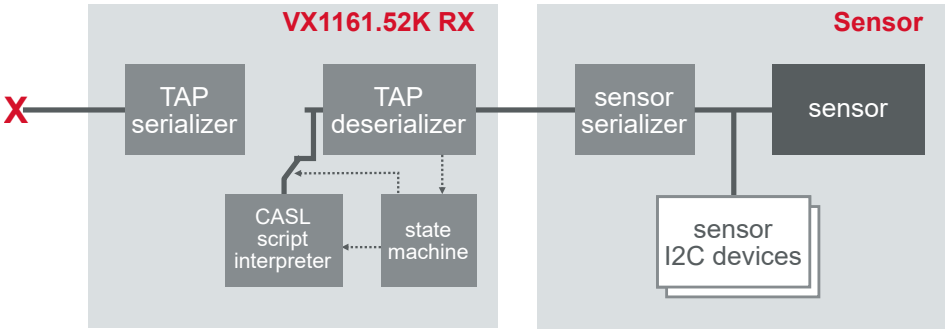


Figure 141: RX mode; I2C architecture; CASL scripts configure deserializer, serializer and sensor

The I2C address of the TAP deserializer can be configured in VXconfig. The chip starts with its hardware strapped address and is then overwritten by the VX1161 firmware with the address configured in VXconfig. Additionally, the SetConfig(8,x) CASL command can be used to change the I2C address dynamically.

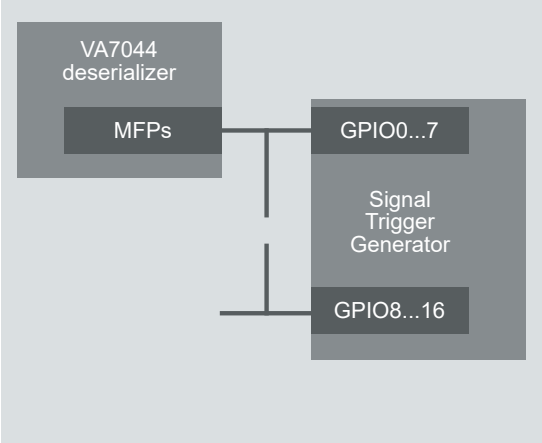
The sensor serializer I2C address is defined by its hardware strapped pins and could be changed via CASL scripts.

The sensor I2C address is defined by the sensor itself. CASL scripts can be used to configure the sensor via I2C transactions. To enable I2C accesses to the sensor, deserializer and serializer registers might have to be set properly in advance.

GPIO Interconnect

The VX1161.52K has a Sensor Trigger Generator which can generate synchronized trigger pulses on one GPIO pin per channel.

For forwarding GPIOs (general purpose input output pins) from the Sensor Trigger Generator to the sensor, the TAP deserializer and the sensor serializer registers must be configured properly. This can be done via CASL scripts. This also allows static GPIO levels to be configured inside the deserializer.



Note
Care must be taken to avoid signal collisions between GPIO outputs and Sensor Trigger Generator outputs.

The GPIOs of the deserializer are connected to the GPIOs of the sensor trigger generator as follows:

Signal of Deserializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**
CH1 - GPIO_0	IO	CH1 - GPIO_0
CH1 - GPIO_1	IO	CH**2** - GPIO_8 permanently connected to serializer GPIO_8
CH1 - GPIO_2	IO	CH1 - GPIO_1
CH1 - GPIO_3	IO	CH1 - GPIO_2
CH1 - GPIO_4	IO	CH1 - GPIO_3
CH1 - GPIO_6	IO	CH1 - GPIO_4
CH1 - GPIO_8	IO	CH1 - GPIO_5
CH1 - GPIO_9	IO	CH1 - GPIO_6
CH1 - GPIO_11	IO	CH1 - GPIO_7
CH3 - GPIO_0	IO	CH3 - GPIO_0
CH3 - GPIO_1	IO	CH**4** - GPIO_8 permanently connected to serializer GPIO_8
CH3 - GPIO_2	IO	CH3 - GPIO_1
CH3 - GPIO_3	IO	CH3 - GPIO_2
CH3 - GPIO_4	IO	CH3 - GPIO_3
CH3 - GPIO_6	IO	CH3 - GPIO_4
CH3 - GPIO_8	IO	CH3 - GPIO_5

Signal of Deserializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**
CH3 - GPIO_9	IO	CH3 - GPIO_6
CH3 - GPIO_11	IO	CH3 - GPIO_7

Signal mapping of deserializer GPIOs to trigger generator GPIOs; directions with view of deserializer.

4.5.21.8 Sensor Trigger Generator

The VX1161.51/52 provides a Sensor Trigger Generator for every channel. This generator can be used to generate a trigger signal to one of the GPIO/MFP pins. If the serializer and deserializer are configured correctly via CASL scripts, this trigger signal will be forwarded to the sensor and triggers the acquisition of data.

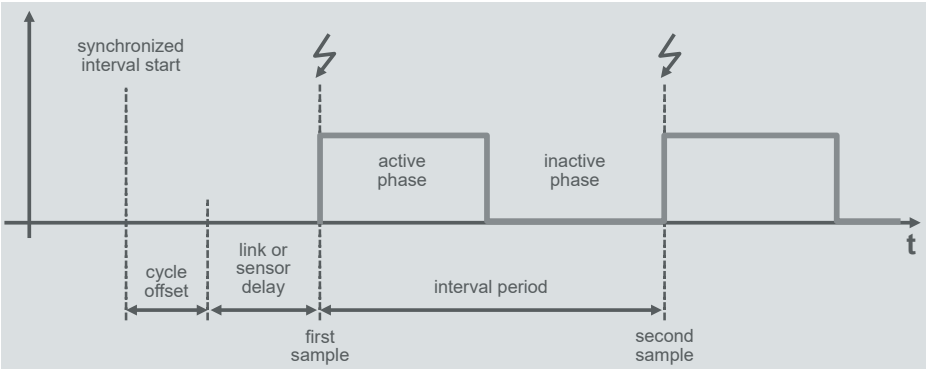


Figure 142: Sensor Trigger Generator with adjustable interval period

The generators of one module use one hardware timer as time base. This timer can be synchronized via PTP between multiple cards. If the PTP service is available, the trigger generation on all channels on all synchronized modules will be synchronous. If the PTP service is not available, only the channels on one module will be synchronous.

The current implementation requires that every VX1161.51/52 can exchange Ethernet broadcast messages with every other VX1161.51/52 during measurement start. These broadcast messages are used to negotiate the precise start of the first Frame Trigger period.

If a measurement setup consists of several VX1161 racks, there must be a path the broadcast messages can take to reach all other cards. Since Windows natively does not forward Ethernet broadcast messages from a receiving port to all other Ethernet ports, another connection is needed. To provide that path, an additional Ethernet cable between all VX1161 racks is required. The Ethernet switch inside the VX1161.22 uplink module will distribute those broadcast messages. Any Ethernet port on the VX1161.22 can be used due to the low bandwidth requirement.

The generators start outputting their trigger signal at start of measurement of the receiving application and it will stop generation at measurement stop. As a result, the sensor might not generate data when the measurement is stopped.
When the measurement is started, the time on all channels starts simultaneously and all generators start with their cycle offset times.

The trigger signal has two phases: the active and the inactive phase. Both can be configured in their duration. Together they define the trigger interval period, which defines the frequency of trigger pulses and the frame rate. Trigger generation always starts with the active phase. The Cycle offset parameter can be used to compensate delays in serial links or sensors. For sensors on fast paths, a delay can be added to trigger at the same time as sensors on slow paths.

For sensors requiring falling edges as trigger, the polarity of the output signal can be selected. If not inverted, the active phase starts with a rising edge which is intended to trigger the sensor.

Note
Care must be taken to avoid signal collisions between serializer or deserializer outputs and Sensor Trigger Generator outputs.

The Sensor Trigger Generator is configured via VXconfig and the following parameters:

Setting	Description
Enable	Activates the Sensor Trigger Generator. If disabled, the GPIOs are high impedance.
Cycle Offset	Defines a static delay from the ideal interval start in nano seconds. Can be used to compensate different delays in between several links or sensors.
Active Phase	Defines the duration of the active phase in nano seconds.
Inactive Phase	Defines the duration of the inactive phase in nano seconds.
GPIO output pin to trigger	GPIO pin of the VX1161.51/52, where the trigger signal shall be output. Check the GPIO/MFP section of your card to see which pins can be used and how they map to deserializer or serializer pins.
GPIO polarity active phase	Can be used to invert the output signal polarity. - checked (default): active phase starts with a rising edge. Active phase is high. Inactive phase is low. - unchecked: active phase starts with a falling edge. Active phase is low. Inactive phase is high.

4.5.21.9 Frame Trigger Detector

The VX1161.51/52 provides a Frame Trigger Detector for every channel in TX mode. This detector can be used to detect the change of one GPIO/MFP pin and trigger the transmission of the next video frame. This feature allows to synchronize frame transmissions to ECU trigger signals and with this to the ECU time domain.

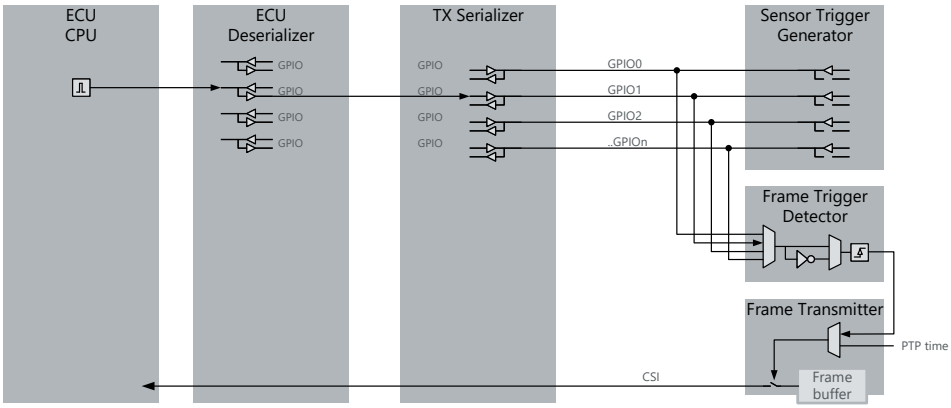


Figure 143: CPU generates frame trigger; Frame Trigger Detector detects edges and triggers the frame transmission

The detector can be fed by one GPIO/MFP of the channel group the channel belongs to. For selecting the active rising or falling edge, chose the polarity of the trigger signal.

Note
Care must be taken to avoid signal collisions between serializer outputs and Sensor Trigger Generator outputs.

The Frame Trigger Detector is configured via VXconfig and the following parameters in the Output Description section.

Setting	Description
GPIO input pin	GPIO pin of the VX1161.51/52, which shall feed the edge detector. Check the GPIO/MFP section of your card to see which pins can be used and how they map to your serializer pins.
GPIO polarity active phase	Specifies whether rising edges (checked) or falling edges (unchecked) on the given GPIO pin trigger the frame transmission.

4.5.21.10 Required Serializer and Deserializer Settings

The following requirements apply to the serializer and deserializer on the VX1161.52K. The settings for the connected serializer at the sensor and deserializer in the ECU are allowed to differ according APHY capabilities.

CSI Continuous Clock

Due to internal technical reasons, the CSI clocks on all channels must run in continuous CSI clock mode only. This mode is initially configured by the VX1161.52 firmware. Changing this setting will interrupt reception of data.

CSI Clock Lane Speed

Due to internal reasons, the CSI clock speed of the serializer must be configured via the Channel Group Mode and not be changed. The VX1161.52K firmware will configure this at start-up. If this setting is changed via CASL scripts, data transmission will no longer be possible.

Link Lock Signals on GPIO Pins

For detecting link status changes, the VX1161 firmware configures GPO_0 in TAP serializer VA7031 to output the INT signal as Lock indicator. This signal is connected to GPIO_0 of the TAP deserializer VA7044 which must be kept as input. If the function of these pins is changed, the TAP will no longer be able to detect unlocks and script execution might be disturbed.

CSI Signal Settings

The CSI lanes of the TAP deserializer VA7044 and TAP serializer VA7031 do not require any lane swap, lane inversion or timing adjustments.

4.5.21.11 POC Supply

The VX1161.52K delivers power to connected sensor heads via the coax cable using Power-over-Coax (POC). It has an internal, isolated supply that is adjustable and independent of VX GND. It can source up to 2 W. The voltage can be switched off or adjusted from 5 V ... 12 V using VXconfig.

In TAP mode, the internal supply can be deactivated, allowing the POC supply of the ECU to be forwarded to the sensor head. This gives the ECU control over the sensor's power.

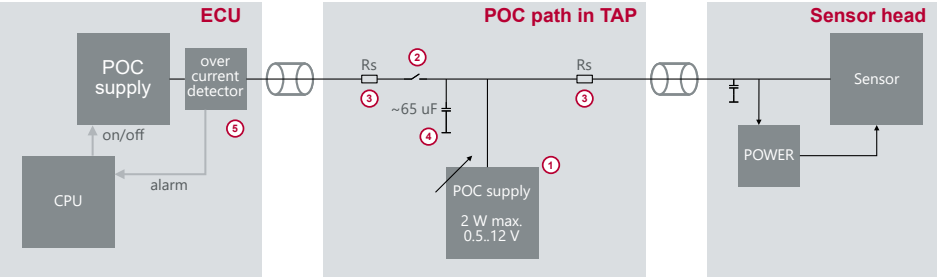


Figure 144: POC structure in VX1161.52K TAP channels

The internal series resistance (Rs) (3) of the POC bypass path is approximately 2 x 2.5 Ohms, and the internal capacitance (4) is approximately 65 µF. This capacitance can cause or increase inrush currents for the ECU that sources power and might trigger overcurrent detection features (5). In such cases, it might be an option to source the sensor from the VX (1) instead of the ECU, but then the ECU would have no control over the sensor power.

Furthermore, the internal VX POC supply can be output to the ECU for use cases in which CSI ingestion PODs require POC supply.

The POC voltage level, generation, and delivery direction are controlled via the POC voltage settings for the TAP_RX and TAP_TX ports in VXconfig as follows:

Note
Sourcing power in TAP mode is supported since VXtools v4.9.

Settings				
POC voltage V_{RX}on TAP RX channel	0V	5...12V	0V	5...12V
POC voltage V_{TX}on TAP TX channel	0V	0V	5...12V	5...12V
Effects				
POC bypass (2)	closed	open	closed	closed
VX POC source (1) voltage	off	V_{RX}	V_{TX}	Min(V_{RX},V_{TX})
Use case	ECU supplies sensor	VX supplies sensor	VX supplies ingest POD	VX supplies sensor and ingest POD

4.5.21.12 Technical Data

Weight	210 g
Power consumption	▶ P_{init}; initialization (~500ms): 500mW ▶ $P_{standby}$; sleep mode: <18 mW ▶ P_{run}; normal operation excl. POC: max.10 W
POC (power over cable)	▶ V_{poc}; voltage sourced by VX1161.52: 0 V, 5...12 V configurable ▶ P_{poc}; maximum power delivered per RX channel; sourced by VX1161.52: 2 W ▶ $V_{PocTapMax}$; voltage forwarded from ECU by TAP: 3...18 V ▶ $I_{PocTapMax}$; max. current forwarded from ECU by TAP: 500 mA ▶ Electrical isolation between TAP_CH1/2, TAP _CH3/4 and chassis max. 50 V ▶ Rpoc; DC resistance of POC forwarding path in TAP: typ. 2.616 Ohm
Minimum power -off time	▶ minimum time between powering-off and re-appying power: >2 s
Start-up time	▶ t_{init}; initialization after power up: ~ 500 ms ▶ $t_{wakeLink}$; wake up after sleep; APHY link stable: < 50 ms, depends on CASL script execution ▶ $t_{wakeReady}$; wake up after sleep; APHY data to Ethernet: < 15 s
Temperature range	-40 °C ... +60 °C
Maximum payload bandwidths	▶ Card to host via Ethernet: 7.2 Gbps net; total for all channels together ▶ Every TAP/ RX channel APHY: up to 6.0 Gbps net
Latencies/delays	▶ $t_{delayGPIO;}$; additional delay of GPIOs; forward, back-ward delay Additional serializer and deserializer delay: ~ one CSI packet.
Connector	(Single signal) FAKRA Type Z; color: cyan; compatible to all other (single signal) FAKRA connector types; 6 GHz max; 25 mating cycles;
Maximum cable length	8 m, in both RX and TX direction at the same time
Part number	180241

4.5.22 VX1161.52L 2xTAP GVIF CXD4966/CXD4967

The VX1161.52L 2xTAP GVIF3 CXD4966/CXD4967 is a two-channel card (CH1...CH4) for GVIF3 data streams with tunneling mode.

TAP means that incoming data from a sensor is forwarded to the ECU and duplicated for the logging system. Both channels contain a SONY CXD4966 deserializer. This device uses the CXD4966's bypass feature to duplicate the sensor data transparently.

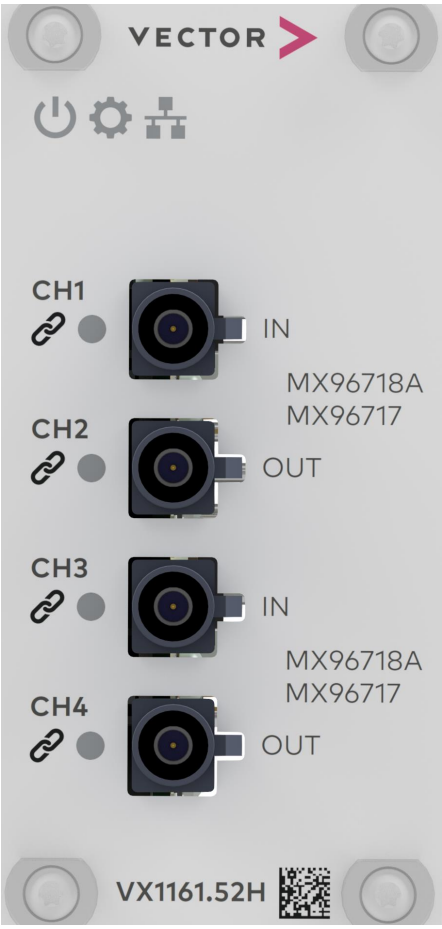


Figure 145: VX1161.52L 2xTAP GVIF3 CXD4966/CXD4967

The operating mode of the card can be configured either in TAP and RX mode. The VX1161.52L supports binary lossless compression (see section Lossless Compression on page 387).

The card can be configured with VXconfig and updated to new software versions with VXupdate. Initialization of the transceivers and remote sensors can be done via CASL language scripts within VXconfig.

Note
VX1161.52L is supported by VXtools v5.1 and later.

4.5.22.1 Lossless Compression

The video interface supports the hardware-based binary lossless compression of video streams. This compression method typically reduces the required bandwidth for video transmission, logging, and storage by 40 % ... 60 %, depending on the video content. Each channel of the interface can be compressed. The compression pipelines support the following pixel formats: RAW8, RAW10, RAW12, RAW16, and YUV422 8-bit. For other pixel formats, please contact Vector.

Four different compression modes are supported:

- **MobilEye on PC**
  This mode compresses the recording on the computer and requires a MobilEye license.

- **MobilEye on VX**
  This mode enables lossless compression on VX hardware and requires a MobilEye license. It reduces the bandwidth from the VX to the recording computer and offloads the necessary computing power.

- **Vector on VX**
  This mode enables lossless compression on the VX hardware and does not require a license. Only the payload of all received CSI long packets (all data types) is fed into the compression pipelines. The CSI headers and footers are stripped off beforehand. The embedded line payload is also fed into the compressor and reconstructed when decompressed. In this mode, all lines (embedded data, visible, and statistic lines) must be the same length as set in VXconfig. Enabling line length padding in the sensor may be necessary. To ensure good compression efficiency and enable later decoding, it is essential that all lines be the same length as the length set in VXconfig. The sensor may need to pad the embedded lines to the full length. Starting with VXtools V5.1, the interface enforces this.

- **Vector on VX V2**
  This mode has been supported since VXtools V5.1 and enables lossless compression on the VX hardware and does not require a license.

  If selected:

  - The CSI packets specified via CSI leading embedded packets and CSI trailing embedded packets of a frame are stored uncompressed.
  - The CSI headers and footers of all packets are also stored uncompressed.
  - The pixels are losslessly compressed on the interface hardware.
  - All three stored parts are secured via individual CRCs to detect data changes during transport or storage.

Each of the two compression pipelines is capable of compression:

- one stream with up to 350 MPx/s RAW8 ... RAW16
- one stream with up to 170 MPx/s YUV422

Vector provides decompression libraries and tools for Windows and Linux.

4.5.22.2 CRC Algorithm

```
std::vector<uint32_t> VCrc32::sCrcTable;

uint32_t VCrc32::Calculate(const uint8_t* rawData, const uint64_t rawDataS-
ize, const uint8_t bitsPerSample)
{
  uint32_t crc = 0xFFFFFFFF;
  const uint32_t specialWidth = 32 - bitsPerSample;
  const int andMask = (1 << bitsPerSample) - 1;
  uint32_t usedBytes = 0;
  const uint16_t* pData = reinterpret_cast<const uint16_t*>(rawData);

  while (usedBytes < rawDataSize)
  {
    /* XOR-in next input byte into MSB of crc and get this MSB, that's our
new intermediate dividend */
const uint16_t pos = static_cast<uint16_t>(((crc ^ (*pData << spe-
cialWidth)) >> specialWidth) & andMask);
    /* Shift out the MSB used for division per look up table and XOR with
the remainder */

    crc = (crc << bitsPerSample) ^ sCrcTable[pos];
    usedBytes += 2;
    pData++;

    if (rawDataSize % 2 != 0 && usedBytes == rawDataSize - 1)
    {
      uint16_t paddedValue = *pData;
      paddedValue = paddedValue & 0xFF;
      pData = &paddedValue;
    }
  }
  return crc;
}



void VCrc32::GenerateCrc32Table(const uint8_t bitsPerSample)
{
  if (!sCrcTable.empty())
  {
    return ;
  }

  const uint32_t specialWidth = 32 - bitsPerSample;
  const uint32_t elementCount = 1 << bitsPerSample;
  sCrcTable.resize(elementCount);

  for (uint32_t dividend = 0; dividend < elementCount; dividend++)
  {
    uint32_t currentByte = dividend << specialWidth;
    for (uint8_t bit = 0; bit < bitsPerSample; bit++)
    {
       if ((currentByte & 0x80000000) != 0)
      {
        constexpr uint32_t polynomial = 0x04C11DB7;
        currentByte <<= 1;
        currentByte ^= polynomial;
       }
       else
      {
         currentByte <<= 1;
       }
     }
    sCrcTable[dividend] = currentByte;
  }
}
```

4.5.22.3 Connectors

- **CH1...4**
  The VX1161.52L provides FAKRA coaxial connectors of Type Z for connecting data sources and data sinks such as video or radar sensors and ECUs. Type Z connectors are compatible to all other coding types of single signal FAKRA connectors. The connector properties can be found in section Technical Data on page 406.

4.5.22.4 LEDs

The status of the card is displayed by system LEDs in the lid top. The status of every channel is displayed by a channel LED next to every channel connector.

Power

Symbol	Sequence	Description
	R R R R R R R R	Device is powered but Firmware not running. The Power LED will light up red at the beginning of the boot sequence of the Interface Card. ► Please contact Vector VX support if the Power LED stays red for more than 10 seconds. In this case a critical error has occurred.
	G G G G G G G G	Firmware loaded successfully.
	G ☐ G ☐ G ☐ G ☐	Response to ”Identify VX device command” (VXconfig).

System

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Firmware not loaded.
	R R R R R R R R	Device is in update-only operating mode. During the boot sequence of the Interface Card, its system components are initialized. If a component fails to initialize properly, the System LED stays red. ► Please perform a FPGA and firmware update as described in section FPGA and Firmware Update on page 53. If that condition persists after the update, please contact Vector VX support.
	G G G G G G G G	Device is in normal operating mode.
	G ☐ G ☐ G ☐ G ☐	Firmware update in progress.

Ethernet

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Backplane link not established.
	R R R R R R R R	Backplane link error.
	G G G G G G G G	Backplane link established.
	G ☐ G ☐ G ☐ G ☐	Backplane link activity.

Channel

Symbol	Sequence	Description
	☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐	Channel disabled.
	R R R R ☐ ☐ ☐ ☐	Channel enabled, invalid configuration. See VXconsole for details.
	R R R R R R R R	Channel enabled, no link established.
	G G G G G G G G	Channel enabled, link established, no data traffic.
	G ☐ G ☐ G ☐ G ☐	Channel enabled, link established, data traffic.

4.5.22.5 Boot and Link-up-Time

When the VX1161 is connected to power, the streaming card must prepare itself for a fast wake-up. For about t_{init} the power consumption will be P_{init}. After that time, the card enters standby mode and the power consumption is reduced to its minimum $P_{standby}$.

When the VX1161 is woken up (see section Connectors on page 176), the card is already prepared and can establish both ECU and sensor link within $t_{wakeLink}$. After wakeup, the card can provide data to the recorder after $t_{wakeReady}$.

The power consumption during normal operation consist of the maximum power of P_{run} drawn by the card as well as the power consumed by supplied cameras P_{POC}.

4.5.22.6 Changing the Mode of Operation

The VX1161.52L can be used in one of the following use cases:

- Video RX/TAP (video reception / logging)

In these modes, sensors or ECUs with GVIF3 transceivers can be directly attached to log or reply streams of data.

The mode of operation can be changed in VXconfig on the Streaming Interface Card dialog.

4.5.22.7 Channel in TAP Mode

The TAP is designed to be as transparent as possible to the ECU.

If a channel is configured in TAP mode, the incoming GVIF3 data stream is received by the CXD4966 TAP deserializer. The data payload (e. g. video or radar frames) is then duplicated. The data payload (e. g., video or radar frames) is forwarded by the GVIF bypass feature of the TAP deserializer and unloaded/-duplicated toward the recording data processor.

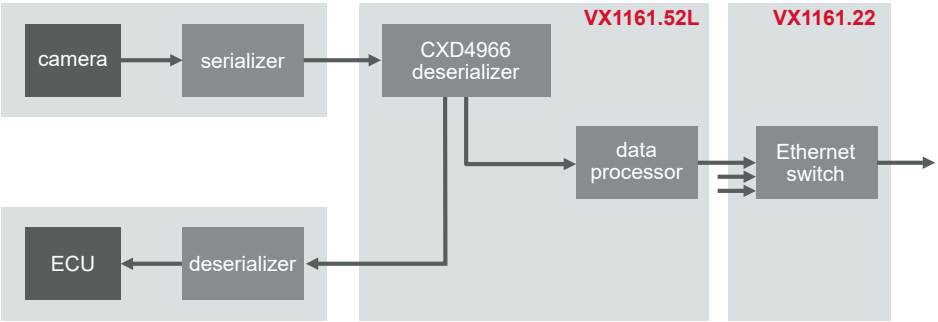


Figure 146: Channel in TAP mode; payload duplication by the deserializer bypass feature

Control accesses from the ECU to the sensor via the I2C control channel or GPIOs are forwarded via the bypass feature of the TAP deserializer.

TAP Initialization

The initialization sequence of the sensor and the ECU links follows this chart:

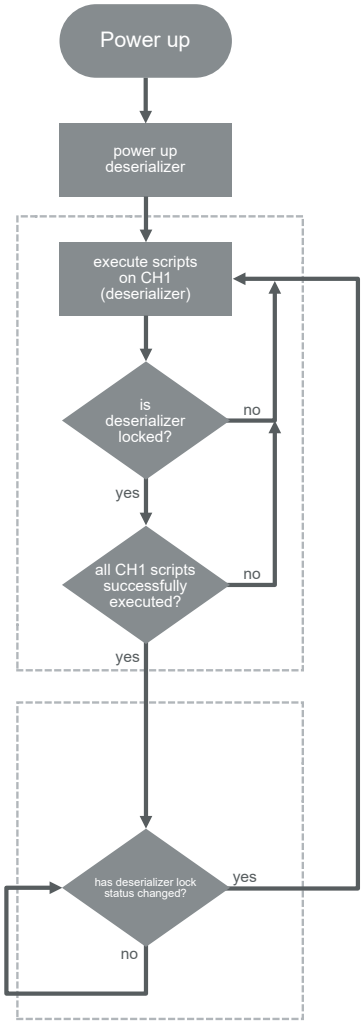


Figure 147: TAP initialization sequence

After power-up, the POC voltage is applied as configured and the deserializer is activated.

Next, the enabled scripts for the TAP RX channel (CH1/CH3) are immediately executed in an initial run (script 1, 2, 3) to configure the TAP deserializer, the sensor serializer and the sensor imager as early as possible.

After the initial script run on the TAP RX channel, the script execution engine waits for a TAP deserializer lock.

When the TAP deserializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the following conditions apply:

1. Script is not configured as "run once" (CASL SetConfig(7,1))
2. OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
3. OR script terminated with a return value ≠ 0 in previous run

If the TAP deserializer link unlocks, the script sequence repeats from the initial TAP deserializer run on.

I2C Architecture

Without the TAP, the ECU is directly connected to the sensor head. Via the ECU deserializer and the sensor serializer it can configure and control all devices in that path. Inserting the TAP via the GVIF bypass feature does not change this. The TAP deserializer is fully transparent between ECU accesses and the sensor serializer.

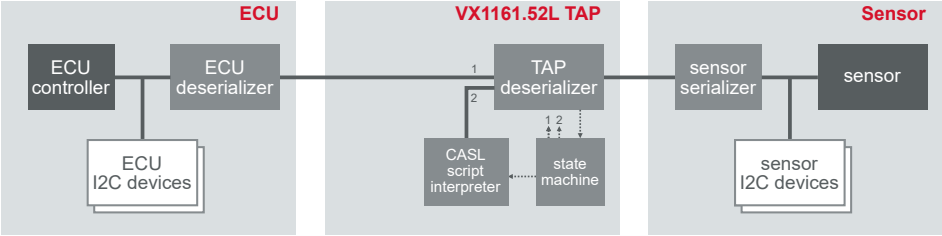


Figure 148: TAP I2C architecture; ECU configures ECU deserializer + sensor serializer + sensor; CASL scripts configure TAP deserializer

The I2C address of the TAP deserializer can be configured in VXconfig. The chip begins with its hardware-strapped address, which is then overwritten by the VX1161 firmware with the address configured in VXconfig. Additionally, the SetConfig(8,x) CASL command can be used to change the I2C address dynamically.

The sensor I2C address is defined by the sensor itself. Accesses by the ECU will transparently be forwarded through the sensor link.

The GVIF3 bypass feature minimizes transmission delays for sensor register access. See the CXD4966 and GVIF3 documentation for details.

MFP Interconnect

With the GVIF bypass, GPIO accesses are transparently routed between the ECU and the sensor serializer.

Additionally, the VX1161.52L has a Sensor Trigger Generator which can generate synchronized trigger pulses on one MFP pin per channel (see section Sensor Trigger Generator on page 401).

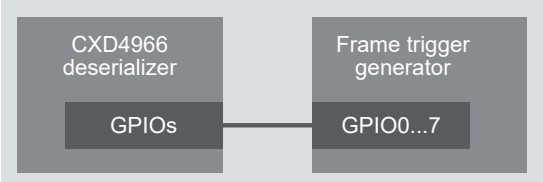


Figure 149: TAP mode; payload reception

Note
Care must be taken to avoid signal collisions between GPIO outputs and Sensor Trigger Generator outputs.

Connected to Signal of Deserializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**
CH1 -GPIO0	IO	CH1 -GPIO_0
CH1 -GPIO1	I, LOCK	CH1 -GPIO_1
CH1 -GPIO2	I	CH1 -GPIO_2
CH1 -GPIO3	-	CH1 -GPIO_3
CH1 -GPIO4	IO	CH1 -GPIO_4
CH1 -GPIO5	O	CH1 -GPIO_5
CH1 -EXT_SM_FAULT	IO	CH1 -GPIO_6
CH1 -SM_FAULT_X	O	CH1 -GPIO_7

Signal mapping of TAP deserializer GPIOs to trigger generator GPIOs; directions with view of deserializer (IO_OD: Input+open drain output); LOCK: Do not use. Signals required internally.

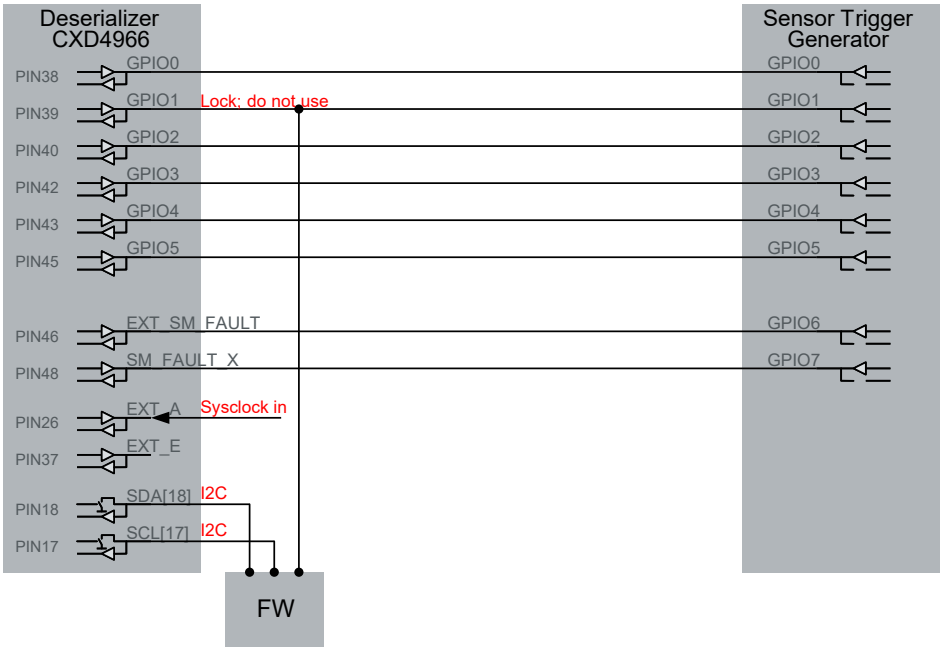


Figure 150: TAP GPIO architecture; GPIO connections between TAP deserializer and Sensor Trigger Generator

4.5.22.8 Channel in RX Mode

If a channel is configured in RX mode, the incoming GVIF3 data stream is received by the CXD4966 deserializer and forwarded to the recording data processing.

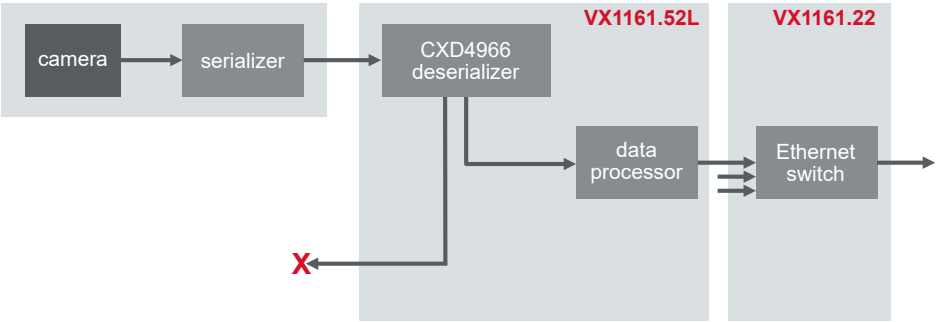


Figure 151: TAP channel in RX mode; payload reception

Initialization

The initialization sequence of the sensor link follows this chart:

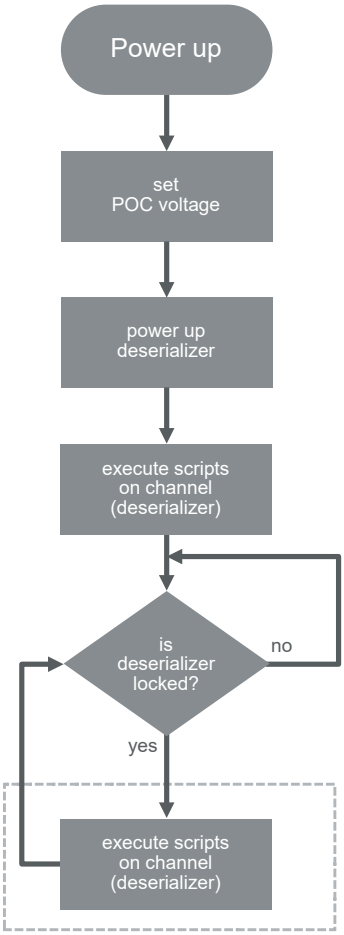


Figure 152: RX initialization sequence

After power-up, the POC voltage is applied as configured and the deserializer is activated.

Next, the enabled scripts for the RX channel are executed immediately (script 1, 2, 3) to configure the deserializer, sensor serializer and sensor as early as possible.

After the initial run, the script execution engine waits for a deserializer lock.

When the deserializer locks, the script execution is repeated for all enabled scripts in a round robin order (1, 2, 3, 1, 2,..), if one of the follwing conditions apply:

- Script is not configured as "run once" (CASL SetConfig(7,1))
- OR script is configured as "repeat endlessly" (CASL SetConfig(5,1))
- OR script terminated with a return value ≠ 0 in previous run

If the deserializer link unlocks, the script sequence repeates from the inital deserializer run on.

I2C Architecture

In RX mode, the TAP deserializer, sensor serializer and sensor need to be configured via CASL scripts. A dynamic sensor reconfiguration is currently not supported.

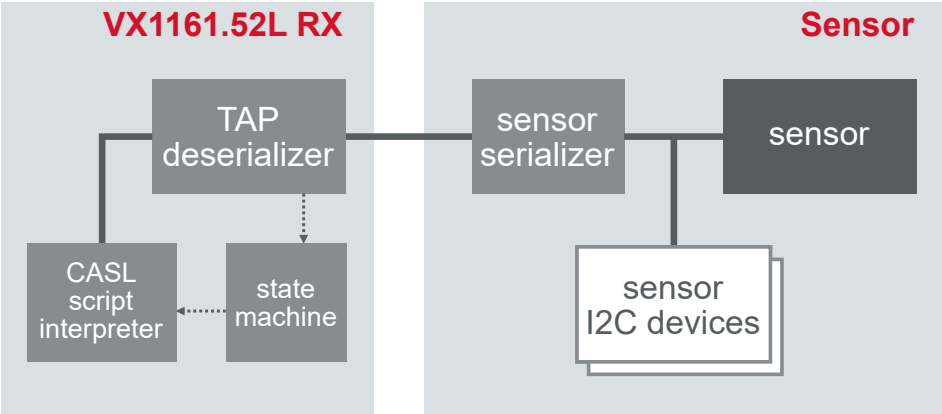


Figure 153: RX mode; I2C architecture; CASL scripts configure deserializer, serializer and sensor

The I2C address of the TAP deserializer can be configured in VXconfig. The chip starts with its hardware strapped address and is then overwritten by the VX1161 firmware with the address configured in VXconfig. Additionally, the SetConfig(8,x) CASL command can be used to change the I2C address dynamically.

The sensor serializer I2C address is defined by its hardware strapped pins and could be changed via CASL scripts.

The sensor I2C address is defined by the sensor itself. CASL scripts can be used to configure the sensor via I2C transactions. To enable I2C accesses to the sensor, deserializer and serializer registers might have to be set properly in advance.

MFP Interconnect

The VX1161.52L has a Sensor Trigger Generator which can generate synchronized trigger pulses on one MFP pin per channel.

For forwarding GPIOs (multi function pins) from the Sensor Trigger Generator to the sensor, the TAP deserializer and the sensor serializer registers must be configured properly. This can be done via CASL scripts.This also enables the configuration of static GPIO levels inside the deserializer.

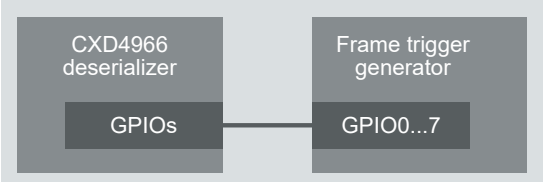


Figure 154: RX mode; payload reception

Note
Care must be taken to avoid signal collisions between GPIO outputs and Sensor Trigger Generator outputs.

Connected to Signal of Deserializer		Signal of Sensor Trigger Generator
Name	**Direction**	**Name**
CH1 -GPIO0	IO	CH1 -GPIO_0
CH1 -GPIO1	I, LOCK	CH1 -GPIO_1
CH1 -GPIO2	I	CH1 -GPIO_2
CH1 -GPIO3	-	CH1 -GPIO_3
CH1 -GPIO4	IO	CH1 -GPIO_4
CH1 -GPIO5	O	CH1 -GPIO_5
CH1 -EXT_SM_FAULT	IO	CH1 -GPIO_6
CH1 -SM_FAULT_X	O	CH1 -GPIO_7

Signal mapping of deserializer GPIOs to trigger generator GPIOs; directions with view of deserializer (IO_OD: Input+open drain output); LOCK: Do not use. Signals required internally.

4.5.22.9 Sensor Trigger Generator

The VX1161.52 provides a Sensor Trigger Generator for every channel. This generator can be used to generate a trigger signal to one of the GPIO/MFP pins. If the serializer and deserializer are configured correctly via CASL scripts, this trigger signal will be forwarded to the sensor and triggers the acquisition of data.

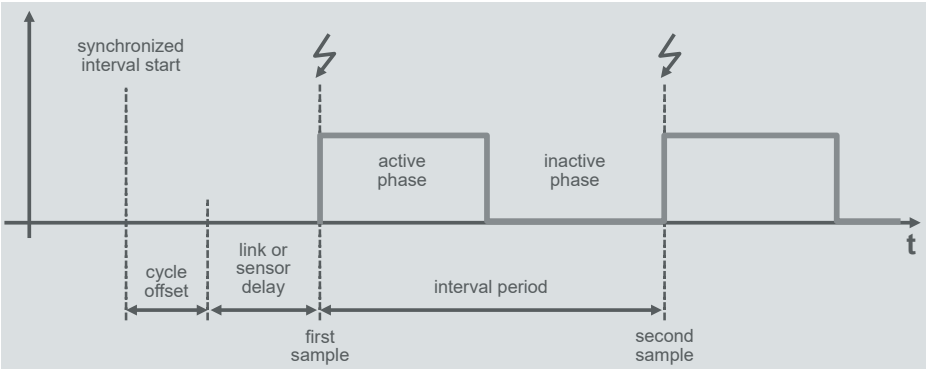


Figure 155: Sensor Trigger Generator with adjustable interval period

The generators of one module use one hardware timer as time base. This timer can be synchronized via PTP between multiple cards. If the PTP service is available, the trigger generation on all channels on all synchronized modules will be synchronous. If the PTP service is not available, only the channels on one module will be synchronous.

The current implementation requires that every VX1161.52 can exchange Ethernet broadcast messages with every other VX1161.52 during measurement start. These broadcast messages are used to negotiate the precise start of the first Frame Trigger period.

If a measurement setup consists of several VX1161 racks, there must be a path the broadcast messages can take to reach all other cards. Since Windows natively does not forward Ethernet broadcast messages from a receiving port to all other Ethernet ports, another connection is needed. To provide that path, an additional Ethernet cable between all VX1161 racks is required. The Ethernet switch inside the VX1161.22 uplink module will distribute those broadcast messages. Any Ethernet port on the VX1161.22 can be used due to the low bandwidth requirement.

The generators start outputting their trigger signal at start of measurement of the receiving application and it will stop generation at measurement stop. As a result, the sensor might not generate data when the measurement is stopped.
When the measurement is started, the time on all channels starts simultaneously and all generators start with their cycle offset times.

The trigger signal has two phases: the active and the inactive phase. Both can be configured in their duration. Together they define the trigger interval period, which defines the frequency of trigger pulses and the frame rate. Trigger generation always starts with the active phase. The Cycle offset parameter can be used to compensate delays in serial links or sensors. For sensors on fast paths, a delay can be added to trigger at the same time as sensors on slow paths.

For sensors requiring falling edges as trigger, the polarity of the output signal can be selected. If not inverted, the active phase starts with a rising edge which is intended to trigger the sensor.

Note
Care must be taken to avoid signal collisions between serializer or deserializer outputs and Sensor Trigger Generator outputs.

The Sensor Trigger Generator is configured via VXconfig and the following parameters:

Setting	Description
Enable	Activates the Sensor Trigger Generator. If disabled, the GPIOs are high impedance.
Cycle Offset	Defines a static delay from the ideal interval start in nano seconds. Can be used to compensate different delays in between several links or sensors.
Active Phase	Defines the duration of the active phase in nano seconds.
Inactive Phase	Defines the duration of the inactive phase in nano seconds.
GPIO output pin to trigger	GPIO pin of the VX1161.52, where the trigger signal shall be output. Check the GPIO/MFP section of your card to see which pins can be used and how they map to deserializer or serializer pins.
GPIO polarity active phase	Can be used to invert the output signal polarity. - checked (default): active phase starts with a rising edge. Active phase is high. Inactive phase is low. - unchecked: active phase starts with a falling edge. Active phase is low. Inactive phase is high.

4.5.22.10 Frame Trigger Detector

The VX1161.52 provides a Frame Trigger Detector for every channel in TX mode. This detector can be used to detect the change of one GPIO/MFP pin and trigger the transmission of the next video frame. This feature allows to synchronize frame transmissions to ECU trigger signals and with this to the ECU time domain.

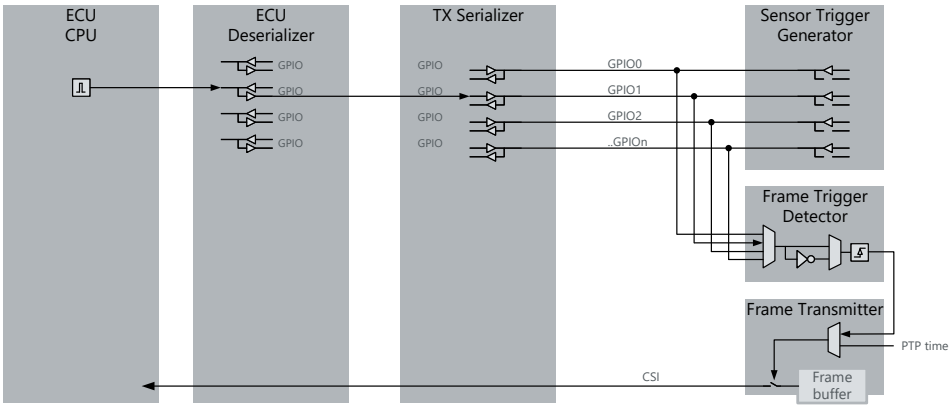


Figure 156: CPU generates frame trigger; Frame Trigger Detector detects edges and triggers the frame transmission

The detector can be fed by one GPIO/MFP of the channel group the channel belongs to. For selecting the active rising or falling edge, chose the polarity of the trigger signal.

Note
Care must be taken to avoid signal collisions between serializer outputs and Sensor Trigger Generator outputs.

The Frame Trigger Detector is configured via VXconfig and the following parameters in the Output Description section.

Setting	Description
GPIO input pin	GPIO pin of the VX1161.52, which shall feed the edge detector. Check the GPIO/MFP section of your card to see which pins can be used and how they map to your serializer pins.
GPIO polarity active phase	Specifies whether rising edges (checked) or falling edges (unchecked) on the given GPIO pin trigger the frame transmission.

4.5.22.11 Required Deserializer Settings

The following requirements apply to the deserializer on the VX1161.52L. The settings for the connected serializer at the sensor and deserializer in the ECU are allowed to differ according GMSL capabilities.

CSI Continuous Clock

Due to internal technical reasons, the CSI clocks on all channels must run in continuous CSI clock mode only. This mode is initially configured by the VX1161.52 firmware. Changing this setting will interrupt reception of data.

CSI Clock Lane Speed

Due to internal reasons, the CSI clock speed of the serializer must be configured via the Channel Group Mode and not be changed. The VX1161.52L firmware will configure this at start-up. If this setting is changed via CASL scripts, data transmission will no longer be possible.

Link Lock Signals on GPIO Pin

To detect link status changes, the VX1161 firmware configures the GPIO1 pin of the CXD4966 TAP deserializer to output the lock signal. If this pin's function is changed, the TAP will no longer be able to detect unlocks, which may disrupt script execution.

CSI Signal Settings

The CSI lanes of the TAP deserializer CXD4966 does not require any lane swap, lane inversion or timing adjustments.

4.5.22.12 POC Supply

The VX1161.52L delivers power to connected sensor heads via the coax cable using Power-over-Coax (POC). It has an internal, isolated supply that is adjustable and independent of VX GND. It can source up to 2 W. The voltage can be switched off or adjusted from 5 V ... 12 V using VXconfig.

In TAP mode, the internal supply can be deactivated, allowing the POC supply of the ECU to be forwarded to the sensor head. This gives the ECU control over the sensor's power.

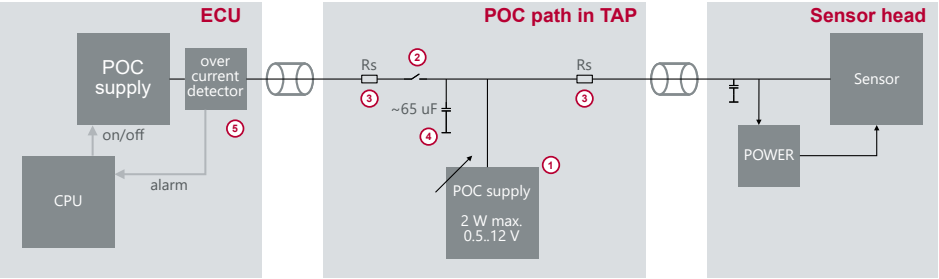


Figure 157: POC structure in VX1161.52L TAP channels

The internal series resistance (Rs) (3) of the POC bypass path is approximately 2 x 2.5 Ohms, and the internal capacitance (4) is approximately 65 µF. This capacitance can cause or increase inrush currents for the ECU that sources power and might trigger overcurrent detection features (5). In such cases, it might be an option to source the sensor from the VX (1) instead of the ECU, but then the ECU would have no control over the sensor power.

Furthermore, the internal VX POC supply can be output to the ECU for use cases in which CSI ingestion PODs require POC supply.

The POC voltage level, generation, and delivery direction are controlled via the POC voltage settings for the TAP_RX and TAP_TX ports in VXconfig as follows:

Note
Sourcing power in TAP mode is supported since VXtools v4.9.

Settings				
POC voltage V_{RX}on TAP RX channel	0V	5...12V	0V	5...12V
POC voltage V_{TX}on TAP TX channel	0V	0V	5...12V	5...12V
Effects				
POC bypass (2)	closed	open	closed	closed
VX POC source (1) voltage	off	V_{RX}	V_{TX}	Min(V_{RX},V_{TX})
Use case	ECU supplies sensor	VX supplies sensor	VX supplies ingest POD	VX supplies sensor and ingest POD

4.5.22.13 Technical Data

Weight	210 g
Power consumption	▶ P_{init}; initialization (~500ms): 500mW ▶ $P_{standby}$; sleep mode: <18 mW ▶ P_{run}; normal operation excl. POC: max.10 W
POC (power over cable)	▶ V_{poc}; voltage sourced by VX1161.52: 0 V, 5...12 V configurable ▶ P_{poc}; maximum power delivered per RX channel; sourced by VX1161.52: 2 W ▶ $V_{PocTapMax}$; voltage forwarded from ECU by TAP: 3...18 V ▶ $I_{PocTapMax}$; max. current forwarded from ECU by TAP: 500 mA ▶ Electrical isolation between TAP_CH1/2, TAP _CH3/4 and chassis max. 50 V ▶ Rpoc; DC resistance of POC forwarding path in TAP: typ. 2.616 Ohm
Minimum power -off time	▶ minimum time between powering-off and re-appying power: >2 s
Start-up time	▶ t_{init}; initialization after power up: ~ 500 ms ▶ $t_{wakeLink}$; wake up after sleep; GVIF3 link stable: < 50 ms, depends on CASL script execution ▶ $t_{wakeReady}$; wake up after sleep; GVIF3 data to Ethernet: < 15 s
Temperature range	-40 °C ... +60 °C
Maximum payload bandwidths	▶ Card to host via Ethernet: 7.2 Gbps net; total for all channels together ▶ Every TAP/ RX channel GVIF3: up to 5.2 Gbps net
Latencies/delays	▶ $t_{delayMFP}$;; additional delay of GPIOs; see CXD4966 and GVIF documentation for details.
Connector	(Single signal) FAKRA Type Z; color: cyan; compatible to all other (single signal) FAKRA connector types; 6 GHz max; 25 mating cycles;
Maximum cable length	8 m, in both RX and TX direction at the same time
Part number	180250

4.5.23 VX1161.51/52 Bandwidth Limits

This section provides an overview of bandwidth constraints for a VX1161.51/52 streaming system. These constraints define the maximum possible data rates that can be transferred via VX1161.51 or VX1161.52 streaming interface cards.

For an error free operation without data loss or data corruption the aggregated data rates at every point of the data path must not exceed the specified limits.

Special attention must be paid to the calculation of bandwidth losses due to inter packet delays (e. g. blanking) and coding overheads.

Additionally, if burst data rates (e. g. CSI packet transfers) exceed individual limits, internal buffer sizes as well as input and output bandwidths must be considered.

Contact Vector for a reliable system evaluation.

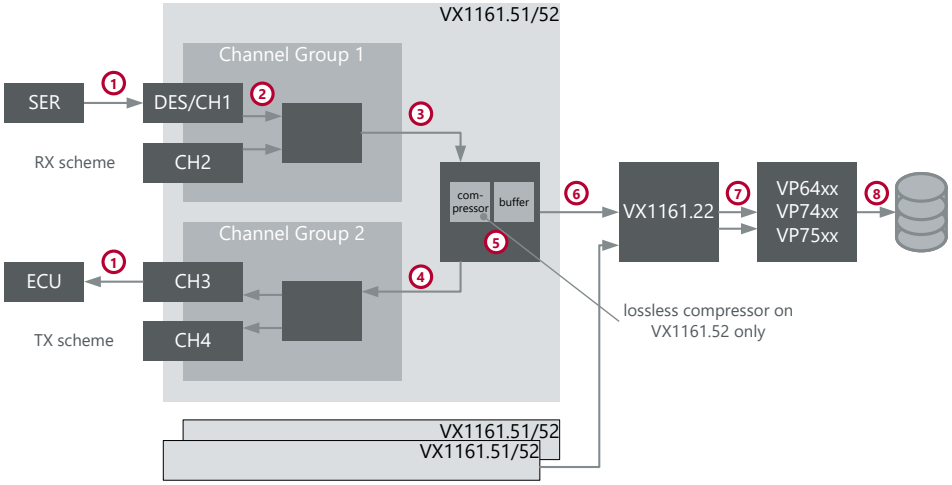


Figure 158: Bandwidth Limits (Note: Figure for documentation purposes only; does not represent actual individual TAP, RX, TX implementation architectures)

Item	Interface Limits
1	Cable limit: - FPD-LINK III: 4.1Gbps (line rate); 3.3 Gbps (net) - FPD-LINK IV: 7.5 Gbps (line rate); 6.3 Gbps (net) - GMSL II pixel mode: 6.0Gbps (line rate); 4.8 Gbps (net) - GMSL II tunneling mode: 6.0Gbps (line rate); 5.4 Gbps (net) reduced by CSI low power state transitions and blanking times (time between packets); typically -10..-20%
2	CSI lane speed limit: - FPDLINK III: 4x 800 Mbps; two channels per Channel Group - FPDLINK IV: 4x 1500 Mbps; one channel per Channel Group - GMSL2: 4x 1200 Mbps; one channel per Channel Group - GMSL2: 4x 800 Mbps; two channels per Channel Group reduced by CSI low power state transitions and blanking times (time between packets); typically -10..-20%
3	TAP/RX mode; Channel Group to VX1161.51/52 Base Module limit - 6.4 Gbps (gross); reduced by CSI low power state transitions and blanking times (time between packets); typically -10..-20%
4	TX mode; VX1161.51/52 Base Module to Channel Group limit - 5.9 Gbps (gross); reduced by CSI low power state transitions and

Item	Interface Limits
	blanking times (time between packets); typically -10..-20%

5	VX1161.52 lossless compressor limit (optional):
	- The four compression pipelines in the VX1161.52 are each capable of compressing
	  - one stream with up to 350 MPx/s RAW8...RAW16
	  - one stream with up to 170 MPx/s YUV422
	System memory bandwidth limit (RX+TX) (after compressor):
	- 18 Gbps

6	VX1161.51/.52 to/from VX1161.22/.23 host uplink card limit: - 7.2 Gbps (net)
7	VX1161.22 Host Uplink card to host(s) limit: - VX1161.22B: 2x 10 Gbps (gross); Ethernet - VX1161.22C: 4x 10 Gbps (gross); Ethernet
8	Logging PC limit: - VP64xx: up to 8.0 Gbps - VP74xx: up to 8.0 Gbps - VP75xx: up to 16 Gbps These limits depend heavily on the measurement setup and must be verified by prototyping.

Gbps: Giga bits per second
(net): maximum effective payload data rate

All above figures and the following text apply to both reception (logging) and transmitting (replay) streaming data.

The possible bandwidth for reception and transmission of streaming data depends on a chain of transfer paths and their individual limits. If several streams are accumulated, their combined data rate must be able to pass all succeeding paths.

When starting at the sensor serializer, the used serial protocol (1) defines the link limit. Depending on the used coax cable protocol, the maximum net link bandwidths named in the transceiver datasheet apply. If the VX1161.5x is used in TAP mode and the link speeds on the ECU and the sensor link differ, the lower speed defines the maximum burst rate and the minimum time between packets (blanking).

Stream data from the sensor serializer is received by the TAP deserializer on the VX1161.51/52 and output to the VX1161.51/52 electronics via a CSI interface (2). This interface always uses 4 CSI lanes. The CSI lane rate can be configured in VXconfig, **Channel Group** configuration.

For CSI lane speeds above 800 Mbps, only one channel per **Channel Group** (2 Channel Groups per card) can be used. For 800 Mbps and below, up to two channels per Channel Group can be used. The total receive bandwidth from one Channel Group is limited to 6.4 Gbps (3). For transmission, the total Channel Group limit is 5.9 Gbps (4).

CSI data streams are encapsulated in short and long packets with headers and footers separated by Low Power States (idle times for vertical and horizontal blanking). This overhead is usually in the range of 10 to 20 % of the gross CSI bandwidth and depends mainly on the blanking time settings of your sensor.

The VX1161.52 has the possibility to compress RAW streaming data (2) binary lossless on the card to relief all succeeding data paths and the recording device from high bandwidths. Every channel has its individual compression pipeline. It is capable of compressing up to ~350 MP/s (megapixels per second). The output data rate of the compressor depends on the incoming data contents and is typically

in the range of 40 to 60 % of the input data rate.

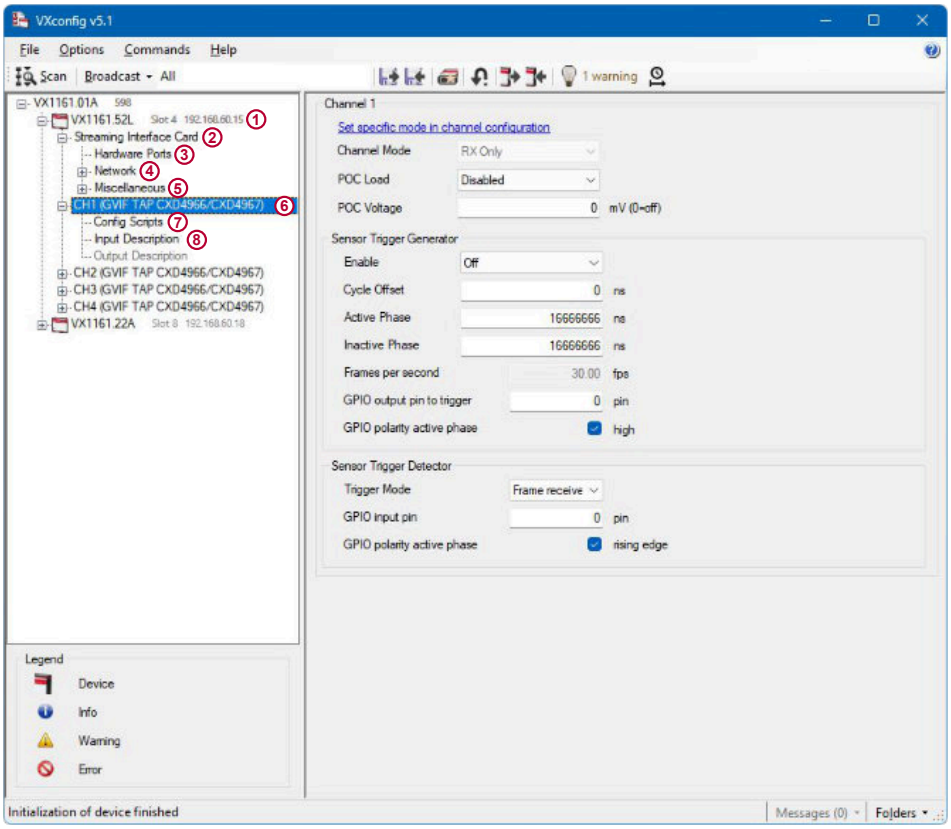
If several streams on one VX1161.51/52 card are received in parallel, the sum of their data must be transferred from the VX1161.5x1/52 card to the VX1161.22 host uplink card. The link between every VX1161.51/52 card and the VX1161.22 host uplink card (6) is capable to send and receive 7.2 Gbps of data in parallel.

The VX1161.22 host uplink card forwards the data of all down link cards to one or more connected storage devices via 10G Ethernet links (7). 10G Ethernet links have a net bandwidth of about 9.9 Gbps when 8 k jumbo frames are used.
If the sum bandwidth of several stream channels exceed the limit of one Ethernet link, their load must be transferred via several 10G Ethernet links. In order to control the proper distribution of data via multiple 10G Ethernet links, you must use individual Ethernet IP subnets. Each subnet must contain one or several VX1161.51/52 cards and one 10G Ethernet port of the storage device.

4.5.24 VXconfig for VX1161.51/.52

General information

To configure the VX1161.51/.52 the VXconig tool must be used.



Item	Description
1	The **VX1161.51** section shows in which slot it is mount and what IP address is assigned. Depending on the hardware variant the **VX device Mode** can be changed in this dialog.
2	For modules in **Video** mode the **Streaming Interface Card** section is shown. For modules in **uC Measurement and Radar** mode the **Radar Interface Card** section is shown,
3	In the **Hardware Ports** section several virtual channels per channel can be routed to a stream receiver.
4	In the **Network** section you can change the IP address as well as Ethernet packet (jumbo frames) sizes.
5	In the **Misc** section uncategorized settings can be changed.
6	For every channel a **Channel** section is shown. There you can adjust channel specific settings.
7	Configure channel initialization scripts in CASL language in the **Config** scripts section.
8	Configure CSI data types, pixel formats and picture geometry settings in the **Input Description** section.

4.5.24.1 Streaming Interface Device

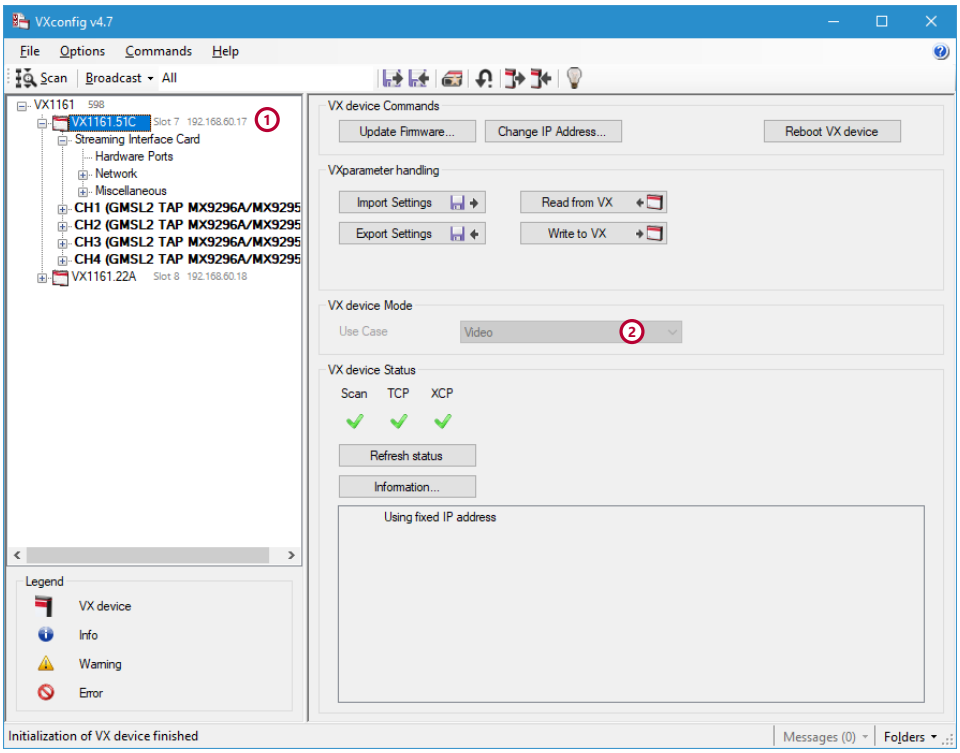


Figure 159: Change use case of card

Item	Description
1	In the system tree view the VX1161 rack slot number and the assigned IP address are shown.
2	Change the VX device mode here. All VX1161.5x modules support the use case **Video** for video data reception or transmission. The VX1161.51C/51D/52C/52C also support the use case **µC Measurement/ Radar**. This setting affects all channels in parallel and it is not possible to run different channels or channel groups in different modes. Different cards can be used in different modes.

After changing the VX device mode it is required to

1. write the configuration to the device first and then to
2. update the firmware (via VXupdate) to the required variant.

VXupdate reads the card mode and channel group configurations from the card for selecting the required images. Therefore configuration changes need to be written to the module prior starting VXupdate.

4.5.24.2 Streaming Interface Card

VX1161.5x modules configured in Video use case (see section Streaming Interface Device on page 411) allow the configuration of the **Module Mode** and **Channel Group Mode**.

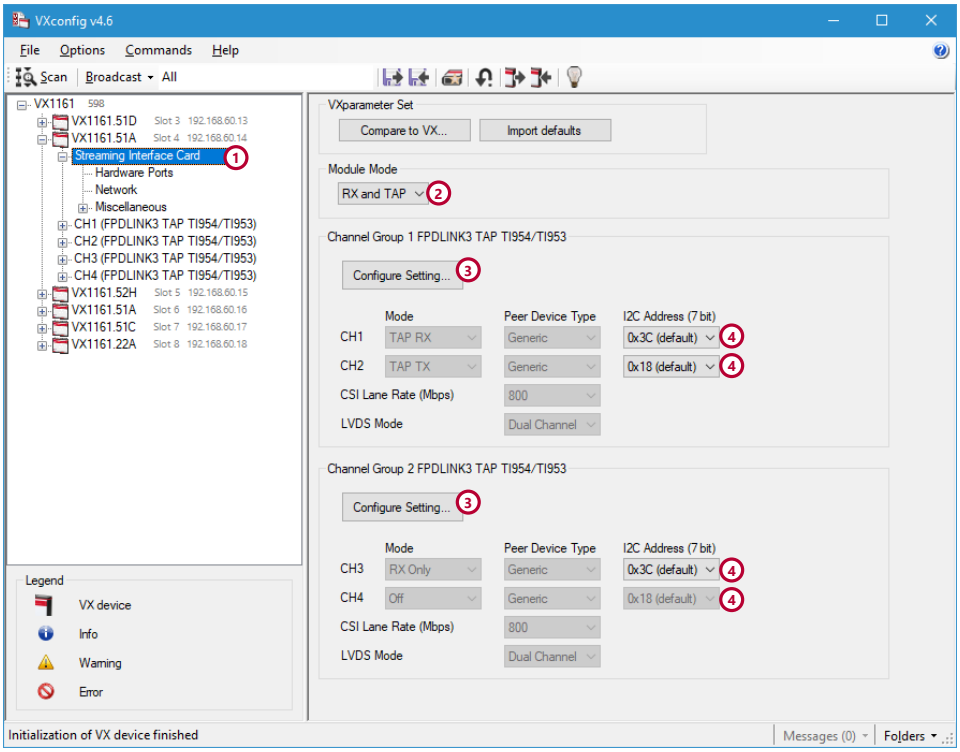


Figure 160: VXconfig

Item	Description
1	The **Streaming Interface Card** section allows changing the channel operating modes and channel hardware specific settings.
2	The possible **Module Modes** depend on the used hardware and their build in transceivers. TAP modules can be configured in either **RX/TAP** (reception) or **TX** (transmission) mode. RX modules or TX modules can only be used in their native mode (RX or TX) and the configuration cannot be changed. This setting affects all channels in parallel and it is not possible to run different channels or channel groups in different modes. Different cards can be used in different modes.
3	Depending on the module mode the **Channel Group Mode** allows setting the channel operation modes to **OFF**, **RX**, **TAP**, **TX**or **TX with sensor** in combination with different CSI lane speeds. Also specific compatibility modes can be selected here.
4	Deserializer and serializer **I2C Addresses** at start-up can be changed from the default address. Only hardware pin strapable addresses are possible. Further addresses and dynamic settings can be done via CASL scripts.

After changing the Module Mode it is required to

1. write the configuration to the device first and then to
2. update the firmware (via VXupdate) to the required variant.

VXupdate reads the **Module Mode** or **Channel Group Mode** from the card for selecting the required images. Therefore configuration changes need to be written to the module prior starting VXupdate.

Use the **Configure Setting** dialog of the **Channel Group** to set the transceiver modes. Depending on the built in transceivers and selected module mode, available transceiver modes are displayed.

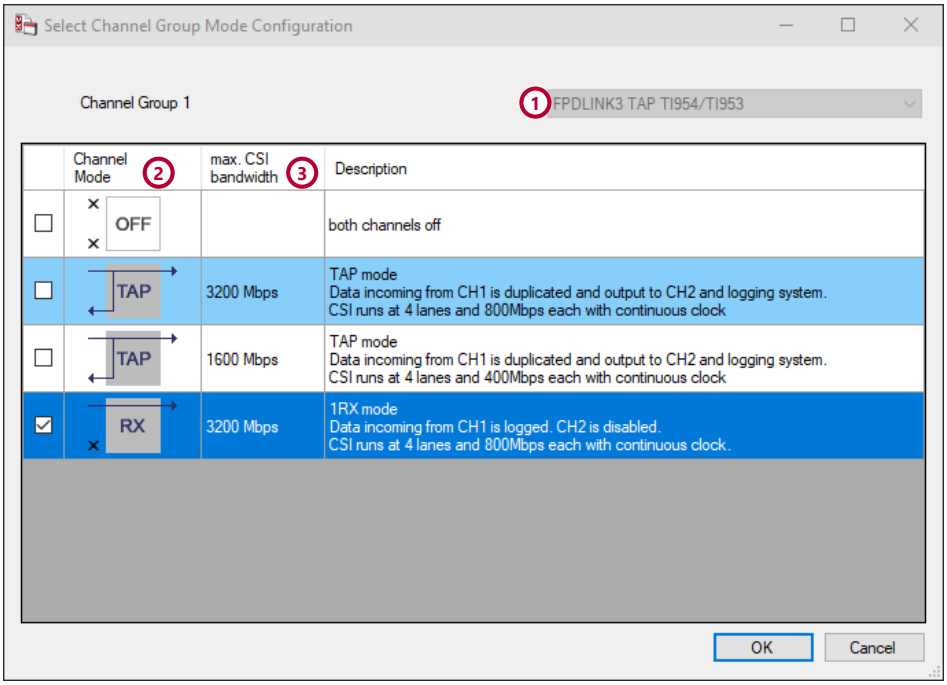


Figure 161: Set Channel Group Mode

Item	Description
1	Possible modes depend on the build in transceivers.
2	The **Channel Mode** field illustrates the channel configuration.
3	The field **max. CSI bandwidth** describes how much data can be received or transmitted by each channel of the **Channel Group**.

Note

Serializers have no transmit buffers. If the CSI bandwidth exceeds the effective net cable bandwidth, data will be lost.

Deserializers have receive buffers in sizes of 16 kB to 32 kB. If the CSI bandwidth is less than the incoming data stream, data will be lost.

After changing the **Module Mode** or **Channel Group Mode** it is required to

1. write the configuration to the device first and then to
2. update the firmware (via VXupdate) to the required variant.

VXupdate reads the card mode and channel group configurations from the card for selecting the required images. Therefore configuration changes need to be written to the module prior starting VXupdate.

Hardware Ports

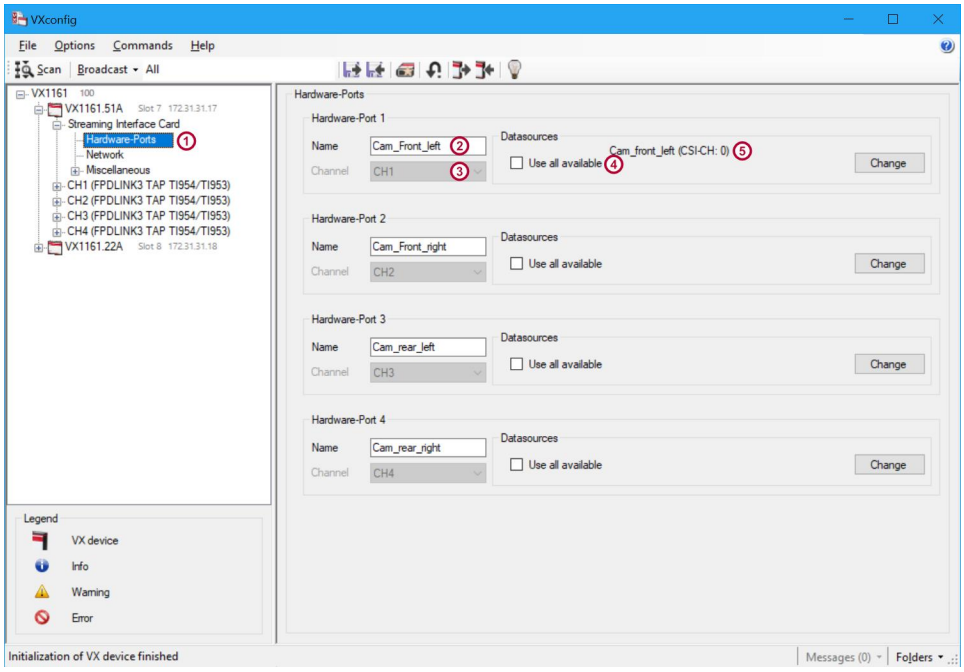


Figure 162: VXconfig

Item	Description
1	High bandwidth data interfaces between the VX1161.51 and a serving or consuming host device (e. g. PC) are named **Hardware Ports**. The sources for those ports can be configured in this dialog.
2	A descriptive port name or text for every **Hardware Port** can be configured. This name will be attached to the port data in the logging file for easier channel or data identification.
3	The **Channel** field defines the sourcing channel for a **Hardware Port**. Currently only a fixed channel can source a **Hardware Port**.
4	A **Hardware Port** can have several data sources for the selected channel (e. g. multiple virtual channels for multiple exposure video streams). If you select **Use all available channels**, all channels which are activated in the **Input Description** section will be associated.

Network

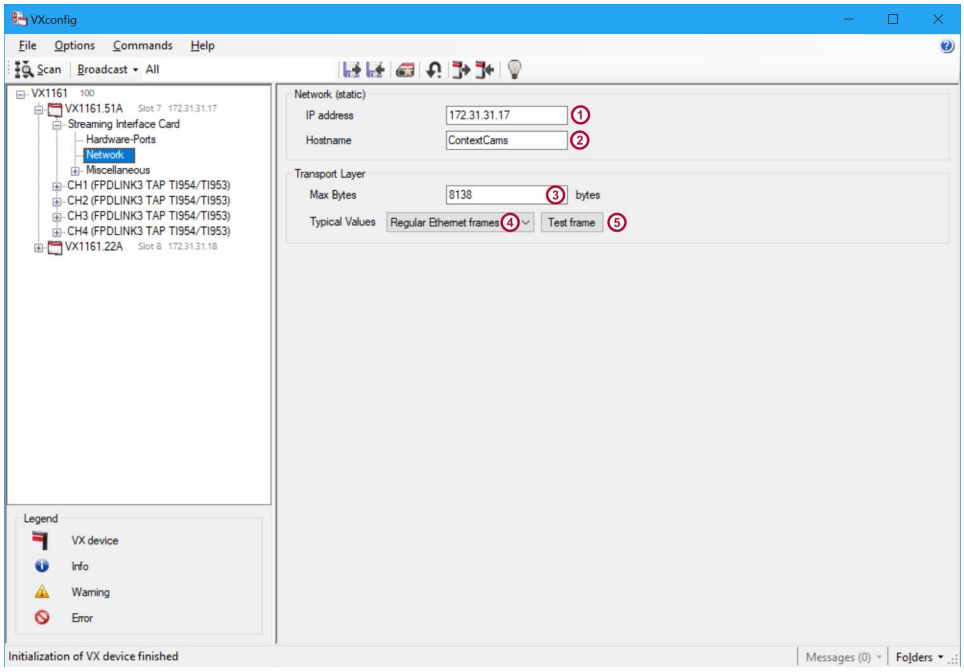


Figure 163: VXconfig

Item	Description
1	Enter a static IP address in this field suitable to your network settings.
2	A host name string can be entered to identify this card (e. g. via ping command).
3	In this field you can specify the maximum Ethernet packet size. This packet size must be supported by your Ethernet adapter and any Ethernet packet switch in between. Choose the largest possible value to unload the receiving device from interrupt load. You can test the set maximum Ethernet packet size which can pass from the card to VXconfig by pressing the **[Test frame]** button.
4	Select one of typical Ethernet frame sizes. You can test the set maximum Ethernet packet size by pressing the **[Test frame]** button.
5	You can test the set maximum Ethernet packet size which can be passed from the card to VXconfig by pressing the **[Test frame]** button.

PTP

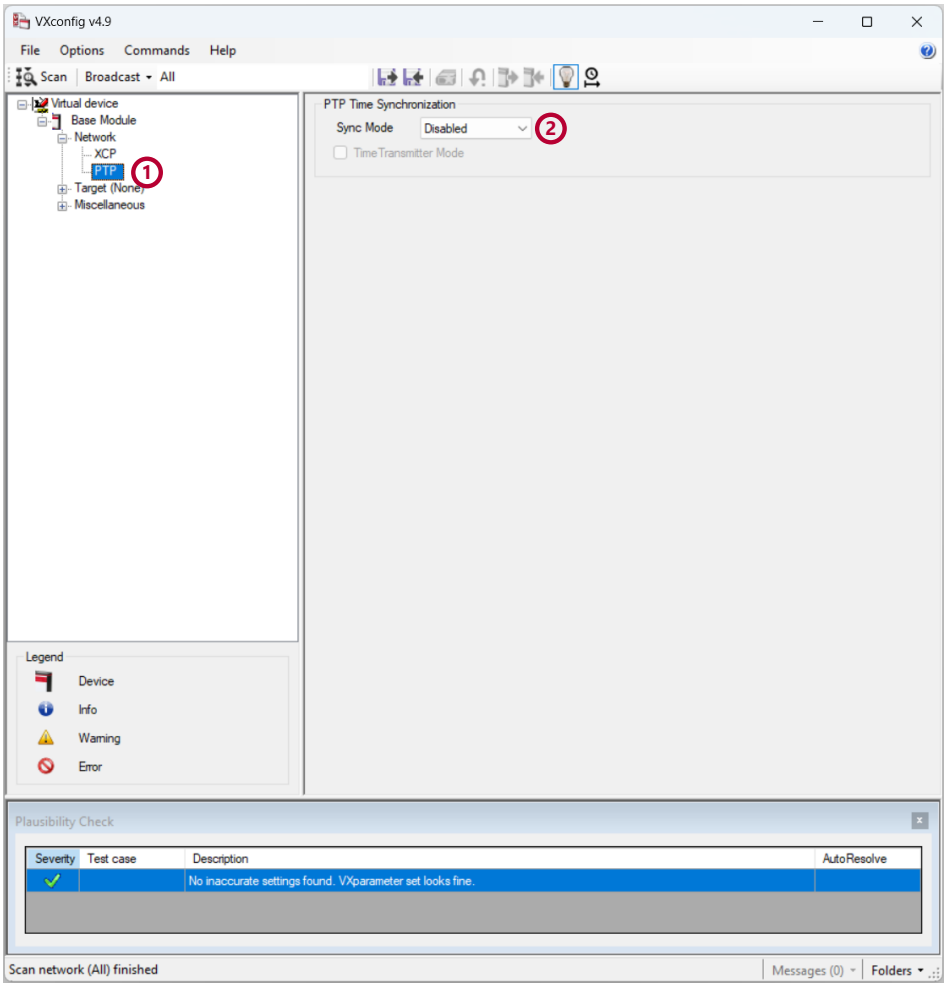
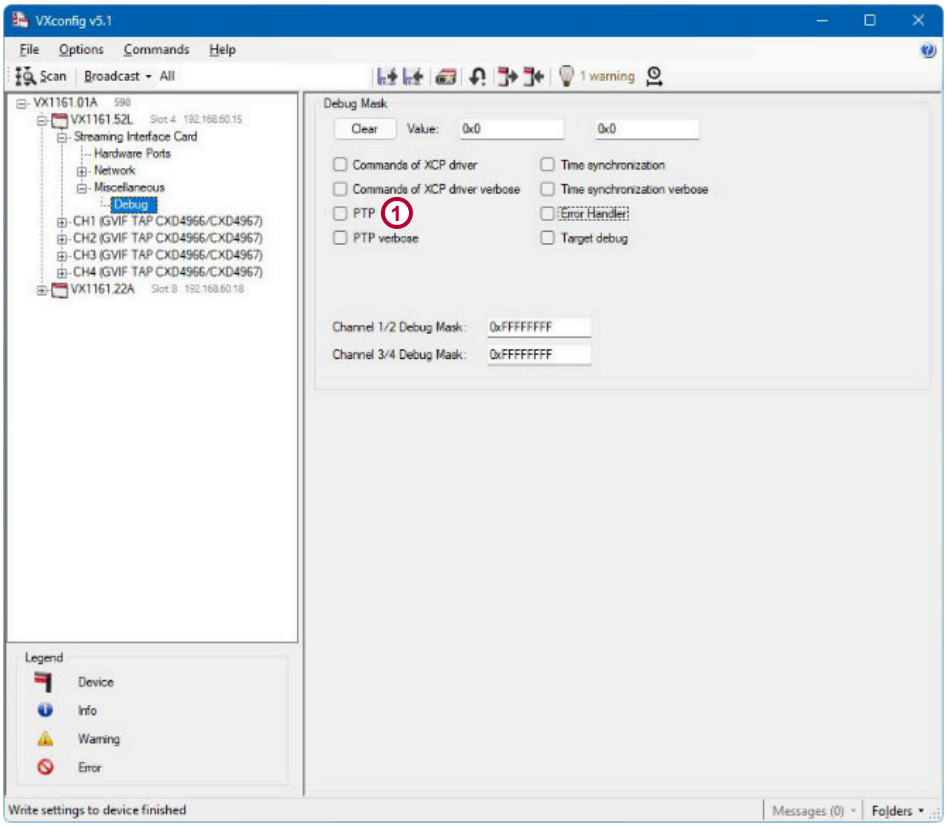


Figure 164: VXconfig

Item	Description
1	In the **PTP** section, you can modify time synchronization settings.
2	With the **Sync Mode** you can adjust the time synchronization method to be used. Select **Disabled** if no PTP timeTransmitter clock exists in your measurement network. Select **Enabled** if the card shall synchronize to an existing PTP timeTransmitter clock of good quality. If the PTP timeTransmitter clock has a jitter higher than 1 µs, select **Enabled (no steady-state observation)** to avoid permanent resynchronization attempts.

Miscellaneous | Debug | PTP



Item	Description
1	If there are problems with PTP synchronization, the two PTP debug flags can be enabled. When enabled, detailed debug prints are generated on the VX console.

Channel Configuration

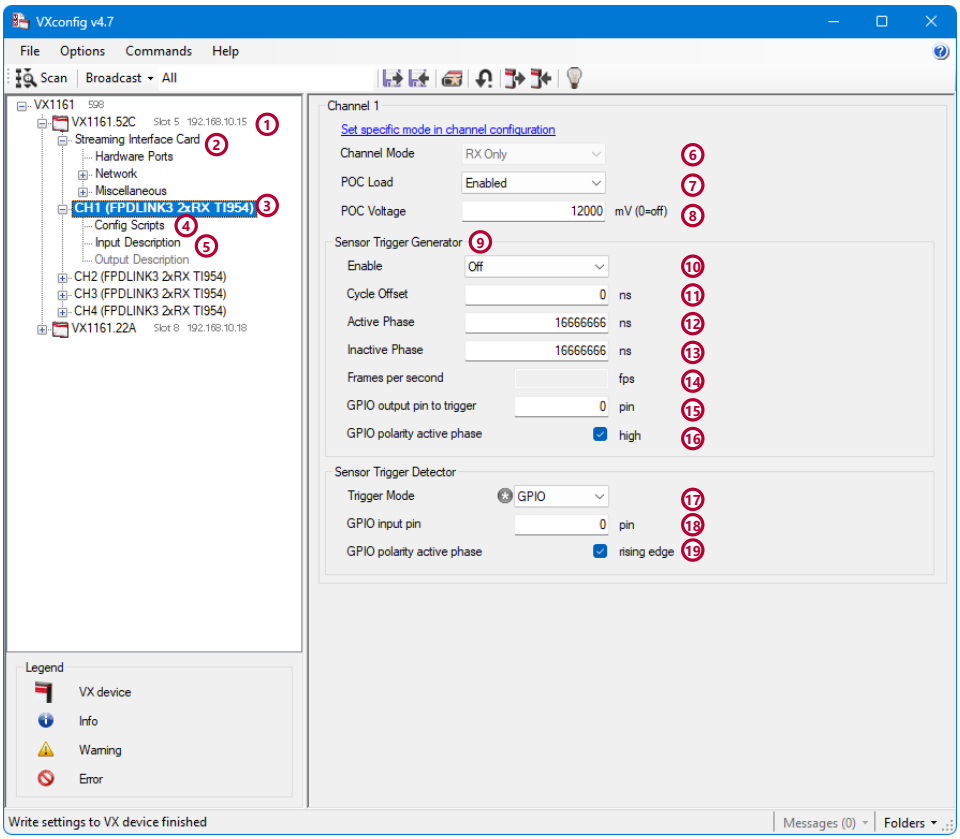


Figure 165: VXconfig

Item	Description
2	The operating mode of the card and its channels must be changed in the **Streaming Interface Card** section.
3	Channel specific settings can be changed in the channel section.
4	Configure channel initialization scripts in CASL language in the **Config Scripts** section.
5	Configure the data type and formats for the selected channel in the **Input Description** section.
6	The operating mode of the selected channel is specified in the **Streaming Interface Card** section (2). Since this setting might be dependent on other channels this setting cannot be changed here. Click the blue hyperlink to open the **Streaming Interface Card** section to change the card and channel modes.
7	The POC load (not POC supply) allows to load (draw power from) the ECU supply by about 1 W. This option is only available if the selected channel is either in **TAP_TX** or **TXonly** mode.
8	Via the **POC Voltage** field, the power-over-coax power source for the selected channel can be configured. The POC power supply can source up to 2 W. If no supply is required, set this value to 0 mV for disabling the source. If a voltage between 5000 mV and 12000 mV is set, the power supply is enabled, and the voltage is adjusted to the selected value. This option is only enabled on channels configured in **RXonly** mode. If the mode for a channel is set to another mode than **RXonly**, the POC Voltage is automatically set to 0 mV.

Item	Description
9	The VX1161.51C provides a Sensor Trigger Generator (see section Sensor Trigger Generator on page 279) for every channel. This generator can be used to generate a trigger signal to one of the GPIO/MFP pins of the serializer or deserializer. When configured correctly via CASL scripts, this trigger signal will be forwarded to the sensor and triggers the acquisition of data.
10	The Sensor Trigger Generator can be enabled or disabled. If disabled, the GPIOs are high impedance.
11	The **Cycle Offset** field defines a static delay from the ideal interval start in nano seconds. It can be used to compensate different delays between several links or sensors.
12	The **Active Phase** field defines the duration of the active phase of the trigger signal in nano seconds.
13	The **Inactive Phase** field defines the duration of the inactive phase of the trigger signal in nano seconds.
14	In the **Frames per second** field, the calculated fps value is displayed. The period time is defined by the active phase and the inactive phase values.
15	Specify on which VX1161.51 GPIO output pin the trigger signal shall be generated. Check the GPIO/MFP section of your card in the VX1000 manual to see which pins can be used and how they map to deserializer or serializer pins.
16	The trigger output signal can be inverted by the GPIO polarity active phase field. If checked (default), the active phase starts with a rising edge, and the inactive phase is low. If unchecked, the active phase starts with a falling edge and the inactive phase is high.
17	Configure the **Sensor Trigger Detector** for RX and TX mode. In RX mode the **Trigger Mode**specifies which event is used to take the timestamp for the received frame. If **GPIO** is selected, the frame timestamp is taken when the configured edge is detected and attached to the frame when it arrives. If **Frame Received** is selected, the frame timestamp is taken on arrival of the StartOfFrame MIPI CSI packet. In TX mode the **Trigger Mode** specifies when the next frame shall be output. Select **PTP** if the frame shall be output when the frame timestamp, provided by the Stream Provider application, equals the synchronized PTP time. Select **unsync** if the VX1161.51/52 shall output the frame when the card local time is used for inter frame timing. Select **GPIO** if the next frame shall be output when the edge detector triggers. See also the "Channel in TX mode" sections of your card for details.
18	If **GPIO** is selected as timestamp source or synchronization mode, **GPIO input pin** specifies which pin to use for triggering frame transmissions. See also the "Channel in TX mode" sections of your card for serializer/deserializer to GPIO mapping.
19	If **GPIO** is selected as timestamp source or synchronization mode, **GPIO polarity active phase** specifies whether rising edges (checked) or falling edges (unchecked) trigger frame transmissions.

Config Scripts

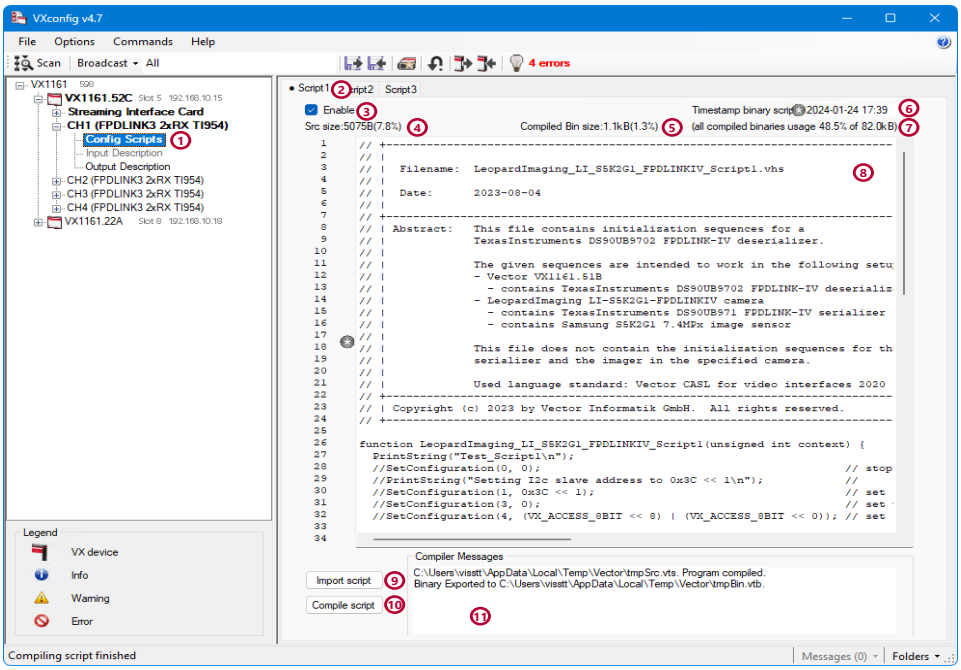


Figure 166: VXconfig

Item	Description
1	In the **Config Scripts** section serializer, deserializer and imager configuration files can be defined which initialize the devices at start up and link status changes.
2	Select one of three scripts for the selected channel. Script1 is executed first, then script 2 and script 3 last.
3	For every script, its execution can be **enabled** or disabled. Disabled scripts are not executed but are kept stored in the interface.
4	Because the space for script sources are limited the size of the current script is displayed.
5	Also the storage space for the compiled binary is limited an is displayed/updated after the script was compiled.
6	The **timestamp** of the binary script shows when the script on the interface was last time modified, compiled and downloaded.
7	The total size of all binaries for the selected channel is also limited and displayed/updated after compilation.
8	Edit or paste your initialization scripts into the **Script** edit field. If the amount of entered source code exceeds the card limit, an error message will pop up during download. The line number of the current cursor position is shown in the current line field (7).
9	If scripts are provided or stored in individual files, they can be **imported** via this button.
10	Press the **[Compile script]** button for syntax check. The script is downloaded to the card together with the entire configuration by pressing the **Write to VX device** menu button.
11	Syntax errors will be reported to the **Compiler Message** box.

Reference

See section Scripting Language (CASL) on page 564

Input Description

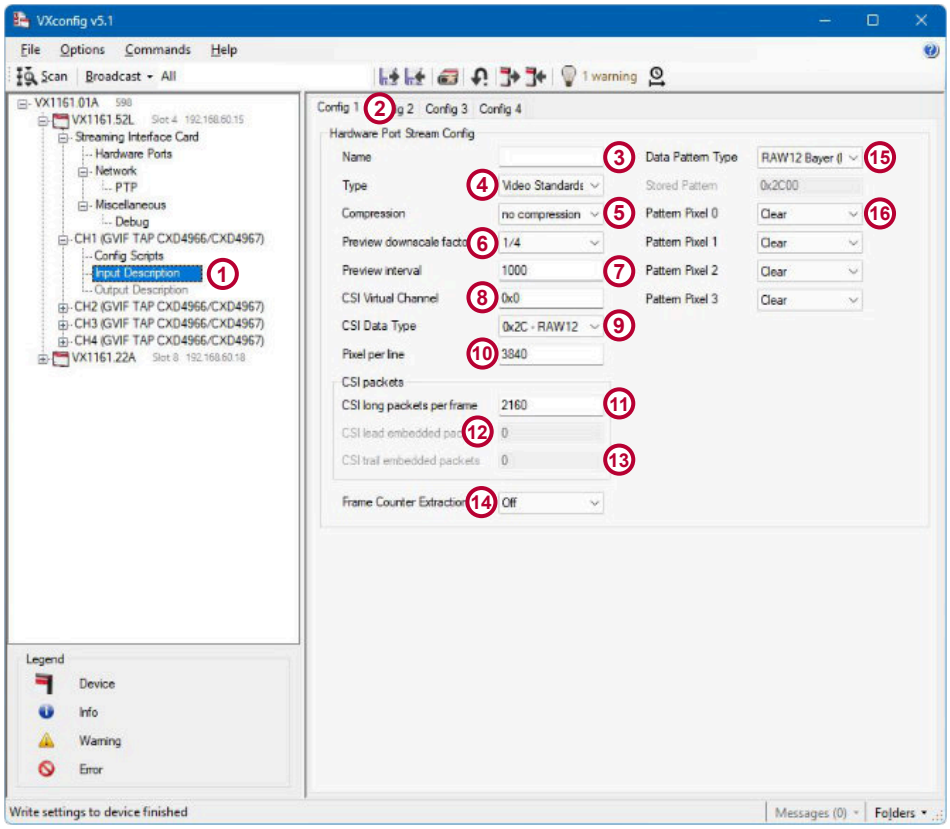
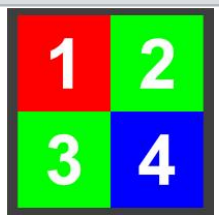


Figure 167: Input Description section in VXconfig

Item	Description
1	In the **Input Description** section, the video data types and formats for receiving can be specified.
2	The VX1161.5x provides up to four hardware channels. The data from those hardware channels are forwarded to the application via Hardware Ports and can carry up to three streams. The properties of those streams need to be configured via the three **Config** tabs. These settings must match the data from your data source. If it does not match, the data cannot not be recorded properly.
3	In the **Name** field, you can enter a free text. It is intended to add a descriptive text to the channel data. It will be recorded together with the sensor data.
4	The **Type** field specifies the kind of data this virtual channel carries. If set to **Not used**, this virtual channel is deactivated. If set to **Video Standards**, this virtual channel can be recorded.
5	In the **Compression** field, the desired video compression method must be selected. - Select **no compression** to disable any compression. - Select **MobilEye on PC** to enable a loss less compressor on the logging PC. This option requires a MobilEye compression library license. - Select **Vector on VX** on VX1161.52 cards only to enable lossless compression on the card hardware . All CSI long packets are fed into the compressor. CSI headers and footers are not recorded. See the Lossless Compression section for details. - Select **Vector in VX V2** on VX1161.52 cards only. If selected, the CSI packets specified via the CSI leading and trailing embedded packets

Item	Description
	of a frame are stored uncompressed. The CSI headers and footers of all packets are also stored uncompressed, and the pixels are losslessly compressed on the card hardware. See the Lossless Compression section for details. See the Lossless Compression section for details.See the Lossless Compression section for details.
6	The **Preview downscale factor** defines the factor by which the recorded video shall be downscaled in both X and Y direction for previewing during recording. A higher downscaling relieves the tool from high data rates but degrades the preview picture quality. This factor does not affect the recorded video, which will always be recorded without any quality loss.
7	With **Preview interval** you can specify the time between two updates of the preview video. So it affects the frame rate of the preview video and the required processing power of the tool. A longer interval relieves the tool.
8	In the **CSI Virtual Channel** field, the desired virtual channel must be selected. If the wrong virtual channel is selected, its data will not be recorded.
9	In the **CSI Data Type** field select the data type your sensor uses to output visible pixel lines. This setting, togehter with the data pattern type, affects how the received data will be interpreted. Selecting a RAW data type will enable the four Pattern Pixel fields. Contact Vector for supporting other data types.
10	Set the number of **Pixel per line** field according to your sensor data. This setting will be used for an error check during recording.
11	Set the **CSI long packets per frame** to the total number of CSI long packets sent by your sensor for the specified virtual channel. This setting will be used for an error check during recording.
12	The **CSI leading embedded packets** field is only used for the **Vector in VX V2** compression mode. It specifies the number of CSI long packets at the beginning of a frame that contain embedded data rather than visible pixels. These packets will bypass the compressor and be logged uncompressed. See the Lossless Compression section for details. The specified lines will be excluded from the line length check for the **Pixel per line** parameter.
13	The **CSI trailing embedded packets** field is only used for the **Vector in VX V2** compression mode. This field defines how many CSI long packets at the end of a frame contain embedded data or statistics rather than visible pixels. These packets will bypass the compressor and be logged uncompressed. See the Lossless Compression section for details.The specified lines will be excluded from the line length check for the **Pixel per line** parameter.
14	The **Frame Counter Extraction** field allows embedded data from supported image sensors to be parsed, and the imager frame counter value is recorded as a frame counter signal in CANape. If **Off** is selected, the VX1161.5x counts the frame counter signal in CANape from the start of the measurement.
15	The **Data pattern type** field specifies alternative data formats for a given CSI Data Type. Contact Vector for supporting other data pattern types.
16	Only for RAW CSI data types the four **Pattern Pixel** fields can be used to specify the colors of the four quadrants of a Bayer color filter. The positions of the quadrants are as follows:

Item	Description
	

Note
The **Input Description** section is only available if your card is configured in RX and TAP mode.

Output Description

Transmission mode is supported since VXtools 4.5.

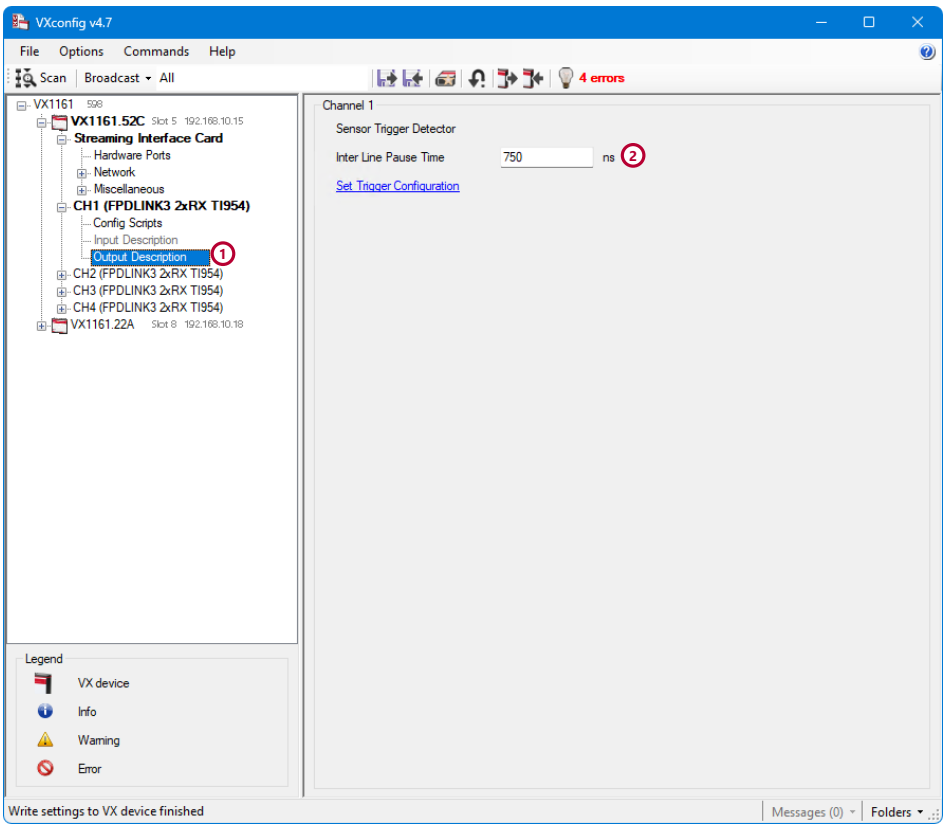


Figure 168: Output Description section in VXconfig

Item	Description
1	In the **Output Description** section, settings for data transmission can be specified.
2	The **Inter Line Pause Time** specifies the time between CSI packets.

Note
The **Output Description** section is only available if your card is configured in TX mode.

4.5.24.3 Radar Interface Card

VX1161.51D modules configured in µC Measurement/ Radar use case (see section Streaming Interface Device on page 411) can ony be used in **RX** mode.

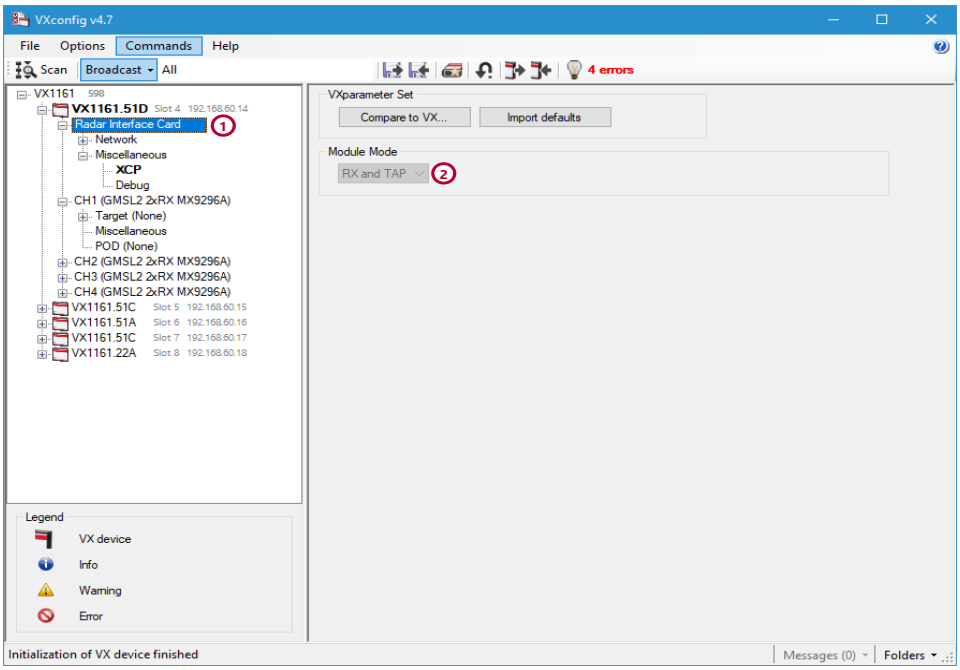


Figure 169: Streaming Module configured as Radar Interface Card in VXconfig

Item	Description
1	Only VX1161.51D can be used as Radar Interface.
2	For µC Measurement/ Radar use case the Module Mode is fixed set to **RX and TAP** and cannot be changed.

4.5.25 Accessories

- Desktop Type Power Supply (page 649)
- Power Cable Molex Mini-Fit 4mm² (page 650)

5 PODs

In this chapter you find the following information:

5.1 VX1452 HSSL POD (obsolete) 431
5.1.1 General Information 431
5.1.2 Connectors 432
5.1.3 LEDs 433
5.1.4 Technical Data 433
5.1.5 Target Interface Connectors 434

5.2 VX1453 HSSL POD 440
5.2.1 General Information 440
5.2.2 Connectors 441
5.2.3 LEDs 442
5.2.4 Technical Data 442
5.2.5 Evalboard Extension Kit 443
5.2.6 Target Interface Connectors 444

5.3 VX1454.01 HSSL POD 456
5.3.1 General Information 456
5.3.2 Connectors 457
5.3.3 LEDs 458
5.3.4 Technical Data 458
5.3.5 Evalboard Extension Kit 459
5.3.6 Target Interface Connectors 460

5.4 VX1461B HSSL POD PCIe 463
5.4.1 General Information 463
5.4.2 Connectors 464
5.4.3 LEDs 465
5.4.4 Technical Data 465
5.4.5 Target Interface Connectors 466

5.5 VX1463.02 HSSL POD AUR+1xRIF 469
5.5.1 General Information 469
5.5.2 Connectors 470
5.5.3 LEDs 471
5.5.4 Technical Data 471
5.5.5 Target Interface Connectors 472

5.6 VX1463.20 HSSL POD AUR+2xRIF 475
5.6.1 General Information 475
5.6.2 Connectors 476
5.6.3 LEDs 477
5.6.4 Technical Data 477

5.6.5 Target Interface Connectors 478

5.7 VX1511.04 HSCT/RHSIF/Zipwire Repeater 481
5.7.1 General Information 481
5.7.2 Connectors 482
5.7.3 LEDs 482
5.7.4 Technical Data 482
5.7.5 Target Interface Connectors 484

5.8 VX1522.01 GMSL2 POD LVDS+DAP 486
5.8.1 General Information 486
5.8.2 Connectors 487
5.8.3 LEDs 488
5.8.4 Technical Data 489
5.8.5 Target Interface Connector 490

5.9 VX1543A Serial POD (Obsolete) 493
5.9.1 General Information 493
5.9.2 Connectors 494
5.9.3 LEDs 494
5.9.4 Technical Data 495
5.9.5 Target Interface Connector 496

5.10 VX1543B Serial POD 502
5.10.1 General Information 502
5.10.2 Connectors 504
5.10.3 LEDs 504
5.10.4 Technical Data 505
5.10.5 Target Interface Connector 506

5.11 VX1543C Serial POD 515
5.11.1 General Information 515
5.11.2 Connectors 516
5.11.3 LEDs 516
5.11.4 Technical Data 517
5.11.5 Target Interface Connector 518

5.12 VX1544A Serial POD 524
5.12.1 General Information 524
5.12.2 Connectors 525
5.12.3 LEDs 526
5.12.4 Technical Data 526
5.12.5 Target Interface Connector 527

5.13 VX1544B Serial POD 535
5.13.1 General Information 535
5.13.2 Connectors 537

5.13.3 LEDs ... 537
5.13.4 Technical Data ... 538
5.13.5 Target Interface Connector ... 539

5.14 VX1544C Serial POD ... 541
5.14.1 General Information ... 541
5.14.2 Connectors ... 543
5.14.3 LEDs ... 543
5.14.4 Technical Data ... 544
5.14.5 Target Interface Connector ... 545

5.15 VX1544D Serial POD ... 547
5.15.1 General Information ... 547
5.15.2 Connectors ... 548
5.15.3 LEDs ... 548
5.15.4 Technical Data ... 549
5.15.5 Target Interface Connector ... 550

5.16 VX1621A XPOD ... 559
5.16.1 General Information ... 559
5.16.2 Connectors ... 560
5.16.3 Technical Data ... 561
5.16.4 Target Interface Connectors ... 562

5.1 VX1452 HSSL POD (obsolete)

5.1.1 General Information

Note
The VX1452 HSSL POD is discontinued.
This means that the POD is still supported by the VXtools and also still gets updates with new features and bug fixes but the hardware is no longer sold.
Drop-in replacement for the VX1452 HSSL POD is the VX1453 HSSL POD.

The VX1452 HSSL POD is the target interface adaptation component of the VX1000 system and has the following features:

- Small form factor eases ECU integration
- Support of microcontrollers with Aurora Data Trace interface
  Aurora: 1-/2- lanes receiver interface with up to 3.125 Gbit/s per lane or 4 lanes receiver interface with up to 1.25 Gbit/s per lane
- Intended use: Customer ECU integration

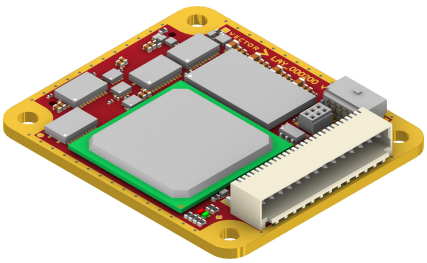


Figure 170: VX1452 HSSL POD

WARNING!
VX1000 Hardware is not designed for high voltage (HV) environments, unless explicitly stated otherwise! If a solution for a HV environment is required, please contact VXsupport@vector.com for guidance.

Note
A detailed list of supported microcontrollers can be found in Supported Micro-controllers on page 19.

Note
The VX1452 HSSL POD is deprecated, and will be discontinued in the future.
For new projects, the VX1453 HSSL POD is recommended as replacement.

5.1.2 Connectors

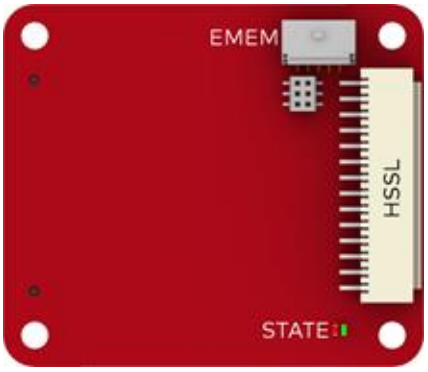


Figure 171: VX1452 top view connectors

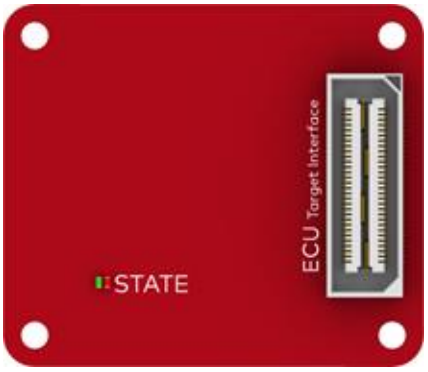


Figure 172: VX1452 bottom view connectors

- **HSSL**
  The HSSL connector supplies power to the POD and allows communication between the VX1452 HSSL POD and the VX1000 Base Module. The connector is compatible with cables of the type VX1379 HSSL Pigtail and VX1381 HSSL2 Pigtail (see section HSSL Cables on page 639).
- **EMEM**
  The EMEM connector is an interface to the optional VX1944 EMEM Power Supply. The latter provides power for the Emulation Memory of a Target Microcontroller and a Calibration Wakeup signal for the ECU.
- **ECU target interface (Samtec QSH-030-01-L-D-A)**
  The ECU target connector provides the interfaces to the target microcontroller and can be directly attached to the ECU (connector on target PCB: QTH-030-xx-L-D-A). The VX1452 HSSL POD can also be connected to the ECU through customer specific interface hardware. Please contact Vector for further information.

Port	Connector on POD	Target Interface type
ECU	QSH-030-01-L-D-A (Samtec)	Aurora

Reference
Further information on the target connector pin assignments can be found in section Target Interface Connectors on page 434.

5.1.3 LEDs

The state of the VX1452 HSSL POD is shown via a green/red LED blink sequence.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off R = red G = green

Sequence	Description
R R R R R R R R	Recovery Image running. Link from VX1000 system active.
R R R R R R R □	Recovery Image running. Link from VX1000 system broken.
G G G G G G G G	Application Image running. Link from VX1000 system active.
G G G G G G G □	Application Image running. Link from VX1000 system broken.
G G G G R R R R	Application Image running. Error detected during system initialization (no measurement possible).

5.1.4 Technical Data

Dimensions (L/W/H)	40 x 34.5 x 9.2 mm
Weight	10 g
Power supply	via VX1000 Base Module
Power consumption	2.3 W at max. operating temperature
Operating temperature range	-40 °C ... +115 °C (max. temperature at FPGA case and SDRAM case)
Part number	22278

Note
Technical drawings can be found at vector.com/vx1000-drawings.

5.1.5 Target Interface Connectors

5.1.5.1 NXP/ST PowerPC

This section describes the pin assignment of the ECU target interface connector between the POD and the ECU. The functional details of the interfaces are described in the section Hardware Integration GuideFor NXP/ST PowerPC micro-controllers, the following interface groups are provided:

- JTAG is mandatory for ECU access and required for basic communication
- AUR (Aurora) is required for DAQ using Data Trace (2 or 4 lanes) with very high DAQ performance and significant low impact on ECU runtime
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	UPOD_EXT	-	-	DNC	-	
3	TCK	JTAG	m	OUT	3V3	
5	TMS	JTAG	m	OUT	3V3	
7	TDI	JTAG	m	OUT	3V3	
9	TDO	JTAG	m	IN	3V3	
11	nTRST/JCOMP	JTAG	(m)	OUT	3V3	Mandatory if available on target micro-controller
13	reserved	-	-	DNC	-	
15	nRDY	JTAG	o	IN	3V3	Recommended for optimized performance*
17	reserved	-	-	DNC	-	
19	RSTOUT	SYS	o	IN	3V3	Strongly recommended to allow software reset detection*
21	reserved	-	-	DNC	-	
23	reserved	-	-	DNC	-	
25	WDGDIS	SYS	o	OUT	3V3	Allows disabling a watchdog on the ECU for flash programming*
27	reserved	-	-	DNC	-	
29	reserved	-	-	DNC	-	
31	reserved	-	-	DNC	-	
33	reserved	-	-	DNC	-	
35	reserved	-	-	DNC	-	
37	reserved	-	-	DNC	-	
39	reserved	-	-	DNC	-	
41	reserved	-	-	DNC	-	
43	reserved	-	-	DNC	-	
45	reserved	-	-	DNC	-	
47	reserved	-	-	DNC	-	
49	reserved	-	-	DNC	-	
51	reserved	-	-	DNC	-	
53	reserved	-	-	DNC	-	
55	reserved	-	-	DNC	-	
57	reserved	-	-	DNC	-	
59	GND			PWR		

* for more information, please refer to the Hardware Integration Guide

Pin	Signal	Group	Need	Dir	I/O	Details
2	reserved	-	-	DNC	-	
4	nRESET	SYS	m	IN/OUT	3V3/5V	Open drain output stage
6	VREF	SYS	o	IN	-	Strongly recommended*
8	reserved	-	-	DNC	-	
10	GND	-	m	PWR	-	
12	AUR TX0p	AUR	m	IN	1.5-V PCML	Differential Aurora data, AC coupled
14	AUR TX0n					
16	GND	-	m	PWR	-	
18	AUR TX1p	AUR	m	IN	1.5-V PCML	Differential Aurora data, AC coupled
20	AUR TX1n					
22	GND	-	m	PWR	-	
24	AUR TX2p	AUR	(m)	IN	1.5-V PCML	Differential Aurora data, AC coupled, required if using Aurora with 4 lanes
26	AUR TX2n					
28	GND	-	m	PWR	-	
30	AUR TX3p	AUR	(m)	IN	1.5-V PCML	Differential Aurora data, AC coupled, required if using Aurora with 4 lanes
32	AUR TX3n					
34	GND	-	m	PWR	-	
36	reserved	-	-	DNC	-	
38	reserved	-	-	DNC	-	
40	GND	-	m	PWR	-	
42	reserved	-	-	DNC	-	
44	reserved	-	-	DNC	-	
46	reserved	-	-	DNC	-	
48	reserved	-	-	DNC	-	
50	reserved	-	-	DNC	-	
52	reserved	-	-	DNC	-	
54	reserved	-	-	DNC	-	
56	reserved	-	-	DNC	-	
58	reserved	-	-	DNC	-	
60	GND	-	m	PWR	-	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

Note
If a target supports one Aurora lane only, it must be connected to AUR_TX0. For more lanes, they must be connected in ascending order on both POD and target side. Swapping of lanes is not possible.

5.1.5.2 Renesas RH850

This section describes the pin assignment of the ECU target interface connector between the POD and the ECU. The functional details of the interfaces are described in the section Hardware Integration Guide

For Renesas RH850 microcontrollers, the following interface groups are provided:

- JTAG is mandatory for ECU access and required for basic communication
- AUD (AUD2/AUD-R) is required for RAM accesses (r/w) with significantly increased performance
- AUR (Aurora) is required for DAQ using Data Trace with very high DAQ performance and significant low impact on ECU runtime
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	UPOD_EXT	-	-	DNC	-	
3	TCK	JTAG	m	OUT	3V3	
5	TMS	JTAG	m	OUT	3V3	
7	TDI	JTAG	m	OUT	3V3	
9	TDO	JTAG	m	IN	3V3	
11	nTRST	JTAG	m	OUT	3V3	
13	reserved	-	-	DNC	-	
15	nRDY	JTAG	m	IN	3V3	
17	reserved	-	-	DNC	-	
19	RSTOUT	SYS	o	IN	3V3	Strongly recommended to allow software reset detection*
21	reserved	-	-	DNC	-	
23	FLMD0	SYS	o	OUT	3V3	Allows ECU flash programming with HSM enabled under certain conditions*
25	WDGDIS	SYS	o	OUT	3V3	Allows disabling an a watchdog on the ECU for flash programming*
27	reserved	-	-	DNC	-	
29	reserved	-	-	DNC	-	
31	reserved	-	-	DNC	-	
33	reserved	-	-	DNC	-	
35	reserved	-	-	DNC	-	
37	reserved	-	-	DNC	-	
39	reserved	-	-	DNC	-	
41	reserved	-	-	DNC	-	
43	AUDATA3	AUD	m	IN	3V3	
45	AUDATA2	AUD	m	IN	3V3	
47	AUDATA1	AUD	m	IN	3V3	
49	AUDATA0	AUD	m	IN	3V3	
51	reserved	-	-	DNC	-	
53	nAUDSYNC	AUD	m	OUT	3V3	
55	nAUDRST	AUD	m	OUT	3V3	
57	AUDCK	AUD	m	OUT	3V3	
59	GND	-	m	PWR	-	

* for more information, please refer to the Hardware Integration Guide

Pin	Signal	Group	Need	Dir	I/O	Details
2	reserved	-	-	DNC	-	
4	nRESET	SYS	m	IN/OUT	3V3/5V	Open drain output stage
6	VREF	SYS	o	IN	-	Strongly recommended*
8	reserved	-	-	DNC	-	
10	GND	-	m	PWR	-	
12	AUR TX0p	AUR	m	IN	1.5-V PCML	Differential Aurora data, AC coupled
14	AUR TX0n					
16	GND	-	m	PWR	-	
18	AUR TX1p	AUR	(m)	IN	1.5-V PCML	Differential Aurora data, AC coupled, required if using Aurora with 2/4 lanes
20	AUR TX1n					
22	GND	-	m	PWR	-	
24	AUR TX2p	AUR	(m)	IN	1.5-V PCML	Differential Aurora data, AC coupled, required if using Aurora with 4 lanes
26	AUR TX2n					
28	GND	-	m	PWR	-	
30	AUR TX3p	AUR	(m)	IN	1.5-V PCML	Differential Aurora data, AC coupled, required if using Aurora with 4 lanes
32	AUR TX3n					
34	GND	-	m	PWR	-	
36	AUR CLKp	AUR	m	OUT	LVDS	Differential Aurora clock, AC coupled, Jitter requirements need to be checked with Vector
38	AUR CLKn					
40	GND	-	m	PWR	-	
42	reserved	-	-	DNC	-	
44	reserved	-	-	DNC	-	
46	reserved	-	-	DNC	-	
48	reserved	-	-	DNC	-	
50	nMSYN	AUR	m	OUT	3V3	Aurora synchronization request
52	nAUR_RST2	AUR	m	OUT	3V3	Aurora domain signal isolation
54	nAUR_RST1	AUR	m	OUT	3V3	Aurora domain reset signal
56	reserved	-	-	DNC	-	
58	reserved	-	-	DNC	-	
60	GND	-	m	PWR	-	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

Note
If a target supports one Aurora lane only, it must be connected to AUR_TX0. For more lanes, they must be connected in ascending order on both POD and target side.

5.2 VX1453 HSSL POD

5.2.1 General Information

The VX1453 HSSL POD is the target interface adaptation component of the VX1000 system and has the following features:

- Small form factor eases ECU integration
- Support of microcontrollers with Aurora Data Trace interface
  Aurora: 1-/2-/4-lanes receiver interface with up to 3.125 Gbit/s per lane
- Support of microcontrollers with Nexus Class 3 Data Trace interface
  NEXUS AUX: up to 16-bit interface, up to 66 MHz MCKO
- Intended use: Customer ECU integration or laboratory environment

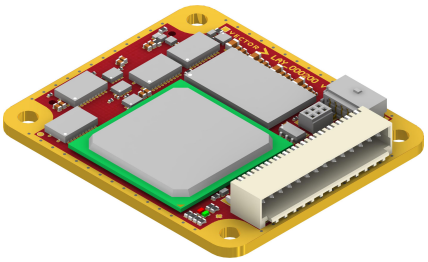


Figure 173: VX1453 HSSL POD

Figure 174: VX1453 with Evalboard Extension Kit

WARNING!
VX1000 Hardware is not designed for high voltage (HV) environments, unless explicitly stated otherwise! If a solution for a HV environment is required, please contact VXsupport@vector.com for guidance.

Note
A detailed list of supported microcontrollers can be found in Supported Micro-controllers on page 19.

5.2.2 Connectors

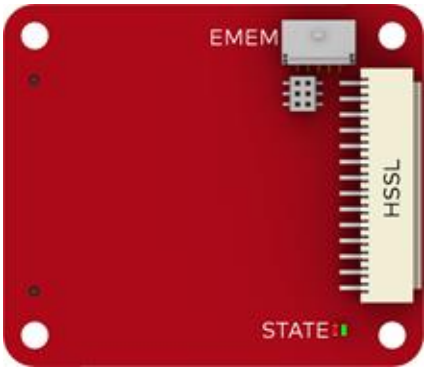


Figure 175: VX1453 top view connectors

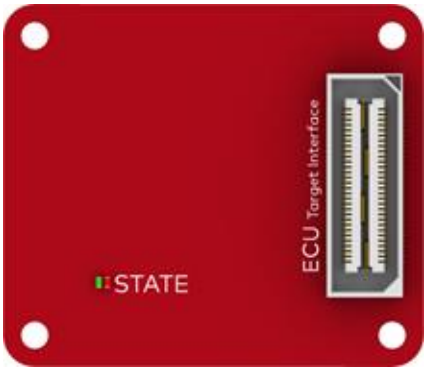


Figure 176: VX1453 bottom view connectors

- **HSSL**
  The HSSL connector supplies power to the POD and allows communication between the VX1453 HSSL POD and the VX1000 Base Module. The connector is compatible with cables of the type VX1379 HSSL Pigtail and VX1381 HSSL2 Pigtail (see section HSSL Cables on page 639).
- **EMEM**
  The EMEM connector is an interface to the optional VX1944 EMEM Power Supply. The latter provides power for the Emulation Memory of a Target Microcontroller and a Calibration Wakeup signal for the ECU.
- **ECU target interface (Samtec QSH-030-01-L-D-A)**
  The ECU target connector provides the interfaces to the target microcontroller either directly (connector on target PCB: QTH-030-xx-L-D-A) or via the Evalboard Extension Kit (see section Evalboard Extension Kit on page 443). The VX1453 HSSL POD can also be connected to the customer ECU through customer specific interface hardware. Please contact Vector for further information.

Port	Connector on POD	Target Interface type
ECU	QSH-030-01-L-D-A (Samtec)	Aurora

Reference
Further information on the target connector pin assignments can be found in section Target Interface Connectors on page 444.

5.2.3 LEDs

The state of the VX1453 HSSL POD is shown via a green/red LED blink sequence.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off R = red G = green

Sequence	Description
R R R R R R R R	Recovery Image running. Link from VX1000 system active.
R R R R R R R □	Recovery Image running. Link from VX1000 system broken.
G G G G G G G G	Application Image running. Link from VX1000 system active.
G G G G G G G □	Application Image running. Link from VX1000 system broken.
G G G G R R R R	Application Image running. Error detected during system initialization (no measurement possible).

5.2.4 Technical Data

Dimensions (L/W/H)	40 x 34.5 x 9 mm
Weight	10 g
Power supply	via VX1000 Base Module
Power consumption	2.3 W at max. operating temperature
Operating temperature range	-40 °C ... +115 °C (max. temperature at FPGA case and SDRAM case) -40 °C ... +85 °C if used with Evalboard Extension Kit
Part number	22219

Note
Technical drawings can be found at vector.com/vx1000-drawings.

5.2.5 Evalboard Extension Kit

The Evalboard Extension Kit is an aluminum enclosure for VX145x HSSL PODs. It is meant for protecting and cooling the VX145x HSSL POD if it is used in a laboratory environment and not built into an ECU housing. A VX1901 EEK Flex PCB followed by a VX1902 EEK Head is attached to the VX145x HSSL POD as interface to evaluation boards. For a detailed description of the different VX1902 EEK Head variants see the corresponding section Adapters on page 582.

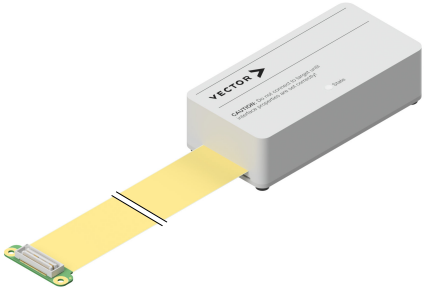


Figure 177: Evalboard Extension Kit incl. VX1901 EEK Flex PCB

Figure 178: VX1902 EEK Head

Various microcontroller types are supported with the Evalboard Extension Kit. The following table gives an overview on the required EEK Heads:

ECU Interface	EEK Heads
Aurora (Infineon)	VX1902.02A (see page 618) VX1902.02B (see page 620)
Aurora (NXP/Renesas)	VX1902.03A (see page 622) VX1902.03B (see page 624)

5.2.6 Target Interface Connectors

5.2.6.1 NXP/ST PowerPC (Aurora)

This section describes the pin assignment of the ECU target interface connector between the POD and the ECU. The functional details of the interfaces are described in the section Hardware Integration Guide

For NXP/ST PowerPC microcontrollers, the following interface groups are provided:

- JTAG is mandatory for ECU access and required for basic communication
- AUR (Aurora) is required for DAQ using Data Trace (2 or 4 lanes) with very high DAQ performance and significant low impact on ECU runtime
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	UPOD_EXT	-	-	DNC	-	
3	TCK	JTAG	m	OUT	3V3	
5	TMS	JTAG	m	OUT	3V3	
7	TDI	JTAG	m	OUT	3V3	
9	TDO	JTAG	m	IN	3V3	
11	nTRST/JCOMP	JTAG	(m)	OUT	3V3	Mandatory if available on target micro-controller
13	reserved	-	-	DNC	-	
15	nRDY	JTAG	o	IN	3V3	Recommended for optimized performance*
17	reserved	-	-	DNC	-	
19	RSTOUT	SYS	o	IN	3V3	Strongly recommended to allow software reset detection*
21	reserved	-	-	DNC	-	
23	reserved	-	-	DNC	-	
25	WDGDIS	SYS	o	OUT	3V3	Allows disabling a watchdog on the ECU for flash programming*
27	reserved	-	-	DNC	-	
29	reserved	-	-	DNC	-	
31	reserved	-	-	DNC	-	
33	reserved	-	-	DNC	-	
35	reserved	-	-	DNC	-	
37	reserved	-	-	DNC	-	
39	reserved	-	-	DNC	-	
41	reserved	-	-	DNC	-	
43	reserved	-	-	DNC	-	
45	reserved	-	-	DNC	-	
47	reserved	-	-	DNC	-	
49	reserved	-	-	DNC	-	
51	reserved	-	-	DNC	-	
53	reserved	-	-	DNC	-	
55	reserved	-	-	DNC	-	
57	reserved	-	-	DNC	-	
59	GND	-	-	PWR	-	

* for more information, please refer to the Hardware Integration Guide

Pin	Signal	Group	Need	Dir	I/O	Details
2	reserved	-	-	DNC	-	
4	nRESET	SYS	m	IN/OUT	3V3/5V	Open drain output stage
6	VREF	SYS	o	IN	-	Strongly recommended*
8	reserved	-	-	DNC	-	
10	GND	-	m	PWR	-	
12	AUR TX0p	AUR	m	IN	1.5-V PCML	Differential Aurora data, AC coupled
14	AUR TX0n					
16	GND	-	m	PWR	-	
18	AUR TX1p	AUR	m	IN	1.5-V PCML	Differential Aurora data, AC coupled
20	AUR TX1n					
22	GND	-	m	PWR	-	
24	AUR TX2p	AUR	(m)	IN	1.5-V PCML	Differential Aurora data, AC coupled, required if using Aurora with 4 lanes
26	AUR TX2n					
28	GND	-	m	PWR	-	
30	AUR TX3p	AUR	(m)	IN	1.5-V PCML	Differential Aurora data, AC coupled, required if using Aurora with 4 lanes
32	AUR TX3n					
34	GND	-	m	PWR	-	
36	AUR CLKp	AUR	m	OUT	LVDS	Differential Aurora clock, AC coupled, Jitter requirements need to be checked with Vector
38	AUR CLKn					
40	GND	-	m	PWR	-	
42	reserved	-	-	DNC	-	
44	reserved	-	-	DNC	-	
46	reserved	-	-	DNC	-	
48	reserved	-	-	DNC	-	
50	reserved	-	-	DNC	-	
52	reserved	-	-	DNC	-	
54	reserved	-	-	DNC	-	
56	reserved	-	-	DNC	-	
58	reserved	-	-	DNC	-	
60	GND	-	m	PWR	-	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

Note
If a target supports one Aurora lane only, it must be connected to AUR_TX0. For more lanes, they must be connected in ascending order on both POD and target side. Swapping of lanes is not possible.

5.2.6.2 NXP/ST PowerPC (NEXUS AUX)

This section describes the pin assignment of the ECU target interface connector between the POD and the ECU. The functional details of the interfaces are described in the section Hardware Integration GuideThe following interface groups are provided:

- JTAG is mandatory for ECU access and required for basic communication
- AUX (Nexus AUX) is required for DAQ using Data Trace (up to 16 bit port width) with very high DAQ performance and insignificant impact on ECU runtime
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	reserved	-	-	DNC	-	
3	TCK	JTAG	m	OUT	3V3	
5	TMS	JTAG	m	OUT	3V3	
7	TDI	JTAG	m	OUT	3V3	
9	TDO	JTAG	m	IN	3V3	
11	nTRST/JCOMP	JTAG	(m)	OUT	3V3	Mandatory if available on target microcontroller
13	reserved	-	-	DNC	-	
15	nRDY	JTAG	o	IN	3V3	Recommended for optimized performance*
17	reserved	-	-	DNC	-	
19	MDO15	AUX	(m)	IN	3V3	Required if using Nexus AUX with a port width of 16 bit.
21	MDO14					
23	MDO13					
25	MDO12					
27	MDO11	AUX	(m)	IN	3V3	Required if using Nexus AUX with a port width of 12 bit or more.
29	MDO10					
31	MDO9					
33	MDO8					
35	MDO7	AUX	(m)	IN	3V3	Required if using Nexus AUX with a port width of 8 bit or more.
37	MDO6					
39	MDO5					
41	MDO4					
43	MDO3	AUX	m	IN	3V3	
45	MDO2					
47	MDO1					
49	MDO0					
51	nMSEO1	AUX	m	OUT	3V3	
53	nMSEO0	AUX	m	IN	3V3	
55	GND	-	m	PWR	-	
57	MCKO	AUX	m	IN	3V3	
59	GND	-	m	PWR	-	

* for more information, please refer to the Hardware Integration Guide

Pin	Signal	Group	Need	Dir	I/O	Details
2	reserved	-	-	DNC	-	
4	nRESET	SYS	m	IN/OUT	3V3/5V	Open drain output stage
6	VREF	SYS	o	IN	-	Strongly recommended*
8	reserved	-	-	DNC	-	
10	reserved	-	-	DNC	-	
12	nRDY	JTAG	o	IN	3V3	Recommended for optimized performance*
14	reserved	-	-	DNC	-	
16	reserved	-	-	DNC	-	
18	reserved	-	-	DNC	-	
20	reserved	-	-	DNC	-	
22	reserved	-	-	DNC	-	
24	reserved	-	-	DNC	-	
26	reserved	-	-	DNC	-	
28	reserved	-	-	DNC	-	
30	reserved	-	-	DNC	-	
32	reserved	-	-	DNC	-	
34	reserved	-	-	DNC	-	
36	reserved	-	-	DNC	-	
38	reserved	-	-	DNC	-	
40	reserved	-	-	DNC	-	
42	reserved	-	-	DNC	-	
44	reserved	-	-	DNC	-	
46	reserved	-	-	DNC	-	
48	reserved	-	-	DNC	-	
50	reserved	-	-	DNC	-	
52	reserved	-	-	DNC	-	
54	reserved	-	-	DNC	-	
56	reserved	-	-	DNC	-	
58	reserved	-	-	DNC	-	
60	GND	-	m	PWR		

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

5.2.6.3 Infineon Aurix

This section describes the pin assignment of the ECU target interface connector between the POD and the ECU. The functional details of the interfaces are described in the section Hardware Integration Guide

For Infineon Aurix microcontrollers, the following interface groups are provided:

- DAP is mandatory ECU access and required for basic communication as well as moderate DAQ performance
- AUR (Aurora) is required for DAQ using Data Trace with very high DAQ performance and significant low impact on ECU runtime
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	UPOD_EXT	-	-	DNC	-	
3	DAP0	DAP	m	OUT	3V3	
5	DAP1	DAP	m	IN/OUT	3V3	
7	reserved	-	-	DNC	-	
9	DAP2	DAP	m	IN/OUT	3V3	
11	DAPEN	DAP	m	OUT	3V3	
13	reserved	-	-	DNC	-	
15	CRC_ERR	AUR	o	OUT	3V3	Allows Aurora error detection and retransmit
17	reserved	-	-	DNC	-	
19	RSTOUT	SYS	o	IN	3V3	Strongly recommended to allow software reset detection*
21	reserved	-	-	DNC	-	
23	reserved	-	-	DNC	-	
25	WDGDIS	SYS	o	OUT	3V3	Allows disabling a watchdog on the ECU for flash programming*
27	reserved	-	-	DNC	-	
29	reserved	-	-	DNC	-	
31	reserved	-	-	DNC	-	
33	reserved	-	-	DNC	-	
35	reserved	-	-	DNC	-	
37	reserved	-	-	DNC	-	
39	reserved	-	-	DNC	-	
41	reserved	-	-	DNC	-	
43	reserved	-	-	DNC	-	
45	reserved	-	-	DNC	-	
47	reserved	-	-	DNC	-	
49	reserved	-	-	DNC	-	
51	reserved	-	-	DNC	-	
53	reserved	-	-	DNC	-	
55	reserved	-	-	DNC	-	
57	reserved	-	-	DNC	-	
59	GND	-	m	PWR	-	

* for more information, please refer to the Hardware Integration Guide

Pin	Signal	Group	Need	Dir	I/O	Details
2	reserved	-	-	DNC	-	
4	nRESET	SYS	m	IN/OUT	3V3/5V	Open drain output stage*
6	VREF	SYS	m	IN	-	
8	reserved	-	-	DNC	-	
10	GND	-	m	PWR	-	
12	AUR TX0p	AUR	m	IN	1.5-V PCML	Differential Aurora data, AC coupled
14	AUR TX0n					
16	GND	-	m	PWR	-	
18	reserved			DNC	-	
20	reserved			DNC	-	
22	GND	-	m	PWR	-	
24	reserved			DNC	-	
26	reserved			DNC	-	
28	GND	-	m	PWR	-	
30	reserved			DNC	-	
32	reserved			DNC	-	
34	GND	-	m	PWR	-	
36	AUR CLKp	AUR	m	OUT	LVDS	Differential Aurora clock, AC coupled, Jitter requirements need to be checked with Vector
38	AUR CLKn					
40	GND	-	m	PWR	-	
42	reserved	-	-	DNC	-	
44	reserved	-	-	DNC	-	
46	reserved	-	-	DNC	-	
48	reserved	-	-	DNC	-	
50	reserved	-	-	DNC	-	
52	reserved	-	-	DNC	-	
54	reserved	-	-	DNC	-	
56	reserved	-	-	DNC	-	
58	reserved	-	-	DNC	-	
60	GND	-	m	PWR	-	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

Note
If a target supports one Aurora lane only, it must be connected to AUR_TX0. For more lanes, they must be connected in ascending order on both POD and target side.

5.2.6.4 Renesas RH850

This section describes the pin assignment of the ECU target interface connector between the POD and the ECU. The functional details of the interfaces are described in the section Hardware Integration Guide

For Renesas RH850 microcontrollers, the following interface groups are provided:

- JTAG is mandatory for ECU access and required for basic communication.
- AUD (AUD2/AUD-R) is required for RAM accesses (r/w) with significantly increased performance
- AUR (Aurora) is required for DAQ using Data Trace (1, 2 or 4 lanes) with very high DAQ performance and significant low impact on ECU runtime
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	UPOD_EXT	-	-	DNC	-	
3	TCK	JTAG	m	OUT	3V3	
5	TMS	JTAG	m	OUT	3V3	
7	TDI	JTAG	m	OUT	3V3	
9	TDO	JTAG	m	IN	3V3	
11	nTRST	JTAG	m	OUT	3V3	
13	reserved	-	-	DNC	-	
15	nRDY	JTAG	m	IN	3V3	
17	reserved	-	-	DNC	-	
19	RSTOUT	SYS	o	IN	3V3	Strongly recommended to allow soft reset detection*
21	reserved	-	-	DNC	-	
23	FLMD0	SYS	o	OUT	3V3	Allows ECU flash programming with HSM enabled under certain conditions*
25	WDGDIS	SYS	o	OUT	3V3	Allows disabling an external watchdog for flash programming*
27	reserved	-	-	DNC	-	
29	reserved	-	-	DNC	-	
31	reserved	-	-	DNC	-	
33	reserved	-	-	DNC	-	
35	reserved	-	-	DNC	-	
37	reserved	-	-	DNC	-	
39	reserved	-	-	DNC	-	
41	reserved	-	-	DNC	-	
43	AUDATA3	AUD	m	IN	3V3	
45	AUDATA2	AUD	m	IN	3V3	
47	AUDATA1	AUD	m	IN	3V3	
49	AUDATA0	AUD	m	IN	3V3	
51	reserved	-	-	DNC	-	
53	nAUDSYNC	AUD	m	OUT	3V3	
55	nAUDRST	AUD	m	OUT	3V3	
57	AUDCK	AUD	m	OUT	3V3	
59	GND	-	m	PWR	-	

* for more information, please refer to the Hardware Integration Guide

Pin	Signal	Group	Need	Dir	I/O	Details
2	reserved	-	-	DNC	-	
4	nRESET	SYS	m	IN/OUT	3V3/5V	Open drain output stage
6	VREF	SYS	o	IN	-	Strongly recommended*
8	reserved	-	-	DNC	-	
10	GND	-	m	PWR	-	
12	AUR TX0p	AUR	m	IN	1.5-V PCML	Differential Aurora data, AC coupled
14	AUR TX0n					
16	GND	-	m	PWR	-	
18	AUR TX1p	AUR	(m)	IN	1.5-V PCML	Differential Aurora data, AC coupled, required if using Aurora with 2/4 lanes
20	AUR TX1n					
22	GND	-	m	PWR	-	
24	AUR TX2p	AUR	(m)	IN	1.5-V PCML	Differential Aurora data, AC coupled, required if using Aurora with 4 lanes
26	AUR TX2n					
28	GND	-	m	PWR	-	
30	AUR TX3p	AUR	(m)	IN	1.5-V PCML	Differential Aurora data, AC coupled, required if using Aurora with 4 lanes
32	AUR TX3n					
34	GND	-	m	PWR	-	
36	AUR CLKp	AUR	m	OUT	LVDS	Differential Aurora clock, AC coupled, Jitter requirements need to be checked with Vector
38	AUR CLKn					
40	GND	-	m	PWR	-	
42	reserved	-	-	DNC	-	
44	reserved	-	-	DNC	-	
46	reserved	-	-	DNC	-	
48	reserved	-	-	DNC	-	
50	nMSYN	AUR	m	OUT	3V3	Aurora synchronization request
52	nAUR_RST2	AUR	m	OUT	3V3	Aurora domain signal isolation
54	nAUR_RST1	AUR	m	OUT	3V3	Aurora domain reset signal
56	reserved	-	-	DNC	-	
58	reserved	-	-	DNC	-	
60	GND	-	m	PWR	-	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

Note
If a target supports one Aurora lane only, it must be connected to AUR_TX0. For more lanes, they must be connected in ascending order on both POD and target side. Swapping of lanes is not possible.

5.3 VX1454.01 HSSL POD

5.3.1 General Information

The VX1454.01 HSSL POD is the target interface adaptation component of the VX1000 system and has the following features:

- Small form factor eases ECU integration
- Support of Renesas RH850E2 with Aurora Data Trace interface
  Aurora: 1-/2-/4 lanes receiver interface, up to 5 Gbit/s per lane
- Intended use: Customer ECU integration or laboratory environment

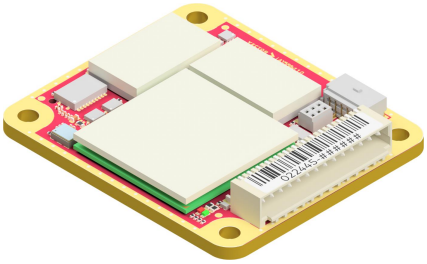


Figure 179: VX1454.01 HSSL POD

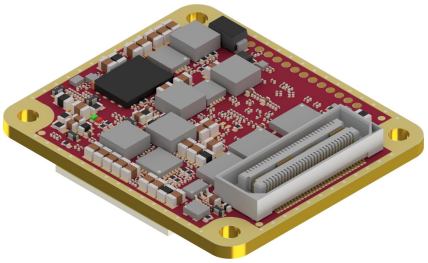


Figure 180: VX1454.01 bottom view

WARNING!
VX1000 Hardware is not designed for high voltage (HV) environments, unless explicitly stated otherwise! If a solution for a HV environment is required, please contact VXsupport@vector.com for guidance.

Note
A detailed list of supported microcontrollers can be found in Supported Microcontrollers on page 19.

5.3.2 Connectors

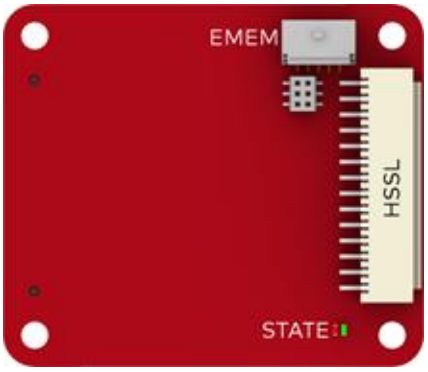


Figure 181: VX1454.01 top view connectors

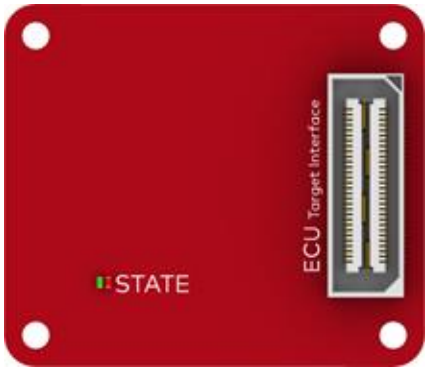


Figure 182: VX1454.01 bottom view connectors

- **HSSL**
  The HSSL connector supplies power to the POD and allows communication between the VX1454.01 HSSL POD and the VX1000 Base Module. The connector is compatible with cables of the type VX1379 HSSL Pigtail and VX1381 HSSL2 Pigtail (see section HSSL Cables on page 639).
- **EMEM**
  The EMEM connector is an interface to the optional VX1944 EMEM Power Supply. The latter provides power for the Emulation Memory of a Target Microcontroller and a Calibration Wakeup signal for the ECU.
- **ECU target interface (Samtec QSH-030-01-L-D-A)**
  The ECU target connector provides the interfaces to the target microcontroller either directly (connector on target PCB: QTH-030-xx-L-D-A) or via the Evalboard Extension Kit (see section Evalboard Extension Kit on page 459). The VX1454.01 HSSL POD can also be connected to the customer ECU through customer specific interface hardware. Please contact Vector for further information.

Port	Connector on POD	Target Interface type
ECU	QSH-030-01-L-D-A (Samtec)	Aurora

Reference
Further information on the target connector pin assignments can be found in section Target Interface Connectors on page 460.

5.3.3 LEDs

The state of the VX1454.01 HSSL POD is shown via a green/red LED blink sequence.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off R = red G = green

Sequence	Description
R R R R R R R R	Recovery Image running. Link from VX1000 system active.
R R R R R R R □	Recovery Image running. Link from VX1000 system broken.
G G G G G G G G	Application Image running. Link from VX1000 system active.
G G G G G G G □	Application Image running. Link from VX1000 system broken.
G G G G R R R R	Application Image running. Error detected during system initialization (no measurement possible).

5.3.4 Technical Data

Dimensions (L/W/H)	40 x 34.5 x 9 mm
Weight	14 g
Power supply	via VX1000 Base Module
Power consumption	2.6 W at max. operating temperature
Operating temperature range	-40 °C ... +115 °C (max. temperature at FPGA case or SDRAM case) -40 °C ... +85 °C if used with Evalboard Extension Kit
Part number	22445

Note
Technical drawings can be found at vector.com/vx1000-drawings.

5.3.5 Evalboard Extension Kit

The Evalboard Extension Kit is an aluminum enclosure for VX145x HSSL PODs. It is meant for protecting and cooling the VX145x HSSL POD if it is used in a laboratory environment and not built into an ECU housing. A VX1901 EEK Flex PCB followed by a VX1902 EEK Head is attached to the VX145x HSSL POD as interface to evaluation boards. For a detailed description of the different VX1902 EEK Head variants see the corresponding section Adapters on page 582.

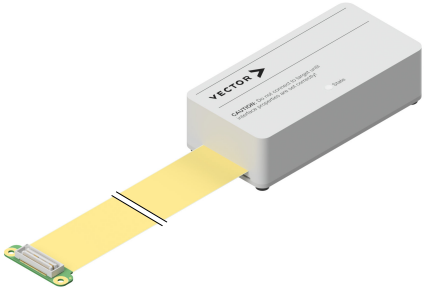


Figure 183: Evalboard Extension Kit incl. VX1901 EEK Flex PCB

Figure 184: VX1902 EEK Head

Various microcontroller types are supported with the Evalboard Extension Kit. The following table gives an overview on the required EEK Heads:

ECU Interface	EEK Heads
Aurora (NXP/Renesas)	VX1902.03A (see page 622) VX1902.03B (see page 624)

5.3.6 Target Interface Connectors

5.3.6.1 Renesas RH850

This section describes the pin assignment of the ECU target interface connector between the POD and the ECU.The functional details of the interfaces are described in the section Hardware Integration Guide

For Renesas RH850 microcontrollers, the following interface groups are provided:

- JTAG is mandatory for ECU access and required for basic communication
- AUD (AUD2/AUD-R) is optional for RAM accesses (r/w) with significantly increased performance
- AUR (Aurora) is optional for DAQ using Data Trace with very high DAQ performance and significant low impact on ECU runtime
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	UPOD_EXT	-	-	DNC	-	
3	TCK	JTAG	m	OUT	3V3	
5	TMS	JTAG	m	OUT	3V3	
7	TDI	JTAG	m	OUT	3V3	
9	TDO	JTAG	m	IN	3V3	
11	nTRST	JTAG	m	OUT	3V3	
13	reserved	-	-	DNC	-	
15	nRDY	JTAG	m	IN	3V3	
17	reserved	-	-	DNC	-	
19	RSTOUT	SYS	o	IN	3V3	Strongly recommended to allow software reset detection*
21	reserved	-	-	DNC	-	
23	FLMD0	SYS	o	OUT	3V3	Allows ECU flash programming with HSM enabled under certain conditions*
25	WDGDIS	SYS	o	OUT	3V3	Allows disabling a watchdog on the ECU for flash programming*
27	reserved	-	-	DNC	-	
29	reserved	-	-	DNC	-	
31	reserved	-	-	DNC	-	
33	reserved	-	-	DNC	-	
35	reserved	-	-	DNC	-	
37	reserved	-	-	DNC	-	
39	reserved	-	-	DNC	-	
41	reserved	-	-	DNC	-	
43	AUDATA3	AUD	m	IN	3V3	
45	AUDATA2	AUD	m	IN	3V3	
47	AUDATA1	AUD	m	IN	3V3	
49	AUDATA0	AUD	m	IN	3V3	
51	reserved	-	-	DNC	-	
53	nAUDSYNC	AUD	m	OUT	3V3	
55	nAUDRST	AUD	m	OUT	3V3	

Pin	Signal	Group	Need	Dir	I/O	Details
57	AUDCK	AUD	m	OUT	3V3	
59	GND	-	m	PWR	-	

* for more information, please refer to the Hardware Integration Guide

Pin	Signal	Group	Need	Dir	I/O	Details
2	reserved	-	-	DNC	-	
4	nRESET	SYS	m	IN/OUT	3V3/5V	Open drain output stage
6	VREF	SYS	o	IN	-	Strongly recommended*
8	reserved	-	-	DNC	-	
10	GND	-	m	PWR	-	
12	AUR TX0p	AUR	m	IN	1.5-V PCML	Differential Aurora data, AC coupled
14	AUR TX0n					
16	GND	-	m	PWR	-	
18	AUR TX1p	AUR	m	IN	1.5-V PCML	Differential Aurora data, AC coupled, required if using Aurora with 2/4 lanes
20	AUR TX1n					
22	GND	-	m	PWR	-	
24	AUR TX2p	AUR	m	IN	1.5-V PCML	Differential Aurora data, AC coupled, required if using Aurora with 4 lanes
26	AUR TX2n					
28	GND	-	m	PWR	-	
30	AUR TX3p	AUR	m	IN	1.5-V PCML	Differential Aurora data, AC coupled, required if using Aurora with 4 lanes
32	AUR TX3n					
34	GND	-	m	PWR	-	
36	AUR CLKp	AUR	m	OUT	LVDS	Differential 125 MHz Aurora clock, AC coupled
38	AUR CLKn					
40	GND	-	m	PWR	-	
42	reserved	-	-	DNC	-	
44	reserved	-	-	DNC	-	
46	reserved	-	-	DNC	-	
48	reserved	-	-	DNC	-	
50	nMSYN	AUR	m	OUT	3V3	Aurora synchronization request
52	nAUR_RST2	AUR	m	OUT	3V3	Aurora domain signal isolation
54	nAUR_RST1	AUR	m	OUT	3V3	Aurora domain reset signal
56	reserved	-	-	DNC	-	
58	reserved	-	-	DNC	-	
60	GND	-	m	PWR	-	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

5.4 VX1461B HSSL POD PCIe

5.4.1 General Information

The VX1461B HSSL POD PCIe is the target interface adaptation component of the VX1000 system and has the following features:

- Small form factor eases ECU integration
- Support of Infineon microcontrollers with Aurora Data Trace interface
  Aurora: 1 lane receiver interface, up to 3.125 Gbit/s
- Support of 1 PCI-Express Gen1 x1 interface
- Intended use: Customer ECU integration

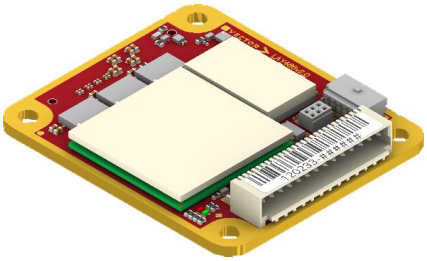


Figure 185: VX1461B HSSL POD PCIe

WARNING!
VX1000 Hardware is not designed for high voltage (HV) environments, unless explicitly stated otherwise! If a solution for a HV environment is required, please contact VXsupport@vector.com for guidance.

5.4.2 Connectors

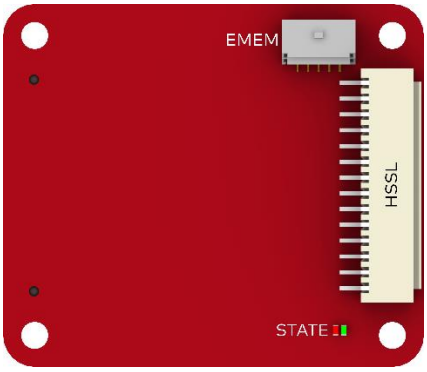


Figure 186: VX1461B top view connectors

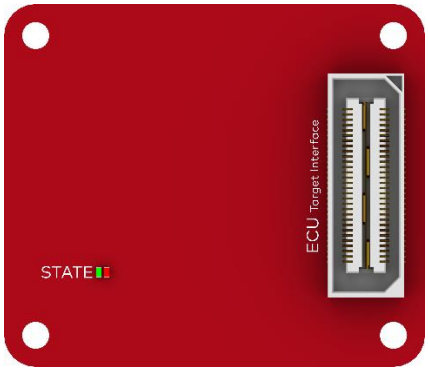


Figure 187: VX1461B bottom view connectors

- **HSSL**
  The HSSL connector supplies power to the POD and allows communication between the VX1461B HSSL POD PCIe and the VX1000 Base Module. The connector is compatible with VX1381 HSSL2 Pigtails.
- **EMEM**
  The EMEM connector is an interface to the optional VX1944 EMEM Power Supply. The latter provides power for the Emulation Memory of a Target Microcontroller and a Calibration Wakeup signal for the ECU.
- **ECU target interface (Samtec QSH-030-01-L-D-A)**
  The ECU target connector provides the interfaces to the target microcontroller and can be directly attached to the ECU (connector on target PCB: QTH-030-x-x-D-A-x). The VX1461B HSSL POD PCIe can also be connected to the ECU through customer specific interface hardware. Please contact Vector for further information.

Port	Connector on POD	Target Interface type
ECU	QSH-030-01-L-D-A (Samtec)	Aurora and PCIe

Reference
Further information on the target connector pin assignments can be found in section Target Interface Connectors on page 466.

5.4.3 LEDs

The state of the VX1461B HSSL POD PCIe is shown via a green/red LED blink sequence.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off R = red G = green

Sequence	Description
R R R R R R R R	Recovery Image running. Link from VX1000 system active.
R R R R R R R □	Recovery Image running. Link from VX1000 system broken.
G G G G G G G G	Application Image running. Link from VX1000 system active.
G G G G G G G □	Application Image running. Link from VX1000 system broken.
G G G G R R R R	Application Image running. Error detected during system initialization (no measurement possible).

5.4.4 Technical Data

Dimensions (L/W/H)	40 x 34.5 x 9 mm
Weight	10 g
Power supply	via VX1000 Base Module
Power consumption	2.5 W at max. operating temperature
Operating temperature range	-40 °C ... +115 °C (max. temperature at FPGA case or SDRAM case)
Part number	120233

5.4.5 Target Interface Connectors

5.4.5.1 General SoC-Architectures / Infineon Aurix

This section describes the pin assignment of the ECU target interface connector between the POD and the ECU. The functional details of the interfaces are described in the section Hardware Integration Guide

For Infineon Aurix microcontrollers, the following interface groups are provided:

- PCIe provides general access for measurment and calibration of SoC-systems
- ETH provides general access for measurment and calibration of SoC-systems
- AUR (Aurora) is optional provided for DAQ using Data Trace for micro-controllers on the ECU supporting this interface
- DAP is optional provided for for basic communication as well as moderate DAQ performance for Infineon microcontrollers on the ECU
- SYS (System) signals are additionally provided for realization of different mea-surement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	UPOD EXT	-	-	DNC	-	
3	reserved	-	-	DNC	-	
5	VREF	SYS	m	IN	-	
7	GND	-	m	PWR	-	
9	DAP0	DAP	m	OUT	3V3	
11	GND	-	m	PWR	-	
13	DAP1	DAP	m	IN/OUT	3V3	
15	GND	-	m	PWR	-	
17	DAP2	DAP	m	IN/OUT	3V3	
19	GND	-	m	PWR	-	
21	DAPEN	DAP	m	OUT	3V3	4.75 kΩ pull-up to VREF on POD side
23	CRC_ERR	AUR	o	OUT	3V3	Allows Aurora error detection
25	reserved	-	-	DNC	-	
27	RSTOUT	SYS	o	IN	3V3	Strongly recommended to allow soft-ware reset detection*
29	EMEM_PWRG	SYS	o	IN	3V3	
31	WDGDIS	SYS	o	OUT	3V3	Allows disabling a watchdog on the ECU for flash programming*
33	nPERST	PCIe	m	IN	3V3	
35	nPRESENT	PCIe	m	OUT	3V3	
37	reserved	-	-	DNC	-	
39	reserved	-	-	DNC	-	
41	reserved	-	-	DNC	-	
43	reserved	-	-	DNC	-	
45	INIT_OK	PCIe	m	IN	3V3	
47	GND	-	m	PWR	-	
49	reserved	-	-	DNC	-	
51	GND	-	m	PWR	-	
53	DAPE1	DAP	m	IN/OUT	3V3	
55	GND	-	m	PWR	-	
57	DAPE0	DAP	m	OUT	3V3	
59	GND	-	m	PWR	-	

* for more information, please refer to the Hardware Integration Guide

Pin	Signal	Group	Need	Dir	I/O	Details
2	reserved	-	-	DNC	-	
4	nRESET	SYS	m	IN/OUT	3V3/5V	Open drain output stage*
6	GND	-	m	PWR	-	
8	AUR_CLK_p	AUR	m	OUT	LVDS	Differential 100 MHz Aurora clock, AC coupled
10	AUR_CLK_n					
12	GND	-	m	PWR	-	
14	PCIE_TX0_p	PCIe	m	IN	1.5-V PCML	Differential PCIe data from ECU to VX1000, AC coupled
16	PCIE_TX0_n					
18	GND	-	m	PWR	-	
20	PCIE_RX0_p	PCIe	m	OUT	1.5-V PCML	Differential PCIe data from VX1000 to ECU, AC coupled
22	PCIE_RX0_n					
24	GND	-	m	PWR	-	
26	reserved	-	-	DNC	-	
28	reserved	-	-	DNC	-	
30	GND	-	m	PWR	-	
32	reserved	-	-	DNC	-	
34	reserved	-	-	DNC	-	
36	GND	-	m	PWR	-	
38	AUR_TX0_p	AUR	m	IN	1.5-V PCML	Differential Aurora data, AC coupled
40	AUR_TX0_n					
42	GND	-	m	PWR	-	
44	SGMII_RX_p	SGMII	m	OUT	1.5-V PCML	Differential SGMII data from VX1000 to ECU, AC coupled
46	SGMII_RX_n					
48	GND	-	m	PWR	-	
50	SGMII_TX_p	SGMII	m	IN	1.5-V PCML	Differential SGMII data from ECU to VX1000, AC coupled
52	SGMII_TX_n					
54	GND	-	m	PWR	-	
56	REFCLK_p	PCIe	m	IN	LVDS	Differential PCIe clock from ECU to VX1000, AC coupled
58	REFCLK_n					
60	GND	-	m	PWR	-	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig** otherwise the POD may be damaged. This restriction does not apply to pins listed as "5 V tolerant".

Note
When mating with another ASP-192273-04 connector, pin 01 will mate with contact 02 on mating connector.

Note
If a target supports one Aurora lane only, it must be connected to AUR_TX0. For more lanes, they must be connected in ascending order on both POD and target side.

5.5 VX1463.02 HSSL POD AUR+1xRIF

5.5.1 General Information

The VX1463.02 HSSL POD AUR+1xRIF is the target interface adaptation component of the VX1000 system and has the following features:

- Small form factor eases ECU integration
- Support of Infineon microcontrollers with Aurora Data Trace interface
  Aurora: 1 lane receiver interface, up to 3.125 Gbit/s
- Support of 1 Infineon MMIC with RIF interface
  RIF: 4 LVDS data lanes receiver, up to 400Mbit/s per lane
- Intended use: Customer ECU integration

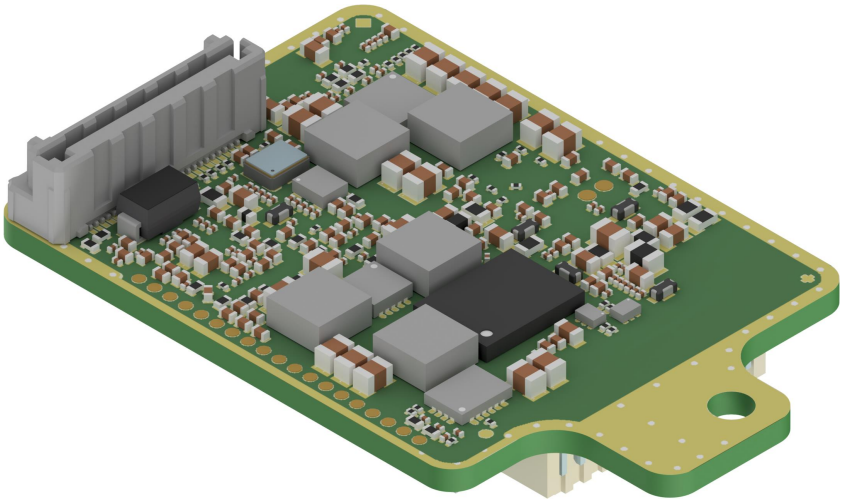


Figure 188: VX1463.02 HSSL POD AUR+1xRIF

WARNING!
VX1000 Hardware is not designed for high voltage (HV) environments, unless explicitly stated otherwise! If a solution for a HV environment is required, please contact VXsupport@vector.com for guidance.

Note
A detailed list of supported microcontrollers can be found in Supported Microcontrollers on page 19.

5.5.2 Connectors

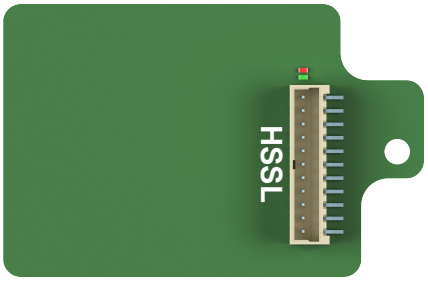


Figure 189: VX1463.02 top view connectors

Figure 190: VX1463.02 bottom view connectors

- **HSSL**
  The HSSL connector supplies power to the POD and allows communication between the VX1463.02 HSSL POD AUR+1xRIF and the VX1000 Base Module. The connector is compatible with VX1381 HSSL2 Pigtails.
- **ECU target interface (Samtec ASP-192273-04)**
  The ECU target connector provides the interfaces to the target microcontroller and can be directly attached to the ECU (connector on target PCB: ASP-192273-04). The VX1463.02 HSSL POD AUR+1xRIF can also be connected to the ECU through customer specific interface hardware. Please contact Vector for further information.

Port	Connector	Target Interface type
ECU	ASP-192273-04 (Samtec)	Aurora and RIF

Reference
Further information on the target connector pin assignments can be found in section Target Interface Connectors on page 472.

5.5.3 LEDs

The state of the VX1463.02 HSSL POD AUR+1xRIF is shown via a green/red LED blink sequence.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off R = red G = green

Sequence	Description
R R R R R R R R	Recovery Image running. Link from VX1000 system active.
R R R R R R R □	Recovery Image running. Link from VX1000 system broken.
G G G G G G G G	Application Image running. Link from VX1000 system active.
G G G G G G G □	Application Image running. Link from VX1000 system broken.
G G G G R R R R	Application Image running. Error detected during system initialization (no measurement possible).

5.5.4 Technical Data

Dimensions (L/W/H)	47 x 30.5 x 13 mm
Weight	10 g
Power supply	from VX1000 Base Module
Power consumption	≈ 2.3 W (typical, DDR3 RAM enabled) ≈ 1.9 W (typical, DDR3 RAM disabled)
Temperature range	-40 °C ... +105 °C (appropriate cooling solution required)
Part number	120272

Note
Technical drawings can be found at vector.com/vx1000-drawings.

5.5.5 Target Interface Connectors

5.5.5.1 Infineon Aurix

This section describes the pin assignment of the ECU target interface connector between the POD and the ECU. The functional details of the interfaces are described in the section Hardware Integration Guide

For Infineon Aurix microcontrollers, the following interface groups are provided:

- DAP is mandatory ECU access and required for basic communication as well as moderate DAQ performance
- AUR (Aurora) is optional for DAQ using Data Trace with very high DAQ performance and significant low impact on ECU runtime
- RIF is provided for DAQ of Radar applications.
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	m	IN	-	
3	DAPEN	DAP	m	OUT	3V3	4.75 kΩ pull-up to VREF on POD side
5	GND	-	m	PWR	-	
7	DAP2	DAP	m	IN/OUT	3V3	DAP data
9	DAP1	DAP	m	IN/OUT	3V3	DAP data
11	GND	-	m	PWR	-	
13	DAP0	DAP	m	DNC	3V3	DAP clock
15	GND	-	m	PWR	-	
17	reserved	-	-	DNC	-	
19	reserved	-	-	DNC	-	
21	reserved	-	-	DNC	-	
23	reserved	-	-	DNC	-	
25	GND	-	m	PWR	-	
27	RIF DATA1 p	RIF	m	IN	LVDS	Differential RIF data
29	RIF DATA1 n					
31	reserved	-	-	DNC	-	
33	RIF FRAME p	RIF	m	IN	LVDS	Differential RIF frame clock
35	RIF FRAME n					
37	reserved	-	-	DNC	-	
39	RIF DATA3 p	RIF	m	IN	LVDS	Differential RIF data
41	RIF DATA3 n					
43	GND	-	m	PWR	-	
45	AUR TX0 n	AUR	m	IN	1.5-V PCML	Differential Aurora data, AC coupled
47	AUR TX0 p					
49	GND	-	m	IN	-	

* for more information, please refer to the Hardware Integration Guide

Pin	Signal	Group	Need	Dir	I/O	Details
2	reserved	-	-	DNC	-	
4	reserved	-	-	DNC	-	
6	GND	-	m	PWR	-	
8	nRESET	SYS	m	IN/OUT	3V3/5V	Open drain output stage.
10	nLVDS_ACTIVE	SYS	(m)	IN	3V3	Required if RIF clock is not continuous*
12	GND	-	m	PWR	-	
14	WDGDIS	SYS	o	OUT	3V3	Allows disabling a watchdog on the ECU for flash programming*
16	GND	-	m	PWR	-	
18	reserved	-	-	DNC	-	
20	reserved	-	-	DNC	-	
22	CRC_ERR	AUR	o	IN	3V3	Allows Aurora error detection and retransmit
24	reserved	-	-	DNC	-	
26	GND	-	-	PWR	-	
28	RIF DATA2 p	RIF	m	IN	LVDS	Differential RIF data
30	RIF DATA2 n					
32	GND	-	m	PWR	-	
34	RIF CLK p	RIF	m	IN	LVDS	Differential RIF clock
36	RIF CLK n					
38	GND	-	m	PWR	-	
40	RIF DATA4 p	RIF	m	IN	LVDS	Differential RIF data
42	RIF DATA4 n					
44	GND	-	m	PWR	-	
46	AUR CLK p	AUR	m	OUT	LVDS	Differential 100 MHz Aurora clock, AC coupled
48	AUR CLK n					
50	GND	-	m	PWR	-	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig** otherwise the POD may be damaged. This restriction does not apply to pins listed as "5 V tolerant".

Note
When mating with another ASP-192273-04 connector, pin 01 will mate with contact 02 on mating connector.

5.6 VX1463.20 HSSL POD AUR+2xRIF

5.6.1 General Information

The VX1463.20 HSSL POD AUR+2xRIF is the target interface adaptation component of the VX1000 system and has the following features:

- Small form factor eases ECU integration
- Support of Infineon microcontrollers with Aurora Data Trace interface
  Aurora: 1 lane receiver interface, up to 3.125 Gbit/s
- Support of 2 Infineon MMICs with RIF interface
  RIF per MMIC: 4 LVDS data lanes receiver, up to 400Mbit/s per lane
- Intended use: Customer ECU integration

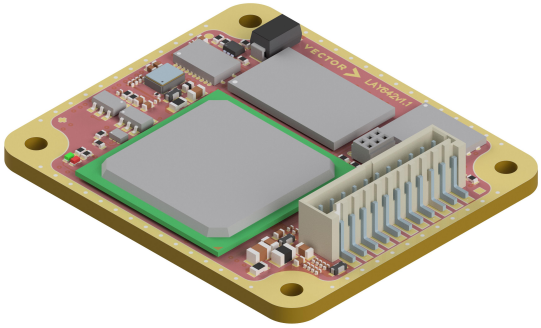


Figure 191: VX1463.20 HSSL POD AUR+2xRIF

WARNING!
VX1000 Hardware is not designed for high voltage (HV) environments, unless explicitly stated otherwise! If a solution for a HV environment is required, please contact VXsupport@vector.com for guidance.

Note
A detailed list of supported microcontrollers can be found in Supported Microcontrollers on page 19.

5.6.2 Connectors

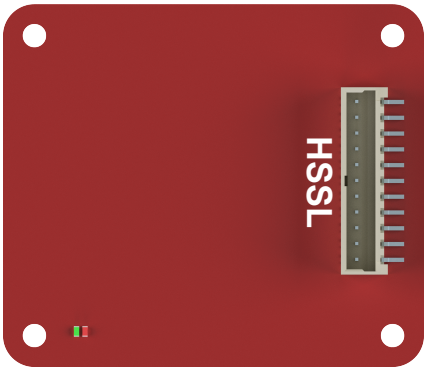


Figure 192: VX1463.20 top view connectors

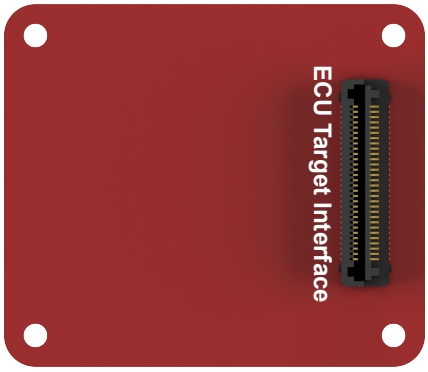


Figure 193: VX1463.20 bottom view connectors

- **HSSL**
  The HSSL connector supplies power to the POD and allows communication between the VX1463.20 HSSL POD AUR+2xRIF and the VX1000 Base Module. The connector is compatible with VX1381 HSSL2 Pigtails.

- **ECU target interface (Samtec ASP-211442-01)**
  The ECU target connector provides the interfaces to the target microcontroller and can be directly attached to the ECU (connector on target PCB: ASP-211442-01). The VX1463.20 HSSL POD AUR+2xRIF can also be connected to the ECU through customer specific interface hardware. Please contact Vector for further information.

Port	Connector	Target Interface type
ECU	ASP-211442-01 (Samtec)	Aurora and RIF

Reference
Further information on the target connector pin assignments can be found in section Target Interface Connectors on page 478.

5.6.3 LEDs

The state of the VX1463.20 HSSL POD AUR+2xRIF is shown via a green/red LED blink sequence.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off R = red G = green

Sequence	Description
R R R R R R R R	Recovery Image running. Link from VX1000 system active.
R R R R R R R □	Recovery Image running. Link from VX1000 system broken.
G G G G G G G G	Application Image running. Link from VX1000 system active.
G G G G G G G □	Application Image running. Link from VX1000 system broken.
G G G G R R R R	Application Image running. Error detected during system initialization (no measurement possible).

5.6.4 Technical Data

Dimensions (L/W/H)	40 x 34.5 x 13 mm
Weight	10 g
Power supply	from VX1000 Base Module
Power consumption	≈ 2.6 W (typical, DDR3 RAM enabled) ≈ 2.2 W (typical, DDR3 RAM disabled)
Temperature range	-40 °C ... +105 °C (appropriate cooling solution required)
Part number	120273

Note
Technical drawings can be found at vector.com/vx1000-drawings.

5.6.5 Target Interface Connectors

5.6.5.1 Infineon Aurix

This section describes the pin assignment of the ECU target interface connector between the POD and the ECU. The functional details of the interfaces are described in the section Hardware Integration Guide

For Infineon Aurix microcontrollers, the following interface groups are provided:

- DAP is mandatory ECU access and required for basic communication as well as moderate DAQ performance
- AUR (Aurora) is optional for DAQ using Data Trace with very high DAQ performance and significant low impact on ECU runtime
- RIF is provided for DAQ of Radar applications.
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	reserved	-	-	DNC	-	
3	reserved	-	-	DNC	-	
5	GND	-	m	PWR		
7	nRESET	SYS	m	IN/OUT	3V3/5V	Open drain output stage
9	nLVDS_ACTIVE	SYS	(m)	IN	3V3	Required if LVDS clock is not continuous*
11	WDGDIS	SYS	o	OUT	3V3	Allows disabling a watchdog on the ECU for flash programming*
13	GND	-	m	PWR	-	
15	reserved	-	-	DNC	-	
17	reserved	-	-	DNC	-	
19	GND	-	m	PWR	-	
21	AUR CLK p	AUR	m	OUT	LVDS	Differential 100MHz Aurora clock, AC coupled
23	AUR CLK n					
25	GND	-	m	PWR	-	
27	RIF1 DATA1 p	RIF	m	IN	LVDS	Differential RIF data (MMIC 1)
29	RIF1 DATA1 n					
31	GND	-	m	PWR	-	
33	RIF1 DATA3 p	RIF	m	IN	LVDS	Differential RIF data (MMIC 1)
35	RIF1 DATA3 n					
37	GND	-	m	PWR	-	
39	RIF1 FRAME p	RIF	m	IN	LVDS	Differential RIF frame clock (MMIC 1)
41	RIF1 FRAME n					
43	GND	-	m	PWR	-	
45	RIF2 DATA1 p	RIF	m	IN	LVDS	Differential RIF data (MMIC 2)
47	RIF2 DATA1 n					
49	GND	-	m	PWR	-	
51	RIF2 DATA3 p	RIF	m	IN	LVDS	Differential RIF data (MMIC 2)
53	RIF2 DATA3 n					
55	GND	-	m	PWR	-	
57	RIF2 FRAME p	RIF	m	IN	LVDS	Differential RIF frame clock (MMIC 2)
59	RIF2 FRAME n					

* for more information, please refer to the Hardware Integration Guide

Pin	Signal	Group	Need	Dir	I/O	Details
2	VREF	SYS	m	IN	-	
4	DAPEN	DAP	m	OUT	3V3	4.75 kΩ pull-up to VREF on POD side
6	GND	-	m	PWR	-	
8	DAP1	DAP	m	IN/OUT	3V3	DAP data
10	DAP2	DAP	m	IN/OUT	3V3	DAP data
12	GND	-	m	PWR	-	
14	DAP0	DAP	m	OUT	3V3	DAP clock
16	GND	-	m	PWR	-	
18	CRC_ERR	AUR	o	OUT	3V3	Allows Aurora error detection and retransmit
20	GND	-	m	PWR	-	
22	AUR TX0 p	AUR	m	IN	1.5-V PCML	Differential Aurora data, AC coupled
24	AUR TX0 n					
26	GND	-	m	PWR	-	
28	RIF1 DATA0 p	RIF	m	IN	LVDS	Differential RIF data (MMIC 1)
30	RIF1 DATA0 n					
32	GND	-	m	PWR	-	
34	RIF1 DATA2 p	RIF	m	IN	LVDS	Differential RIF data (MMIC 1)
36	RIF1 DATA2 n					
38	GND	-	m	PWR	-	
40	RIF1 CLK p	RIF	m	IN	LVDS	Differential RIF clock (MMIC 1)
42	RIF1 CLK n					
44	GND	-	m	PWR	-	
46	RIF2 DATA0 p	RIF	m	IN	LVDS	Differential RIF data (MMIC 2)
48	RIF2 DATA0 n					
50	GND	-	m	PWR	-	
52	RIF2 DATA2 p	RIF	m	IN	LVDS	Differential RIF data (MMIC 2)
54	RIF2 DATA2 n					
56	GND	-	m	PWR	-	
58	RIF2 CLK p	RIF	m	IN	LVDS	Differential RIF clock (MMIC 2).
60	RIF2 CLK n					

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD

Pin direction	Description
IN/OUT	Input/Output of the POD
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage. Does not apply to pins listed as "5 V tolerant".

Note
When mating with another ASP-211442-01 connector, Pin 01 will mate with contact 02 on mating connector.

5.7 VX1511.04 HSCT/RHSIF/Zipwire Repeater

5.7.1 General Information

The VX1511.04 HSCT/Zipwire Repeater is an active adapter connecting the VX1000 Base Module to a target microcontroller using the HSCT/Zipwire/RHSIF interface and has the following features:

- Small form factor eases ECU integration
- Support of Infineon microcontrollers with HSCT interface
- Support of ST/NXP microcontrollers with Zipwire interface
- Support of Renesas microcontrollers with RHSIF interface
- Interface speed: Up to 320 MHz
- Intended use: Customer ECU integration or laboratory environment

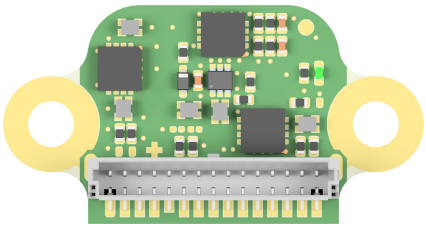


Figure 194: VX1511.04 HSCT/Zipwire Repater

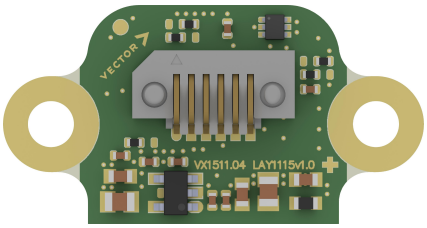


Figure 195: VX1511.04 bottom view

WARNING!
VX1000 Hardware is not designed for high voltage (HV) environments, unless explicitly stated otherwise! If a solution for a HV environment is required, please contact VXsupport@vector.com for guidance.

Note
A detailed list of supported microcontrollers can be found in Supported Microcontrollers on page 19.

5.7.2 Connectors

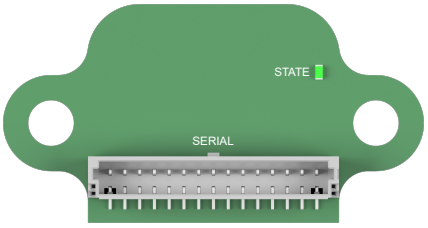


Figure 196: VX1511.04 top view connectors

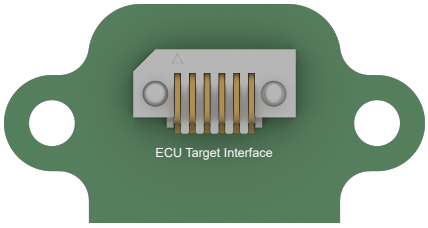


Figure 197: VX1511.04 bottom view connectors

- **Serial**
  The Serial connector supplies power to the POD and allows communication between the VX1511.04 HSCT/Zipwire Repeater and the VX1000 Base Module. The connector is compatible with the VX1311 Serial Pigtails for VX1544x/VX1543B Serial PODs (see section VX1311 Serial2 Cables on page 659).
- **ECU target connector (Samtec ASP-179922-03)**
  The ECU target connector provides the interface to the microcontroller.

Port	Connector on POD	Target Interface Type
ECU	ASP-179922-03 (Samtec)	HSCT/Zipwire/RHSIF

Reference
Further information on the target connector pin assignments can be found in section Target Interface Connectors on page 484.

5.7.3 LEDs

The state of the VX1511.04 HSCT/Zipwire Repeater is shown via a green LED. The following block icons are used to describe this LED status:
☐ = off G = green

LED Status	Description
☐	Repeater is not powered.
G	Repeater is powered up.

5.7.4 Technical Data

Dimensions (L/W/H)	25.3 x 14.8 x 8.5 mm
Weight	2 g
Power supply	via VX1000 Base Module
Power consumption	0.8 W at max. operating temperature
Operating temperature range	-40 °C ... +125 °C
Part Number	22569

Note
Technical drawings can be found at vector.com/vx1000-drawings.

5.7.5 Target Interface Connectors

5.7.5.1 Infineon HSCT, PowerPC Zipwire

This section describes the pin assignment of the ECU target interface connector between the Repeater and the ECU. The functional details of the interfaces are described in the section Hardware Integration Guide The following interface groups are provided:

- HSCT (HSCT/Zipwire/RHSIF) is mandatory for ECU access and provides the DAQ functionality.

Pin	Signal	Group	Need	Dir	I/O	Details
1	REFCLK	HSCT	m	IN	3V3/5V	HSCT reference clock, provided by the ECU*
2	GND	-	m	PWR	-	
3	TXD p	HSCT	m	IN	LVDS	HSCT data from the ECU to the VX1000*
4	TXD n					
5	RXD p	HSCT	m	OUT	LVDS	HSCT data from the VX1000 to the ECU*
6	RXD n					

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

5.8 VX1522.01 GMSL2 POD LVDS+DAP

5.8.1 General Information

The VX1522.01 GMSL2 POD LVDS+DAP is the target interface adaptation of the VX1161.51D/.52D 4xRX GMSL2 MX9296 Streaming Interface Card. It is intended to be mount inside an electronic control unit (ECU) for measuring high bandwidth LVDS data streams and accessing CPU debug interfaces.

The generic receive hardware provides several LVDS and single ended signals to connect to the target device. With different firmwares the hardware can be adapted to different target interface protocols. High performance data processing features, such as filtering or conversion, can be added. Via the single ended signals CPU debug interfaces such as DAP, SWD, JTAG, ... can be accessed.

The generic hardware has the following key features:

- Support of one LVDS receiver interface; 1 clock and up to 5 lanes; at up to 1000 Mbit/s per lane and 5 Gbit/s total; details on request
- Up to 8 single ended signals; up to 160 MHz
- Up to 5 Gbit/s total data rate towards VX1161.51D/.52D
- Supplied by VX1161.51D/.52D; no current consumed from ECU
- Small form factor eases ECU integration

Examples of supported interfaces:

- TI AWR49xx radar LVDS
- Calterah S244, S344 radar LVDS
- Infineon microcontrollers with DAP interface
- Contact Vector for further devices and details.

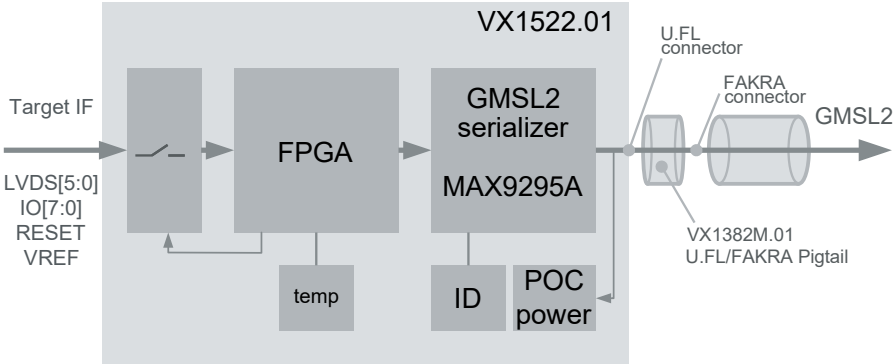


Figure 198: VX1522.01 Block diagram

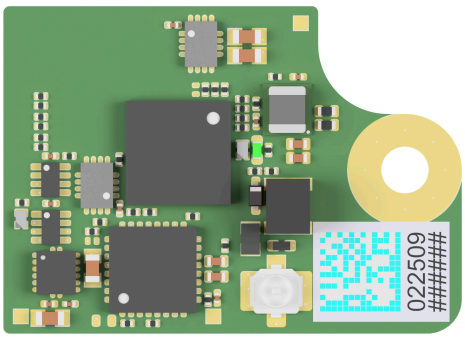


Figure 199: VX1522.01 GMSL2 POD LVDS+DAP

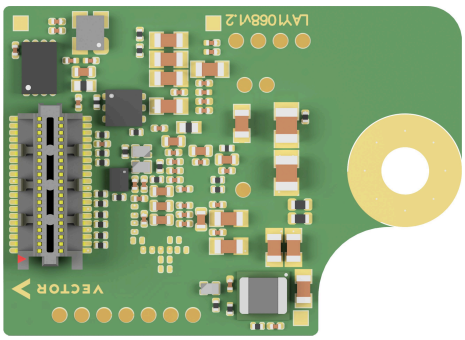


Figure 200: VX1522.01 bottom view

WARNING!
VX1000 Hardware is not designed for high voltage (HV) environments, unless explicitly stated otherwise! If a solution for a HV environment is required, please contact VXsupport@vector.com for guidance.

WARNING!
Unplugging the U.FL coaxial connector without extraction tool Hirose U.FL-LP-N-2 might damage either the connector on the PCB or on the cable!

5.8.2 Connectors

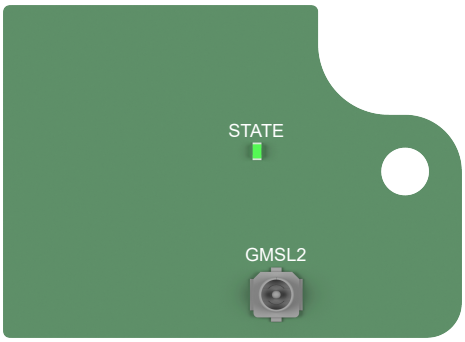


Figure 201: VX1522.01 top view connectors

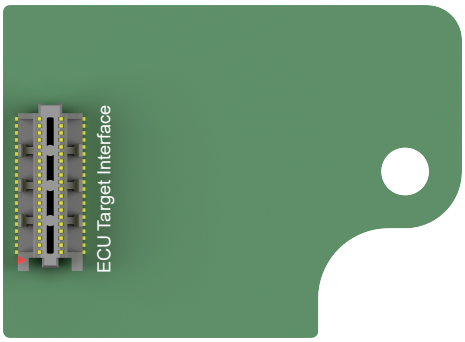


Figure 202: VX1522.01 bottom view connectors

- **ECU target interface (Samtec ST4-20-2.50-L-D)**
  The ECU connector provides the interfaces to the target and is usually directly plugged into the mating connector on the ECU PCB (mating connector on target PCB: SS4-20-x.xx-L-D). For project specific adaptations, please contact Vector.

- **GMSL2 (U.FL)**
  The U.FL connector supplies power to the POD and allows communication between the VX1522.01 GMSL2 POD and the VX1000 Base Module. The connector is compatible with the VX1382.Mx FAKRA Panelmount (see section FAKRA Cables on page 636). Unplugging the U.FL coaxial connector without extraction tool Hirose U.FL-LP-N-2 might damage either the connector on the PCB or on the cable.

Port	Connector on POD	Target Interface type
ECU	ST4-20-2.50-L-D (Samtec)	DAP/RIF

Reference
Further information on the target connector pin assignments can be found in section Connectors on page 487.

5.8.3 LEDs

The state of the VX1522.01 GMSL2 POD is shown via a green LED blink sequence.

The LED indicator shows a recurring sequence of blink codes. The following block icons are used to describe this sequence:
□ = off G = green

Sequence	Description
□□□□□□□□	POD not powered or disabled. Link from VX1000 system broken.
G G G G G G G G	POD initialization. Link from VX1000 system initializing.
G G G G □□□□	POD running, no data traffic. Link from VX1000 system active.
G □ G □ G □ G □	POD running, LVDS data traffic. Link from VX1000 system active.

5.8.4 Technical Data

Dimensions (L/W/H)	26.2 x 19.0 x 8.8 mm
Weight	2.5 g
Power supply	via VX1000 Base Module
Power consumption	1.8 W at max. operating temperature
Operating temperature range	-40 °C ... +100 °C (max. temperature at FPGA case, appropriate cooling solution required; contact Vector if a wider range is required)
Part Number	22509

Note
Technical drawings can be found at vector.com/vx1000-drawings.

5.8.5 Target Interface Connector

5.8.5.1 Infineon Aurix

This section describes the pin assignment of the ECU target interface connector between the POD and the ECU. The functional details of the interfaces are described in the section Hardware Integration Guide For Infineon Aurix micro-controllers, the following interface groups are provided:

- DAP is mandatory for ECU access and required for basic communication.
- RIF (LVDS) is optional for capturing radar raw data over LVDS (2 or 4 lanes) from an MMIC.
- SYS (System) signals are additionally provided for realization of different measurement and calibration features.

Pin	Signal	Group	Need	Dir	I/O	Details
1	GND	-	m	PWR	-	
3	RIF CLK p	RIF	m	IN	LVDS	Differential RIF clock
5	RIF CLK n					
7	GND	-	m	PWR	-	
9	RIF DATA0 p	RIF	m	IN	LVDS	Differential RIF data
11	RIF DATA0 n					
13	GND	-	m	PWR	-	
15	RIF DATA1 p	RIF	m	IN	LVDS	Differential RIF data
17	RIF DATA1 n					
19	GND	-	m	PWR	-	
21	DAP0	DAP	m	OUT	3V3	DAP clock
23	GND	-	m	PWR	-	
25	DAP1	DAP	m	IN	3V3	DAP data
27	GND	-	m	PWR	-	
29	nRSTOUT	SYS	o	IN/OUT	3V3	Strongly recommended to allow soft reset detection*
31	WDGDIS	SYS	o	OUT	3V3	Allows disabling a watchdog on the ECU for flash programming*
33	GND	-	m	IN	-	
35	VREF	SYS	o	IN	3V3	Strongly recommended*
37	GND	-	m	PWR	-	
39	reserved	-	-	DNC	-	

* for more information, please refer to the Hardware Integration Guide

Pin	Signal	Group	Need	Dir	I/O	Details
2	GND	-	-	PWR	-	
4	RIF FRAME p	RIF	m	IN	LVDS	Differential RIF frame clock
6	RIF FRAME n					
8	GND	-	-	PWR	-	
10	RIF DATA3 p	RIF	(m)	IN	LVDS	Differential RIF data, required if using RIF with 4 lanes
12	RIF DATA3 n					
14	GND	-	-	PWR	-	
16	RIF DATA2 p	RIF	(m)	IN	LVDS	Differential RIF data, required if using RIF with 4 lanes
18	RIF DATA2 n					
20	GND	-	-	PWR	-	
22	DAPEN	DAP	m	OUT	3V3	4.75 kΩ pull-up to VREF on POD side
24	DAP2	DAP	m	IN/OUT	3V3	DAP data
26	GND	-	-	PWR	-	
28	reserved	-	-	DNC	-	
30	GND		m	PWR	-	
32	reserved	-	-	DNC	-	
34	nLVDS_ACTIVE	SYS	(m)	IN	3V3	Required if LVDS clock is not continuous*
36	GND	-	m	PWR	-	
38	nRESET	SYS	m	IN/OUT	3V3	Open drain output stage
40	GND	-	-	PWR	-	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

5.9 VX1543A Serial POD (Obsolete)

5.9.1 General Information

The VX1543A Serial POD is the target interface adaptation component of the VX1000 system and has the following features:

- Support of the following microcontroller interfaces:
  - Infineon DAP up to 80 MHz
  - NXP/ST PowerPC JTAG (Nexus Class 2+)
  - Renesas RH850 JTAG (Nexus Class 2+)
- Supports target interface voltages of 3.3 V and 5 V
- Very small form factor, very low power dissipation and automotive-graded
- Intended use: Customer ECU integration and laboratory environment

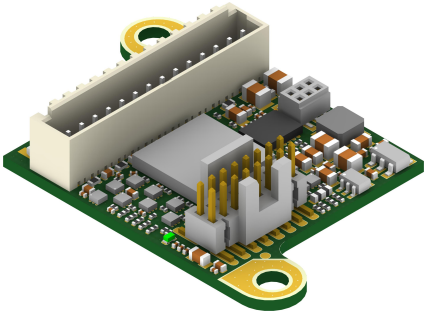


Figure 203: VX1543A Serial POD

WARNING!
VX1000 Hardware is not designed for high voltage (HV) environments, unless explicitly stated otherwise! If a solution for a HV environment is required, please contact VXsupport@vector.com for guidance.

WARNING!
Always ensure that the POD target interface is correctly configured or disabled before connecting a POD to the target. Wrong settings can damage both systems.

WARNING!
The Serial POD must be handled with care when physically attaching to or detaching from a target! Neither the POD nor the target cables nor the VX1927 Flex-PCBs are designed for frequent mating cycles!

Note
A detailed list of supported microcontrollers can be found in Supported Microcontrollers on page 19.

Note
The VX1543A Serial POD is deprecated, and will be discontinued in the future. Please contact vx-support@vector.com to find a proper replacement for your project.

5.9.2 Connectors

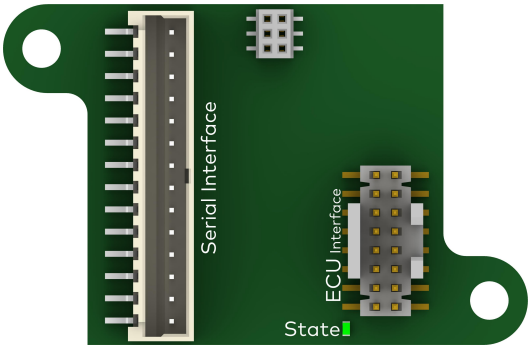


Figure 204: VX1543A top view connectors

- **Serial**
  The Serial connector supplies power to the POD and allows communication between the VX1543A and the VX1000 Base Module. The connector is compatible with cables of the type VX134xA Serial Pigtail (see section VX134x Serial Cables on page 657).
- **ECU**
  The VX1543A can access a microcontroller of a customer ECU or a Development/Evaluation board via the target ECU interface connector.

Port	Connector	Target Interface type
ECU	16-pin Samtec	DAP (2-pin mode) JTAG (Nexus Class 2+)

5.9.3 LEDs

The state of the Serial POD is shown via a green/red LED blink sequence.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off R = red G = green

Sequence	Description
□□□□GGGG	ECU interface disabled.
□G□G□G□G	POD in configuration mode.
GGGGGGGG	ECU interface configured.

Note
The LED behavior has changed since VXtools 4.4. Please see VX1000 Manual V4.3 or earlier for the former LED sequence description.

5.9.4 Technical Data

Dimensions (L/W/H)	35.4 x 23 x 7 mm
Weight	3 g
Power supply	via VX1000 Base Module
Power consumption	0.8 W at max. operating temperature
Operating temperature range	-40 °C ... +115 °C (max. temperature at FPGA case)
Part number	22216

Note
Technical drawings can be found at vector.com/vx1000-drawings.

5.9.5 Target Interface Connector

5.9.5.1 Infineon DAP

ECU (DAP-IFX)

The following table shows the pin assignment of the ECU connector of the VX1543A configured for DAP-IFX. The functional details of the interfaces are described in the section Hardware Integration Guide The pin direction is shown as seen from the POD. For microcontrollers with DAP, the following interface groups are provided:

- DAP is mandatory for ECU access and provides the DAQ functionality
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	IN	-	Strongly recommended*
2	DAP1	DAP	m	IN/OUT	3V3/5V	
3	GND	-	m	PWR	-	
4	DAP0	DAP	m	OUT	3V3/5V	
5	GND	-	m	PWR	-	
6	reserved	-	-	DNC	-	
7	GND	-	m	PWR	-	
8	DAPEN	DAP	m	OUT	3V3/5V	4.75 kΩ pull-up to VREF on POD side
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	3V3/5V	
11	reserved	-	-	DNC	-	
12	WDGDIS	SYS	o	OUT	3V3/5V	Allows disabling an external watchdog for flash programming*
13	reserved	-	-	DNC	-	
14	reserved	-	-	DNC	-	
15	VREF	SYS	o	IN	-	Connected to pin 1 on POD side
16	nRSTOUT	SYS	o	IN	3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD

Pin direction	Description
IN/OUT	Input/Output of the POD
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

Note
FTSH connectors are non-locking and therefore must be fixed if a vibration-proof connection is required. If the VX1543A is ordered with a sealed housing, all connectors on the VX1543A are fixed.

Target cable

For the Infineon DAP configuration a target cable is required which connects the appropriate 10 pins of the 16-pin connector on the VX1543A Serial POD to the 10-pin connector on the target side.

The Infineon DAP target cable is delivered together with the VX1543A Serial POD if it was ordered with Infineon DAP option.

Target (DAP-IFX)

On the target microcontroller side a Samtec FTSH-105-01-L-DV-K-P is required. The pin assignment for Infineon DAP is shown in the following table. The pin direction is shown as seen from the target.

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	OUT	-	Strongly recommended*
2	DAP1	DAP	m	IN/OUT	3V3/5V	
3	GND	-	m	PWR	-	
4	DAP0	DAP	m	IN	3V3/5V	
5	GND	-	m	PWR	-	
6	reserved	-	-	DNC	-	
7	GND	-	m	PWR	-	
8	DAPEN	DAP	m	IN	3V3/5V	4.75 kΩ pull-up to VREF on POD side
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Pin direction	Description
IN	Input of the ECU
OUT	Output of the ECU
IN/OUT	Input/Output of the ECU
PWR	Power supply
DNC	Do not connect

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The VX1543A contains a pull-up resistor of 4.7 kΩ on the DAPEN (nTRST) pin to VREF. During power-up the ECU microcontroller needs a valid logic high level on the nTRST pin to enable the DAP Interface. This must be kept in mind when designing a pull-down resistor on nTRST of the target ECU.

5.9.5.2 JTAG

ECU (JTAG)

The following table shows the pin assignment of the ECU connector of the VX1543A configured for JTAG mode. The functional details of the interfaces are described in the section Hardware Integration Guide The pin direction is shown as seen from the POD. For microcontrollers with JTAG, the following interface groups are provided:

- JTAG is mandatory for ECU access and provides the DAQ functionality
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	IN	-	Strongly recommended*
2	TMS	JTAG	m	OUT	3V3/5V	
3	GND	-	m	PWR	-	
4	TCK	JTAG	m	OUT	3V3/5V	
5	GND	-	m	PWR	-	
6	TDO	JTAG	m	IN	3V3/5V	
7	GND	-	m	PWR	-	
8	TDI	JTAG	m	OUT	3V3/5V	
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	3V3/5V	
11	reserved	-	-	DNC	-	
12	WDGDIS	SYS	o	OUT	3V3/5V	Allows disabling an external watchdog for flash programming*
13	nRDY	JTAG	o	IN	3V3/5V	Recommended for optimized performance*
14	nTRST/JCOMP	JTAG	(m)	OUT	3V3/5V	Mandatory if available on target microcontroller
15	VREF	SYS	o	IN	-	Connected to pin 1 on POD side
16	nRSTOUT	SYS	o	IN	3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD

Pin direction	Description
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

Note
FTSH connectors are non-locking and therefore must be fixed if a vibration-proof connection is required. If the VX1543A is ordered with a sealed housing, all connectors on the VX1543A are fixed.

Target cable

For the JTAG configuration a target cable is required which connects the appropriate 16 pins of the 16-pin connector on the VX1543A to the 16-pin connector on the target side.

The JTAG target cable is delivered together with the VX1543A if it was ordered with JTAG option.

Target (JTAG)

On the target microcontroller side a Samtec FTSH-108-01-L-DV-K-P is required. The pin assignment for JTAG is shown in the following table. The pin direction is shown as seen from the target.

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	OUT	-	Strongly recommended*
2	TMS	JTAG	m	IN	3V3/5V	
3	GND	-	m	PWR	-	
4	TCK	JTAG	m	IN	3V3/5V	
5	GND	-	m	PWR	-	
6	TDO	JTAG	m	OUT	3V3/5V	
7	GND	-	m	PWR	-	
8	TDI	JTAG	m	IN	3V3/5V	
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	3V3/5V	
11	reserved	-	-	DNC	-	
12	WDGDIS	SYS	o	IN	3V3/5V	Allows disabling an external watch-dog for flash programming*
13	nRDY	JTAG	o	OUT	3V3/5V	Recommended for optimized per-formance*
14	nTRST/JCOMP	JTAG	(m)	IN	3V3/5V	Mandatory if available on target microcontroller
15	VREF	SYS	o	OUT	-	Connected to pin 1 on POD side
16	nRSTOUT	SYS	o	OUT	3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Pin direction	Description
IN	Input of the ECU
OUT	Output of the ECU
IN/OUT	Input/Output of the ECU
PWR	Power supply
DNC	Do not connect

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

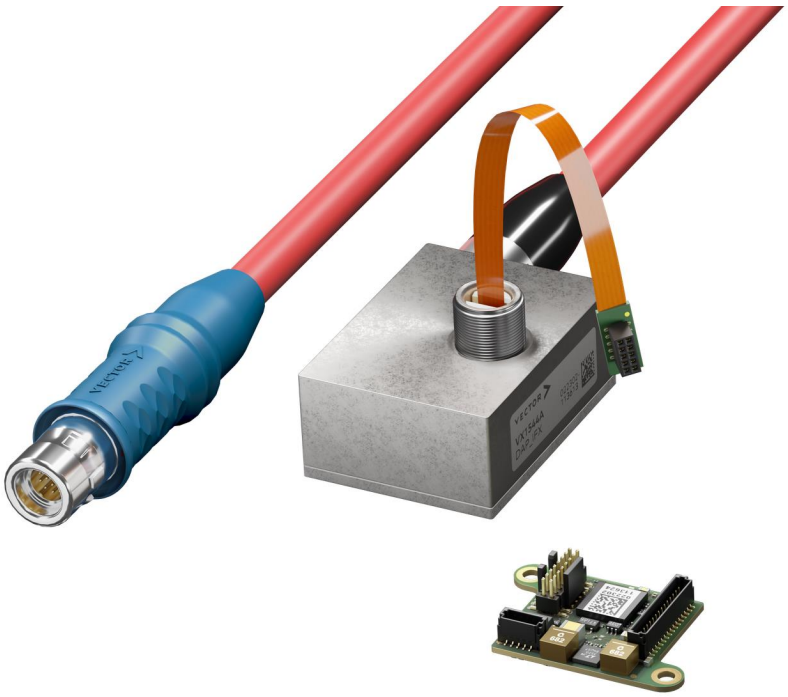
I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

5.10 VX1543B Serial POD

5.10.1 General Information

The VX1543B Serial POD is the target interface adaptation component of the VX1000 system and has the following features:

- Support of the following microcontroller interfaces:
  - Infineon DAP up to 80 MHz
  - NXP/ST PowerPC JTAG (Nexus Class 2+)
  - Renesas RH850 JTAG (Nexus Class 2+)
  - ARM SWD
  - ARM JTAG
- Supports target interface voltages of 1.8 V, 3.3 V and 5 V
- Very small form factor, very low power dissipation and automotive-graded
- Intended use: Customer ECU integration and laboratory environment



VX1543B Serial POD

WARNING!
VX1000 Hardware is not designed for high voltage (HV) environments, unless explicitly stated otherwise! If a solution for a HV environment is required, please contact VXsupport@vector.com for guidance.

WARNING!
Always ensure that the POD target interface is correctly configured or disabled before connecting a POD to the target. Wrong settings can damage both systems.

WARNING!
The Serial POD must be handled with care when physically attaching to or detaching from a target! Neither the POD nor the target cables nor the VX1927 Flex-PCBs are designed for frequent mating cycles!

WARNING!
Opening the enclosure lid may compromise the seal, even if the lid is reattached.

Note
A detailed list of supported microcontrollers can be found in Supported Microcontrollers on page 19.

5.10.2 Connectors

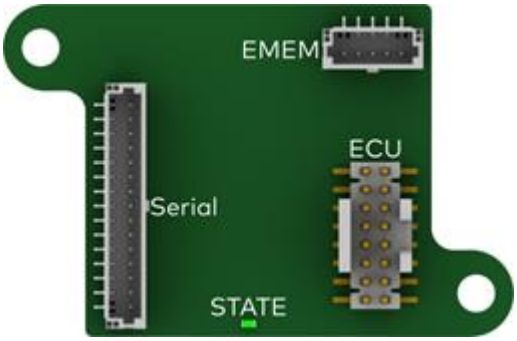


Figure 205: VX1543B top view connectors

- **Serial**
  The Serial connector supplies power to the POD and allows communication between the VX1543B and the VX1000 Base Module. The connector is compatible with cables of the type:
  - VX134xB Serial Pigtail (see page 657)
  - VX1311G / VX1311H Serial2 Pigtail (see page 659)
- **EMEM**
  When an Emulation RAM power supply and Calibration Wakeup functionality is needed, a VX1944 EMEM power supply can be attached via this connector.
- **ECU**
  The VX1543B can access a microcontroller of a customer ECU or a Development/Evaluation board via the target ECU interface connector.

Port	Connector	Target Interface type
ECU	16-pin Samtec	DAP (2-pin mode, 3-pin wide mode) JTAG SWD

5.10.3 LEDs

The state of the Serial POD is shown via a green/red LED blink sequence.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off R = red G = green

Sequence	Description
□□□□GGGG	ECU interface disabled.
□G□G□G□G	POD in configuration mode.
GGGGGGGG	ECU interface configured.

Note
The LED behavior has changed since VXtools 4.4. Please see VX1000 Manual V4.3 or earlier for the former LED sequence description.

5.10.4 Technical Data

Dimensions (L/W/H)	35.4 x 23 x 8 mm (without enclosure) 44 x 32.5 x 18 mm (with enclosure)
Weight	3 g (without enclosure)
Power supply	via VX1000 Base Module
Power consumption	0.8 W at max. operating temperature
Operating temperature range	-40 °C ... +115 °C (max. temperature at FPGA case)
Part number	120301

Note
Technical drawings can be found at vector.com/vx1000-drawings.

5.10.5 Target Interface Connector

5.10.5.1 Infineon DAP

ECU (DAP-IFX)

The following table shows the pin assignment of the ECU connector of the VX1543B configured for DAP-IFX. The functional details of the interfaces are described in the section Hardware Integration Guide The pin direction is shown as seen from the POD. For microcontrollers with DAP, the following interface groups are provided:

- DAP is mandatory for ECU access and provides the DAQ functionality
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	IN	-	Strongly recommended*
2	DAP1	DAP	m	IN/OUT	1V8/3V3/5V	
3	GND	-	m	PWR	-	
4	DAP0	DAP	m	OUT	1V8/3V3/5V	
5	GND	-	m	PWR	-	
6	DAP2	DAP	(m)	IN/OUT	1V8/3V3/5V	Mandatory if using DAP Wide Mode
7	GND	-	m	PWR	-	
8	DAPEN	DAP	m	OUT	1V8/3V3/5V	4.75 kΩ pull-up to VREF on POD side
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	1V8/3V3/5V	
11	reserved	-	-	DNC	-	
12	WDGDIS	SYS	o	OUT	1V8/3V3/5V	Allows disabling an external watchdog for flash programming*
13	reserved	-	-	DNC	-	
14	reserved	-	-	DNC	-	
15	VREF	SYS	o	IN	-	Connected to Pin 1 on POD side
16	nRSTOUT	SYS	o	IN	1V8/3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

Note
FTSH connectors are non-locking and therefore must be fixed if a vibration-proof connection is required. If the VX1543B is ordered with a sealed housing, all connectors on the VX1543B are fixed.

Target cable

For the Infineon DAP configuration a target cable is required which connects the appropriate 10 pins of the 16-pin connector on the VX1543B Serial POD to the 10-pin connector on the target side.

The Infineon DAP target cable is delivered together with the VX1543B Serial POD if it was ordered with Infineon DAP option.

Target (DAP-IFX)

On the target microcontroller side a Samtec FTSH-105-01-L-DV-K-P is required. The pin assignment for Infineon DAP is shown in the following table. The pin direction is shown as seen from the target.

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	OUT	-	Strongly recommended*
2	DAP1	DAP	m	IN/OUT	1V8/3V3/5V	
3	GND	-	m	PWR	-	
4	DAP0	DAP	m	IN	1V8/3V3/5V	
5	GND	-	m	PWR	-	
6	DAP2	DAP	(m)	IN/OUT	1V8/3V3/5V	Mandatory if using DAP Wide Mode
7	GND	-	m	PWR	-	
8	DAPEN	DAP	m	IN	1V8/3V3/5V	4.75 kΩ pull-up to VREF on POD side
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	1V8/3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Pin direction	Description
IN	Input of the ECU
OUT	Output of the ECU
IN/OUT	Input/Output of the ECU
PWR	Power supply
DNC	Do not connect

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The VX1543B contains a pull-up resistor of 4.7 kΩ on the DAPEN (nTRST) pin to VREF. During power-up the ECU microcontroller needs a valid logic high level on the nTRST pin to enable the DAP Interface. This must be kept in mind when designing a pull-down resistor on nTRST of the target ECU.

5.10.5.2 JTAG

ECU (JTAG)

The following table shows the pin assignment of the ECU connector of the VX1543B configured for JTAG mode. The functional details of the interfaces are described in the section Hardware Integration Guide The pin direction is shown as seen from the POD. For microcontrollers with JTAG, the following interface groups are provided:

- JTAG is mandatory for ECU access and provides the DAQ functionality
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	IN	-	Strongly recommended*
2	TMS	JTAG	m	OUT	1V8/3V3/5V	
3	GND	-	m	PWR	-	
4	TCK	JTAG	m	OUT	1V8/3V3/5V	
5	GND	-	m	PWR	-	
6	TDO	JTAG	m	IN	1V8/3V3/5V	
7	GND	-	m	PWR	-	
8	TDI	JTAG	m	OUT	1V8/3V3/5V	
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	1V8/3V3/5V	
11	reserved	-	-	DNC	-	
12	WDGDIS	SYS	o	OUT	1V8/3V3/5V	Allows disabling an external watchdog for flash programming*
13	nRDY	JTAG	o	IN	1V8/3V3/5V	Recommended for optimized performance*
14	nTRST/JCOMP	JTAG	(m)	OUT	1V8/3V3/5V	Mandatory if available on target microcontroller
15	VREF	SYS	o	IN	-	Connected to Pin 1 on POD side
16	nRSTOUT	SYS	o	IN	1V8/3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD

Pin direction	Description
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

Note
FTSH connectors are non-locking and therefore must be fixed if a vibration-proof connection is required. If the VX1543B is ordered with a sealed housing, all connectors on the VX1543B are fixed.

Target cable

For the JTAG configuration a target cable is required which connects the appropriate 16 pins of the 16-pin connector on the VX1543B to the 16-pin connector on the target side.

The JTAG target cable is delivered together with the VX1543B if it was ordered with JTAG option.

Target (JTAG)

On the target microcontroller side a Samtec FTSH-108-01-L-DV-K-P is required. The pin assignment for JTAG is shown in the following table. The pin direction is shown as seen from the target.

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	OUT	-	Strongly recommended*
2	TMS	JTAG	m	IN	1V8/3V3/5V	
3	GND	-	m	PWR	-	
4	TCK	JTAG	m	IN	1V8/3V3/5V	
5	GND	-	m	PWR	-	
6	TDO	JTAG	m	OUT	1V8/3V3/5V	
7	GND	-	m	PWR	-	
8	TDI	JTAG	m	IN	1V8/3V3/5V	
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	1V8/3V3/5V	
11	reserved	-	-	DNC	-	
12	WDGDIS	SYS	o	IN	1V8/3V3/5V	Allows disabling an external watchdog for flash programming*
13	nRDY	JTAG	o	OUT	1V8/3V3/5V	Recommended for optimized performance*
14	nTRST/JCOMP	JTAG	(m)	IN	1V8/3V3/5V	Mandatory if available on target microcontroller
15	VREF	SYS	o	OUT	-	Connected to Pin 1 on POD side
16	nRSTOUT	SYS	o	OUT	1V8/3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Pin direction	Description
IN	Input of the ECU
OUT	Output of the ECU
IN/OUT	Input/Output of the ECU
PWR	Power supply
DNC	Do not connect

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

5.10.5.3 SWD

ECU (SWD)

The following table shows the pin assignment of the ECU connector of the VX1543B configured for SWD. The functional details of the interfaces are described in the section Hardware Integration Guide The pin direction is shown as seen from the POD. For microcontrollers with SWD, the following interface groups are provided:

- SWD is mandatory for ECU access and provides the DAQ functionality
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	IN	-	Strongly recommended*
2	SWDIO	SWD	m	IN/OUT	1V8/3V3/5V	
3	GND	-	m	PWR	-	
4	SWCLK	SWD	m	OUT	1V8/3V3/5V	
5	GND	-	m	PWR	-	
6	reserved	-	-	DNC	-	
7	GND	-	m	PWR	-	
8	reserved	-	-	DNC	-	
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	1V8/3V3/5V	
11	reserved	-	-	DNC	-	
12	WDGDIS	SYS	o	OUT	1V8/3V3/5V	Allows disabling an external watchdog for flash programming*
13	reserved	-	-	DNC	-	
14	reserved	-	-	DNC	-	
15	VREF	SYS	o	IN	-	Connected to Pin 1 on POD side
16	nRSTOUT	SYS	o	IN	1V8/3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

Note
FTSH connectors are non-locking and therefore must be fixed if a vibration-proof connection is required. If the VX1543B is ordered with a sealed housing, all connectors on the VX1543B are fixed.

Target cable

For SWD a target cable is required which connects the appropriate 10 pins of the 16-pin connector on the VX1543B Serial POD to the 10-pin connector on the target side.

The SWD target cable is delivered together with the VX1543B Serial POD if it was ordered with SWD option.

Target (SWD)

On the target microcontroller side a Samtec FTSH-105-01-L-DV-K-P is required. The pin assignment for SWD is shown in the following table. The pin direction is shown as seen from the target.

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	OUT	-	Strongly recommended*
2	SWDIO	SWD	m	IN/OUT	1V8/3V3/5V	
3	GND	-	m	PWR	-	
4	SWCLK	SWD	m	IN	1V8/3V3/5V	
5	GND	-	m	PWR	-	
6	reserved	-	-	DNC	-	
7	GND	-	m	PWR	-	
8	reserved	-	-	DNC	-	
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	1V8/3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Pin direction	Description
IN	Input of the ECU

Pin direction	Description
OUT	Output of the ECU
IN/OUT	Input/Output of the ECU
PWR	Power supply
DNC	Do not connect

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

5.11 VX1543C Serial POD

5.11.1 General Information

The VX1543C Serial POD is the target interface adaptation component of the VX1000 system and has the following features:

- Support of the following microcontroller interfaces:
  - Infineon DAP up to 80 MHz
  - NXP/ST PowerPC JTAG (Nexus Class 2+)
  - Renesas RH850 JTAG (Nexus Class 2+)
- Supports target interface voltages of 3.3 V and 5 V
- Very small form factor, very low power dissipation and automotive-graded
- Intended use: Customer ECU integration and laboratory environment

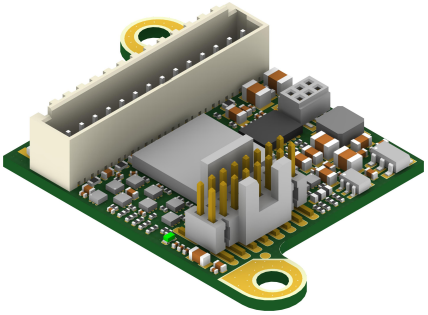


Figure 206: VX1543C Serial POD

WARNING!
VX1000 Hardware is not designed for high voltage (HV) environments, unless explicitly stated otherwise! If a solution for a HV environment is required, please contact VXsupport@vector.com for guidance.

WARNING!
Always ensure that the POD target interface is correctly configured or disabled before connecting a POD to the target. Wrong settings can damage both systems.

WARNING!
The Serial POD must be handled with care when physically attaching to or detaching from a target! Neither the POD nor the target cables nor the VX1927 Flex-PCBs are designed for frequent mating cycles!

Note
A detailed list of supported microcontrollers can be found in Supported Microcontrollers on page 19.

5.11.2 Connectors

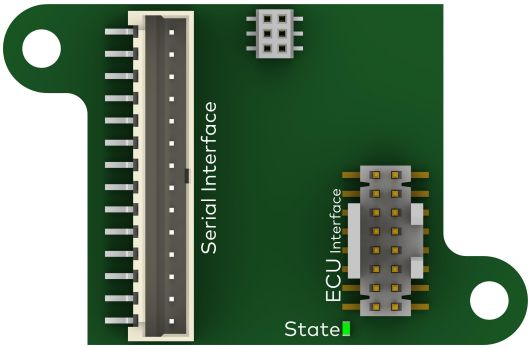


Figure 207: VX1543C top view connectors

- **Serial**
  The Serial connector supplies power to the POD and allows communication between the VX1543C and the VX1000 Base Module. The connector is compatible with cables of the type:
  - VX134xB Serial Pigtail (see page 657)
  - VX1311G / VX1311H Serial2 Pigtail (see page 659)
- **ECU**
  The VX1543C can access a microcontroller of a customer ECU or a Development/Evaluation board via the target ECU interface connector.

Port	Connector	Target Interface type
ECU	16-pin Samtec	DAP (2-pin mode) JTAG (Nexus Class 2+) SWD

5.11.3 LEDs

The state of the Serial POD is shown via a green/red LED blink sequence.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off R = red G = green

Sequence	Description
□□□□GGGG	ECU interface disabled.
□G□G□G□G	POD in configuration mode.
GGGGGGGG	ECU interface configured.

Note
The LED behavior has changed since VXtools 4.4. Please see VX1000 Manual V4.3 or earlier for the former LED sequence description.

5.11.4 Technical Data

Dimensions (L/W/H)	35.4 x 23 x 7 mm
Weight	3 g
Power supply	via VX1000 Base Module
Power consumption	0.8 W at max. operating temperature
Operating temperature range	-40 °C ... +115 °C (max. temperature at FPGA case)
Part number	22395

Note
Technical drawings can be found at vector.com/vx1000-drawings.

5.11.5 Target Interface Connector

5.11.5.1 Infineon DAP

ECU (DAP-IFX)

The following table shows the pin assignment of the ECU connector of the VX1543C configured for DAP-IFX. The functional details of the interfaces are described in the section Hardware Integration Guide The pin direction is shown as seen from the POD. For microcontrollers with DAP, the following interface groups are provided:

- DAP is mandatory for ECU access and provides the DAQ functionality
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	IN	-	Strongly recommended*
2	DAP1	DAP	m	IN/OUT	3V3/5V	
3	GND	-	m	PWR	-	
4	DAP0	DAP	m	OUT	3V3/5V	
5	GND	-	m	PWR	-	
6	reserved	-	-	DNC	-	
7	GND	-	m	PWR	-	
8	DAPEN	DAP	m	OUT	3V3/5V	4.75 kΩ pull-up to VREF on POD side
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	3V3/5V	
11	reserved	-	-	DNC	-	
12	WDGDIS	SYS	o	OUT	3V3/5V	Allows disabling an external watchdog for flash programming*
13	reserved	-	-	DNC	-	
14	reserved	-	-	DNC	-	
15	VREF	SYS	o	IN	-	Connected to pin 1 on POD side
16	nRSTOUT	SYS	o	IN	3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD

Pin direction	Description
IN/OUT	Input/Output of the POD
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

Note
FTSH connectors are non-locking and therefore must be fixed if a vibration-proof connection is required. If the VX1543C is ordered with a sealed housing, all connectors on the VX1543C are fixed.

Target cable

For the Infineon DAP configuration a target cable is required which connects the appropriate 10 pins of the 16-pin connector on the VX1543C Serial POD to the 10-pin connector on the target side.

The Infineon DAP target cable is delivered together with the VX1543C Serial POD if it was ordered with Infineon DAP option.

Target (DAP-IFX)

On the target microcontroller side a Samtec FTSH-105-01-L-DV-K-P is required. The pin assignment for Infineon DAP is shown in the following table. The pin direction is shown as seen from the target.

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	OUT	-	Strongly recommended*
2	DAP1	DAP	m	IN/OUT	3V3/5V	
3	GND	-	m	PWR	-	
4	DAP0	DAP	m	IN	3V3/5V	
5	GND	-	m	PWR	-	
6	reserved	-	-	DNC	-	
7	GND	-	m	PWR	-	
8	DAPEN	DAP	m	IN	3V3/5V	4.75 kΩ pull-up to VREF on POD side
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Pin direction	Description
IN	Input of the ECU
OUT	Output of the ECU
IN/OUT	Input/Output of the ECU
PWR	Power supply
DNC	Do not connect

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The VX1543C contains a pull-up resistor of 4.7 kΩ on the DAPEN (nTRST) pin to VREF. During power-up the ECU microcontroller needs a valid logic high level on the nTRST pin to enable the DAP Interface. This must be kept in mind when designing a pull-down resistor on nTRST of the target ECU.

5.11.5.2 JTAG

ECU (JTAG)

The following table shows the pin assignment of the ECU connector of the VX1543C configured for JTAG mode. The functional details of the interfaces are described in the section Hardware Integration Guide The pin direction is shown as seen from the POD. For microcontrollers with JTAG, the following interface groups are provided:

- JTAG is mandatory for ECU access and provides the DAQ functionality
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	IN	-	Strongly recommended*
2	TMS	JTAG	m	OUT	3V3/5V	
3	GND	-	m	PWR	-	
4	TCK	JTAG	m	OUT	3V3/5V	
5	GND	-	m	PWR	-	
6	TDO	JTAG	m	IN	3V3/5V	
7	GND	-	m	PWR	-	
8	TDI	JTAG	m	OUT	3V3/5V	
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	3V3/5V	
11	reserved	-	-	DNC	-	
12	WDGDIS	SYS	o	OUT	3V3/5V	Allows disabling an external watchdog for flash programming*
13	nRDY	JTAG	o	IN	3V3/5V	Recommended for optimized performance*
14	nTRST/JCOMP	JTAG	(m)	OUT	3V3/5V	Mandatory if available on target microcontroller
15	VREF	SYS	o	IN	-	Connected to pin 1 on POD side
16	nRSTOUT	SYS	o	IN	3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD

Pin direction	Description
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

Note
FTSH connectors are non-locking and therefore must be fixed if a vibration-proof connection is required. If the VX1543C is ordered with a sealed housing, all connectors on the VX1543C are fixed.

Target cable

For the JTAG configuration a target cable is required which connects the appropriate 16 pins of the 16-pin connector on the VX1543C to the 16-pin connector on the target side.

The JTAG target cable is delivered together with the VX1543C if it was ordered with JTAG option.

Target (JTAG)

On the target microcontroller side a Samtec FTSH-108-01-L-DV-K-P is required. The pin assignment for JTAG is shown in the following table. The pin direction is shown as seen from the target.

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	OUT	-	Strongly recommended*
2	TMS	JTAG	m	IN	3V3/5V	
3	GND	-	m	PWR	-	
4	TCK	JTAG	m	IN	3V3/5V	
5	GND	-	m	PWR	-	
6	TDO	JTAG	m	OUT	3V3/5V	
7	GND	-	m	PWR	-	
8	TDI	JTAG	m	IN	3V3/5V	
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	3V3/5V	
11	reserved	-	-	DNC	-	
12	WDGDIS	SYS	o	IN	3V3/5V	Allows disabling an external watch-dog for flash programming*
13	nRDY	JTAG	o	OUT	3V3/5V	Recommended for optimized per-formance*
14	nTRST/JCOMP	JTAG	(m)	IN	3V3/5V	Mandatory if available on target microcontroller
15	VREF	SYS	o	OUT	-	Connected to pin 1 on POD side
16	nRSTOUT	SYS	o	OUT	3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Pin direction	Description
IN	Input of the ECU
OUT	Output of the ECU
IN/OUT	Input/Output of the ECU
PWR	Power supply
DNC	Do not connect

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

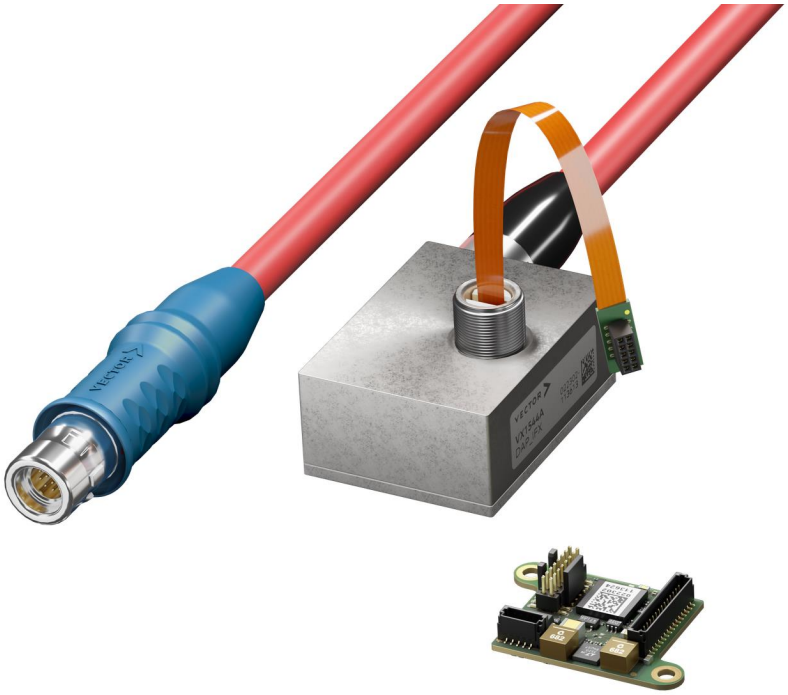
I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

5.12 VX1544A Serial POD

5.12.1 General Information

The VX1544A Serial POD is the target interface adaptation component of the VX1000 system and has the following features:

- Support of the following microcontrollers interfaces:
  - Infineon DAP up to 120 MHz
  - NXP/ST PowerPC JTAG (Nexus Class 2+)
  - Renesas RH850 JTAG (Nexus Class 2+)
  - Infineon HSCT
  - PowerPC Zipwire
  - Renesas RHSIF
  - ARM SWD
  - ARM JTAG
- Supports target interface voltages of 1.8 V, 3.3 V and 5 V
- Very small form factor, very low power dissipation and automotive-graded
- Intended use: Customer ECU integration and laboratory environment



VX1544A Serial POD

WARNING!
VX1000 Hardware is not designed for high voltage (HV) environments, unless explicitly stated otherwise! If a solution for a HV environment is required, please contact VXsupport@vector.com for guidance.

WARNING!
The Serial POD must be handled with care when physically attaching to or detaching from a target! Neither the POD nor the target cables nor the VX1927 Flex-PCBs are designed for frequent mating cycles!

WARNING!
Opening the enclosure lid may compromise the seal, even if the lid is reattached.

Note
A detailed list of supported microcontrollers can be found in Supported Micro-controllers on page 19.

Plastic carrier

When ordered without housing, the VX1544A Serial POD is delivered with a plastic carrier to allow mounting the double sided populated POD PCB onto electrically conductive surfaces.

5.12.2 Connectors

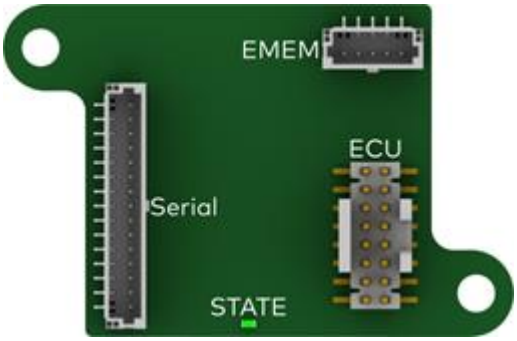


Figure 208: VX1544A top view connectors

- **Serial**
  The Serial connector supplies power to the POD and allows communication between the VX1544A Serial POD and the VX1000 Base Module. The connector is compatible with cables of the type:
  - VX134xB Serial Pigtail (see page 657)
  - VX1311G / VX1311H Serial2 Pigtail (see page 659)
- **EMEM**
  When an Emulation RAM power supply and Calibration Wakeup functionality is needed, a VX1944 EMEM power supply can be attached via this connector.
- **ECU**
  The VX1544A Serial POD can access a microcontroller of a customer ECU or a Development/Evaluation board via the target ECU interface connector.

Port	Connector	Target Interface type
ECU	16-pin Samtec	DAP (2-pin mode, 3-pin wide mode) JTAG Infineon HSCT / PowerPC Zipwire SWD

5.12.3 LEDs

The state of the Serial POD is shown via a green/red LED blink sequence.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off R = red G = green

Sequence	Description
□□□□GGGG	ECU interface disabled.
□G□G□G□G	POD in configuration mode.
GGGGGGGG	ECU interface configured.

Note
The LED behavior has changed since VXtools 4.4. Please see VX1000 Manual V4.3 or earlier for the former LED sequence description.

5.12.4 Technical Data

Dimensions (L/W/H)	35.4 x 23 x 8 mm (without enclosure) 44 x 32.5 x 18 mm (with enclosure)
PCB Weight	3 g (without enclosure)
Power supply	via VX1000 Base Module
Power consumption	0.8 W at max. operating temperature
Operating temperature range	-40 °C ... +115 °C (max. temperature at FPGA case)
Part number	22302

Note
Technical drawings can be found at vector.com/vx1000-drawings.

5.12.5 Target Interface Connector

5.12.5.1 Infineon DAP

ECU (DAP-IFX)

The following table shows the pin assignment of the ECU connector of the VX1544A configured for DAP-IFX. The functional details of the interfaces are described in the section Hardware Integration Guide The pin direction is shown as seen from the POD. For microcontrollers with DAP, the following interface groups are provided:

- DAP is mandatory for ECU access and provides the DAQ functionality
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	IN	-	Strongly recommended*
2	DAP1	DAP	m	IN/OUT	1V8/3V3/5V	
3	GND	-	m	PWR	-	
4	DAP0	DAP	m	OUT	1V8/3V3/5V	
5	GND	-	m	PWR	-	
6	DAP2	DAP	(m)	IN/OUT	1V8/3V3/5V	Mandatory if using DAP Wide Mode
7	GND	-	m	PWR	-	
8	DAPEN	DAP	m	OUT	1V8/3V3/5V	4.75 kΩ pull-up to VREF on POD side
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	1V8/3V3/5V	
11	reserved	-	-	DNC	-	
12	WDGDIS	SYS	o	OUT	1V8/3V3/5V	Allows disabling an external watchdog for flash programming*
13	reserved	-	-	DNC	-	
14	reserved	-	-	DNC	-	
15	VREF	SYS	o	IN	-	Connected to pin 1 on POD side
16	nRSTOUT	SYS	o	IN	1V8/3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD

Pin direction	Description
OUT	Output of the POD
IN/OUT	Input/Output of the POD
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

WARNING!
The VX1544A contains a pull-up resistor of 4.7 kΩ on the DAPEN (nTRST) pin to VREF. During power-up the ECU microcontroller needs a valid logic high level on the nTRST pin to enable the DAP Interface. This must be kept in mind when designing a pull-down resistor on nTRST of the target ECU.

Note
FTSH connectors are non-locking and therefore must be fixed if a vibration-proof connection is required. If the VX1544A is ordered with a sealed housing, all connectors on the VX1544A are fixed.

FlexPCB

For the Infineon DAP configuration a specific adapter is required which connects the appropriate 10 pins of the 16-pin connector on the VX1544A to the 10-pin connector on the target side.

The VX1927C DAP-IFX FlexPCB for VX154x (see page 602) is optionally available.

5.12.5.2 JTAG

ECU (JTAG)

The following table shows the pin assignment of the ECU connector of the VX1544A configured for JTAG mode. The functional details of the interfaces are described in the section Hardware Integration Guide The pin direction is shown as seen from the POD. For microcontrollers with JTAG, the following interface groups are provided:

- JTAG is mandatory for ECU access and provides the DAQ functionality
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	IN	-	Strongly recommended*
2	TMS	JTAG	m	OUT	1V8/3V3/5V	
3	GND	-	m	PWR	-	
4	TCK	JTAG	m	OUT	1V8/3V3/5V	
5	GND	-	m	PWR	-	
6	TDO	JTAG	m	IN	1V8/3V3/5V	
7	GND	-	m	PWR	-	
8	TDI	JTAG	m	OUT	1V8/3V3/5V	
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	1V8/3V3/5V	
11	reserved	-	-	DNC	-	
12	WDGDIS	SYS	o	OUT	1V8/3V3/5V	Allows disabling an external watchdog for flash programming*
13	nRDY	JTAG	o	IN	1V8/3V3/5V	Recommended for optimized performance*
14	nTRST/JCOMP	JTAG	(m)	OUT	1V8/3V3/5V	Mandatory if available on target microcontroller
15	VREF	SYS	o	IN	-	Connected to pin 1 on POD side
16	nRSTOUT	SYS	o	IN	1V8/3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD

Pin direction	Description
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

Note
FTSH connectors are non-locking and therefore must be fixed if a vibration-proof connection is required. If the VX1544A is ordered with a sealed housing, all connectors on the VX1544A are fixed.

FlexPCB

For the JTAG configuration a specific adapter is required which connects the 16-pin connector on the VX1544A to the 16-pin connector on the target side.

The VX1927A JTAG-PT FlexPCB for VX154x (see page 596) is optional available. Depending on the target microcontroller type, an additional adapter PCB may be required. For more information please contact VXsupport@vector.com.

5.12.5.3 Infineon HSCT, Renesas RHSIF, PowerPC Zipwire

ECU

The following table shows the pin assignment of the ECU connector of the VX1544A configured for JTAG mode. The functional details of the interfaces are described in the section Hardware Integration Guide The pin direction is shown as seen from the POD. For microcontrollers with HSCT/RHSIF/Zipwire, the following interface groups are provided:

- HSCT (HSCT/RHSIF/Zipwire) is mandatory for ECU access and provides the DAQ functionality
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	IN	-	Strongly recommended*
2	TXDp	HSCT	m	IN	LVDS	Pairwise required with pin 8 TXDn
3	GND	-	m	PWR	-	
4	REFCLK	HSCT	m	IN	1V8/3V3/5V	
5	GND	-	m	PWR	-	
6	RXDp	HSCT	m	OUT	LVDS	Pairwise required with pin 14 RXDn
7	GND	-	m	PWR	-	
8	TXDn	HSCT	m	IN	LVDS	Pairwise required with pin 2 TXDp
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	1V8/3V3/5V	
11	reserved	-	-	DNC	-	
12	WDGDIS	SYS	o	OUT	1V8/3V3/5V	Allows disabling an external watchdog for flash programming*
13	reserved	-	-	DNC	-	
14	RXDn	HSCT	m	OUT	LVDS	Pairwise required with pin 6 RXDp
15	VREF	SYS	o	IN	-	Connected to pin 1 on POD side
16	nRSTOUT	SYS	o	IN	1V8/3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD

Pin direction	Description
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

Note
FTSH connectors are non-locking and therefore must be fixed if a vibration-proof connection is required. If the VX1544A is ordered with a sealed housing, all connectors on the VX1544A are fixed.

FlexPCB

For the HSCT/Zipwire/RHSIF configuration a specific adapter is required which connects the 16-pin connector on the VX1544A to the 10-pin connector on the target side.

The VX1927D Zipwire/HSCT FlexPCB for VX154x (see page 605) is optional available.

5.12.5.4 SWD

ECU (SWD)

The following table shows the pin assignment of the ECU connector of the VX1543B configured for SWD. The functional details of the interfaces are described in the section Hardware Integration Guide The pin direction is shown as seen from the POD. For microcontrollers with SWD, the following interface groups are provided:

- SWD is mandatory for ECU access and provides the DAQ functionality
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	IN	-	Strongly recommended*
2	SWDIO	SWD	m	IN/OUT	1V8/3V3/5V	
3	GND	-	m	PWR	-	
4	SWCLK	SWD	m	OUT	1V8/3V3/5V	
5	GND	-	m	PWR	-	
6	reserved	-	-	DNC	-	
7	GND	-	m	PWR	-	
8	reserved	-	-	DNC	-	
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	1V8/3V3/5V	
11	reserved	-	-	DNC	-	
12	WDGDIS	SYS	o	OUT	1V8/3V3/5V	Allows disabling an external

Pin	Signal	Group	Need	Dir	I/O	Details
						watchdog for flash programming*
13	reserved	-	-	DNC	-	
14	reserved	-	-	DNC	-	
15	VREF	SYS	o	IN	-	Connected to Pin 1 on POD side
16	nRSTOUT	SYS	o	IN	1V8/3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

Note
FTSH connectors are non-locking and therefore must be fixed if a vibration-proof connection is required. If the VX1543B is ordered with a sealed housing, all connectors on the VX1543B are fixed.

Target cable

For SWD a target cable is required which connects the appropriate 10 pins of the 16-pin connector on the VX1544A Serial POD to the 10-pin connector on the target side.

The SWD target cable is delivered together with the VX1544A Serial POD if it was ordered with SWD option.

Target (SWD)

On the target microcontroller side a Samtec FTSH-105-01-L-DV-K-P is required. The pin assignment for SWD is shown in the following table. The pin direction is shown as seen from the target.

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	OUT	-	Strongly recommended*
2	SWDIO	SWD	m	IN/OUT	1V8/3V3/5V	
3	GND	-	m	PWR	-	
4	SWCLK	SWD	m	IN	1V8/3V3/5V	
5	GND	-	m	PWR	-	
6	reserved	-	-	DNC	-	
7	GND	-	m	PWR	-	
8	reserved	-	-	DNC	-	
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	1V8/3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Pin direction	Description
IN	Input of the ECU
OUT	Output of the ECU
IN/OUT	Input/Output of the ECU
PWR	Power supply
DNC	Do not connect

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

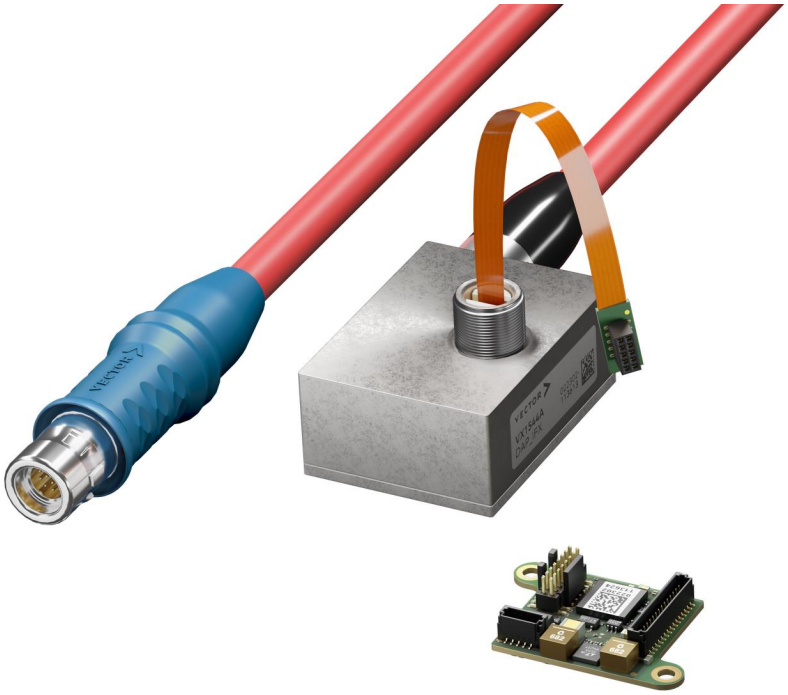
I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

5.13 VX1544B Serial POD

5.13.1 General Information

The VX1544B Serial POD is the target interface adaptation component of the VX1000 system and has the following features:

- Support of the following microcontrollers interfaces:
  - Infineon DAP up to 160 MHz
- Supports target interface voltage of 3.3 V
- Very small form factor, very low power dissipation and automotive-graded
- Intended use: Customer ECU integration and laboratory environment



VX1544B Serial POD

WARNING!
VX1000 Hardware is not designed for high voltage (HV) environments, unless explicitly stated otherwise! If a solution for a HV environment is required, please contact VXsupport@vector.com for guidance.

WARNING!
The Serial POD must be handled with care when physically attaching to or detaching from a target! Neither the POD nor the target cables nor the VX1927 Flex-PCBs are designed for frequent mating cycles!

WARNING!
Opening the enclosure lid may compromise the seal, even if the lid is reattached.

Note
A detailed list of supported microcontrollers can be found in Supported Micro-controllers on page 19.

Plastic carrier

When ordered without housing, the VX1544B Serial POD is delivered with a plastic carrier to allow mounting the double sided populated POD PCB onto electrically conductive surfaces.

5.13.2 Connectors

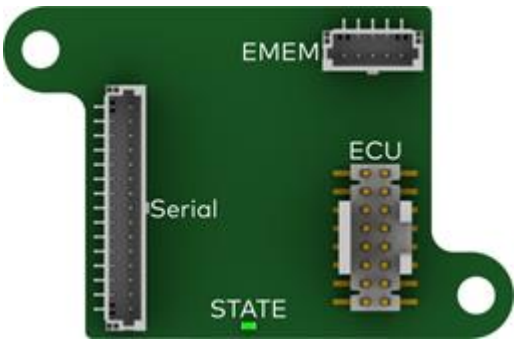


Figure 209: VX1544B top view connectors

- **Serial**
  The Serial connector supplies power to the POD and allows communication between the VX1544B Serial POD and the VX1000 Base Module. The connector is compatible with cables of the type:
  - VX134xB Serial Pigtail (see page 657)
  - VX1311G / VX1311H Serial2 Pigtail (see page 659)
- **EMEM**
  When an Emulation RAM power supply and Calibration Wakeup functionality is needed, a VX1944 EMEM power supply can be attached via this connector.
- **ECU**
  The VX1544B Serial POD can access a microcontroller of a customer ECU or a Development/Evaluation board via the target ECU interface connector.

Port	Connector	Target Interface type
ECU	16-pin Samtec	DAP (2-pin mode, 3-pin wide mode)

5.13.3 LEDs

The state of the Serial POD is shown via a green/red LED blink sequence.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off R = red G = green

Sequence	Description
□□□□GGGG	ECU interface disabled.
□G□G□G□G	POD in configuration mode.
GGGGGGGG	ECU interface configured.

Note
The LED behavior has changed since VXtools 4.4. Please see VX1000 Manual V4.3 or earlier for the former LED sequence description.

5.13.4 Technical Data

Dimensions (L/W/H)	35.4 x 23 x 8 mm (without enclosure) 44 x 32.5 x 18 mm (with enclosure)
PCB Weight	3 g (without enclosure)
Power supply	via VX1000 Base Module
Power consumption	0.8 W at max. operating temperature
Operating temperature range	-40 °C ... +115 °C (max. temperature at FPGA case)
Part number	22276

Note
Technical drawings can be found at vector.com/vx1000-drawings.

5.13.5 Target Interface Connector

5.13.5.1 Infineon DAP

ECU (DAP-IFX)

The following table shows the pin assignment of the ECU connector of the VX1544B configured for DAP-IFX. The functional details of the interfaces are described in the section Hardware Integration Guide The pin direction is shown as seen from the POD. For microcontrollers with DAP, the following interface groups are provided:

- DAP is mandatory for ECU access and provides the DAQ functionality
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	IN	-	Strongly recommended*
2	DAP1	DAP	m	IN/OUT	3V3	
3	GND	-	m	PWR	-	
4	DAP0	DAP	m	OUT	3V3	
5	GND	-	m	PWR	-	
6	DAP2	DAP	(m)	IN/OUT	3V3	Mandatory if using DAP Wide Mode
7	GND	-	m	PWR	-	
8	DAPEN	DAP	m	OUT	3V3	4.75 kΩ pull-up to VREF on POD side
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	3V3	
11	reserved	-	-	DNC	-	
12	WDGDIS	SYS	o	OUT	3V3	Allows disabling an external watchdog for flash programming*
13	reserved	-	-	DNC	-	
14	reserved	-	-	DNC	-	
15	VREF	SYS	o	IN	-	Connected to pin 1 on POD side
16	nRSTOUT	SYS	o	IN	3V3	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD

Pin direction	Description
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

WARNING!
The VX1544B contains a pull-up resistor of 4.7 kΩ on the DAPEN (nTRST) pin to VREF. During power-up the ECU microcontroller needs a valid logic high level on the nTRST pin to enable the DAP Interface. This must be kept in mind when designing a pull-down resistor on nTRST of the target ECU.

Note
FTSH connectors are non-locking and therefore must be fixed if a vibration-proof connection is required. If the VX1544B is ordered with a sealed housing, all connectors on the VX1544B are fixed.

FlexPCB

For the Infineon DAP configuration a specific adapter is required which connects the appropriate 10 pins of the 16-pin connector on the VX1544B to the 10-pin connector on the target side.

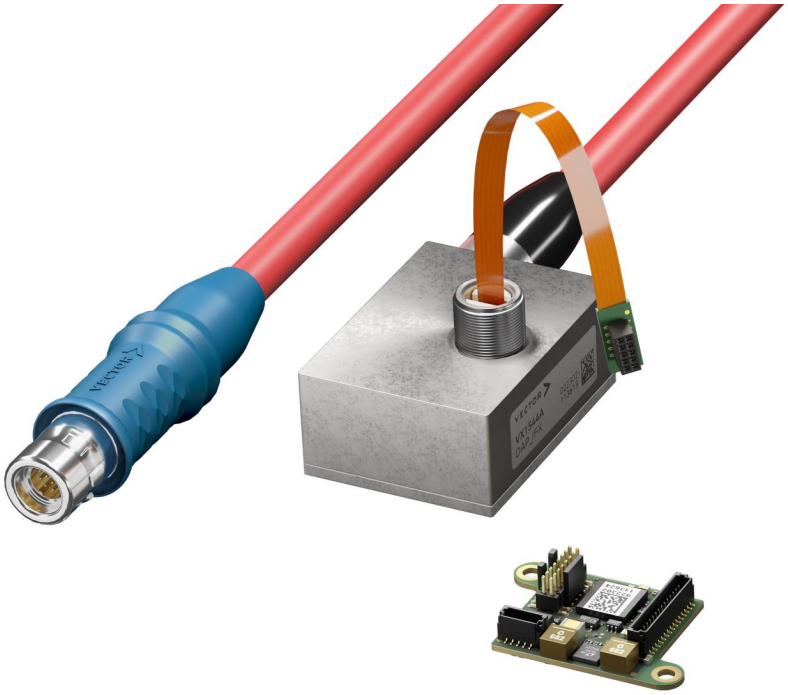
The VX1927C DAP-IFX FlexPCB for VX154x (see page 602) is optional available. This adapter features DAP frequencies up to 160MHz.

5.14 VX1544C Serial POD

5.14.1 General Information

The VX1544C Serial POD is the target interface adaptation component of the VX1000 system and has the following features:

- Support of the following microcontrollers interfaces:
  - Infineon DAP up to 160 MHz
- Supports target interface voltage of 5 V
- Very small form factor, very low power dissipation and automotive-graded
- Intended use: Customer ECU integration and laboratory environment



VX1544C Serial POD

WARNING!
VX1000 Hardware is not designed for high voltage (HV) environments, unless explicitly stated otherwise! If a solution for a HV environment is required, please contact VXsupport@vector.com for guidance.

WARNING!
The Serial POD must be handled with care when physically attaching to or detaching from a target! Neither the POD nor the target cables nor the VX1927 Flex-PCBs are designed for frequent mating cycles!

WARNING!
Opening the enclosure lid may compromise the seal, even if the lid is reattached.

Note
A detailed list of supported microcontrollers can be found in Supported Micro-controllers on page 19.

Plastic carrier

When ordered without housing, the VX1544C Serial POD is delivered with a plastic carrier to allow mounting the double sided populated POD PCB onto electrically conductive surfaces.

5.14.2 Connectors

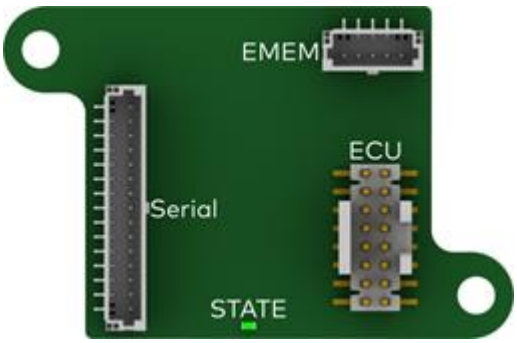


Figure 210: VX1544C top view connectors

- **Serial**
  The Serial connector supplies power to the POD and allows communication between the VX1544C Serial POD and the VX1000 Base Module. The connector is compatible with cables of the type:
  - VX134xB Serial Pigtail (see page 657)
  - VX1311G / VX1311H Serial2 Pigtail (see page 659)
- **EMEM**
  When an Emulation RAM power supply and Calibration Wakeup functionality is needed, a VX1944 EMEM power supply can be attached via this connector.
- **ECU**
  The VX1544C Serial POD can access a microcontroller of a customer ECU or a Development/Evaluation board via the target ECU interface connector.

Port	Connector	Target Interface type
ECU	16-pin Samtec	DAP (2-pin mode, 3-pin wide mode)

5.14.3 LEDs

The state of the Serial POD is shown via a green/red LED blink sequence.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off R = red G = green

Sequence	Description
□□□□GGGG	ECU interface disabled.
□G□G□G□G	POD in configuration mode.
GGGGGGGG	ECU interface configured.

Note
The LED behavior has changed since VXtools 4.4. Please see VX1000 Manual V4.3 or earlier for the former LED sequence description.

5.14.4 Technical Data

Dimensions (L/W/H)	35.4 x 23 x 8 mm (without enclosure) 44 x 32.5 x 18 mm (with enclosure)
PCB Weight	3 g (without enclosure)
Power supply	via VX1000 Base Module
Power consumption	0.8 W at max. operating temperature
Operating temperature range	-40 °C ... +115 °C (max. temperature at FPGA case)
Part number	22616

Note
Technical drawings can be found at vector.com/vx1000-drawings.

5.14.5 Target Interface Connector

5.14.5.1 Infineon DAP

ECU (DAP-IFX)

The following table shows the pin assignment of the ECU connector of the VX1544C configured for DAP-IFX. The functional details of the interfaces are described in the section Hardware Integration Guide The pin direction is shown as seen from the POD. For microcontrollers with DAP, the following interface groups are provided:

- DAP is mandatory for ECU access and provides the DAQ functionality
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	IN	-	Strongly recommended*
2	DAP1	DAP	m	IN/OUT	5V	
3	GND	-	m	PWR	-	
4	DAP0	DAP	m	OUT	5V	
5	GND	-	m	PWR	-	
6	DAP2	DAP	(m)	IN/OUT	5V	Mandatory if using DAP Wide Mode
7	GND	-	m	PWR	-	
8	DAPEN	DAP	m	OUT	5V	4.75 kΩ pull-up to VREF on POD side
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	5V	
11	reserved	-	-	DNC	-	
12	WDGDIS	SYS	o	OUT	5V	Allows disabling an external watchdog for flash programming*
13	reserved	-	-	DNC	-	
14	reserved	-	-	DNC	-	
15	VREF	SYS	o	IN	-	Connected to pin 1 on POD side
16	nRSTOUT	SYS	o	IN	5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD

Pin direction	Description
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The VX1544C contains a pull-up resistor of 4.7 kΩ on the DAPEN (nTRST) pin to VREF. During power-up the ECU microcontroller needs a valid logic high level on the nTRST pin to enable the DAP Interface. This must be kept in mind when designing a pull-down resistor on nTRST of the target ECU.

Note
FTSH connectors are non-locking and therefore must be fixed if a vibration-proof connection is required. If the VX1544C is ordered with a sealed housing, all connectors on the VX1544C are fixed.

FlexPCB

For the Infineon DAP configuration a specific adapter is required which connects the appropriate 10 pins of the 16-pin connector on the VX1544C to the 10-pin connector on the target side.

Please see section Adapters on page 582 for the available generic VX1927 DAP Flex PCBs.

5.15 VX1544D Serial POD

5.15.1 General Information

The VX1544D Serial POD is the target interface adaptation component of the VX1000 system.

It is intended to be directly plugged into a connector on the target PCB, and therefore has an ECU connector with a higher stacking height compared to the other VX154x Serial PODs.

The VX1544D has the following features:

- Support of the following microcontroller interfaces:
  - Infineon DAP up to 120 MHz
  - NXP/ST PowerPC JTAG (Nexus Class 2+)
  - Renesas RH850 JTAG (Nexus Class 2+)
  - Infineon HSCT
  - NXP/ST PowerPC Zipwire
  - Renesas RHSIF
  - ARM SWD
  - ARM JTAG
- Supports target interface voltages of 1.8 V, 3.3 V and 5 V
- Very small form factor, very low power dissipation and automotive-graded
- Intended use: Customer ECU integration and laboratory environment

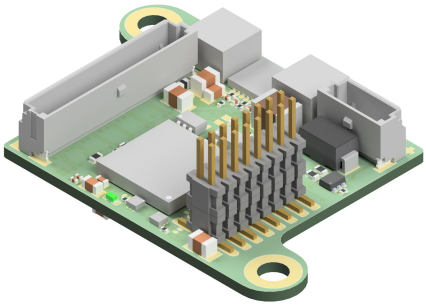


Figure 211: VX1544D Serial POD

WARNING!
VX1000 Hardware is not designed for high voltage (HV) environments, unless explicitly stated otherwise! If a solution for a HV environment is required, please contact VXsupport@vector.com for guidance.

WARNING!
The Serial POD must be handled with care when physically attaching to or detaching from a target! Neither the POD nor the target cables nor the VX1927 Flex-PCBs are designed for frequent mating cycles!

Note
A detailed list of supported microcontrollers can be found in Supported Microcontrollers on page 19.

5.15.2 Connectors

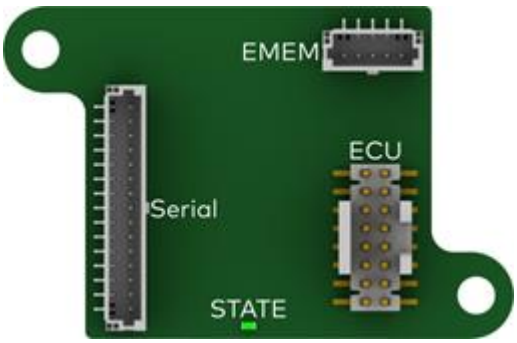


Figure 212: VX1544D top view connectors

- **Serial**
  The Serial connector supplies power to the POD and allows communication between the VX1544D Serial POD and the VX1000 Base Module. The connector is compatible with cables of the type:
  - VX134xB Serial Pigtail (see page 657)
  - VX1311G / VX1311H Serial2 Pigtail (see page 659)
- **EMEM**
  When an Emulation RAM power supply and Calibration Wakeup functionality is needed, a VX1944 EMEM power supply can be attached via this connector.
- **ECU**
  The VX1544D Serial POD can access a microcontroller of a customer ECU or a Development/Evaluation board via the target ECU interface connector.

Port	Connector	Target Interface type
ECU	16-pin Samtec	DAP (2-pin mode, 3-pin wide mode) JTAG Infineon HSCT / PowerPC Zipwire SWD

5.15.3 LEDs

The state of the Serial POD is shown via a green/red LED blink sequence.

The LED indicators show a recurring sequence of colors and blink codes. The following block icons are used to describe this sequence:
□ = off R = red G = green

Sequence	Description
□□□□GGGG	ECU interface disabled.
□G□G□G□G	POD in configuration mode.
GGGGGGGG	ECU interface configured.

Note
The LED behavior has changed since VXtools 4.4. Please see VX1000 Manual V4.3 or earlier for the former LED sequence description.

5.15.4 Technical Data

Dimensions (L/W/H)	35.4 x 23 x 11 mm
PCB Weight	3 g
Power supply	via VX1000 Base Module
Power consumption	0.8 W at max. operating temperature
Operating temperature range	-40 °C ... +115 °C (max. temperature at FPGA case)
Part number	22315

Note
Technical drawings can be found at vector.com/vx1000-drawings.

5.15.5 Target Interface Connector

5.15.5.1 Infineon DAP

ECU (DAP-IFX)

The following table shows the pin assignment of the ECU connector of the VX1544D configured for DAP-IFX. The functional details of the interfaces are described in the section Hardware Integration Guide The pin direction is shown as seen from the POD. For microcontrollers with DAP, the following interface groups are provided:

- DAP is mandatory for ECU access and provides the DAQ functionality
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	IN	-	Strongly recommended*
2	DAP1	DAP	m	IN/OUT	1V8/3V3/5V	
3	GND	-	m	PWR	-	
4	DAP0	DAP	m	OUT	1V8/3V3/5V	
5	GND	-	m	PWR	-	
6	DAP2	DAP	(m)	IN/OUT	1V8/3V3/5V	Mandatory if using DAP Wide Mode
7	GND	-	m	PWR	-	
8	DAPEN	DAP	m	OUT	1V8/3V3/5V	4.75 kΩ pull-up to VREF on POD side
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	1V8/3V3/5V	
11	reserved	-	-	DNC	-	
12	WDGDIS	SYS	o	OUT	1V8/3V3/5V	Allows disabling an external watchdog for flash programming*
13	reserved	-	-	DNC	-	
14	reserved	-	-	DNC	-	
15	VREF	SYS	o	IN	-	Connected to pin 1 on POD side
16	nRSTOUT	SYS	o	IN	1V8/3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD

Pin direction	Description
OUT	Output of the POD
IN/OUT	Input/Output of the POD
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

Note
FTSH connectors are non-locking and therefore must be fixed if a vibration-proof connection is required.

Target (DAP-IFX)

On the target microcontroller side a Samtec FTSH-105-01-L-DV-K-P is required to directly connect the VX1544D Serial POD to the PCB with the target micro-controller. The pin assignment for Infineon DAP is shown in the following table. The pin direction is shown as seen from the target.

Pin	Signal	Group	Rel	Dir	I/O	Details
1	VREF	SYS	o	OUT	-	Strongly recommended*
2	DAP1	DAP	m	IN/OUT	1V8/3V3/5V	
3	GND	-	m	PWR	-	
4	DAP0	DAP	m	IN	1V8/3V3/5V	
5	GND	-	m	PWR	-	
6	DAP2	DAP	(m)	IN/OUT	1V8/3V3/5V	Mandatory if using DAP Wide Mode
7	GND	-	m	PWR	-	
8	DAPEN	DAP	m	IN	1V8/3V3/5V	4.75 kΩ pull-up to VREF on POD side
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	1V8/3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Pin direction	Description
IN	Input of the ECU

Pin direction	Description
OUT	Output of the ECU
IN/OUT	Input/Output of the ECU
PWR	Power supply
DNC	Do not connect

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The VX1544D contains a pull-up resistor of 4.7 kΩ on the DAPEN (nTRST) pin to VREF. During power-up the ECU microcontroller needs a valid logic high level on the nTRST pin to enable the DAP Interface. This must be kept in mind when designing a pull-down resistor on nTRST of the target ECU.

5.15.5.2 JTAG

ECU (JTAG)

The following table shows the pin assignment of the ECU connector of the VX1544D configured for JTAG mode. The functional details of the interfaces are described in the section Hardware Integration Guide The pin direction is shown as seen from the POD. For microcontrollers with JTAG, the following interface groups are provided:

- JTAG is mandatory for ECU access and provides the DAQ functionality
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	IN	-	Strongly recommended*
2	TMS	JTAG	m	OUT	1V8/3V3/5V	
3	GND	-	m	PWR	-	
4	TCK	JTAG	m	OUT	1V8/3V3/5V	
5	GND	-	m	PWR	-	
6	TDO	JTAG	m	IN	1V8/3V3/5V	
7	GND	-	m	PWR	-	
8	TDI	JTAG	m	OUT	1V8/3V3/5V	
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	1V8/3V3/5V	
11	reserved	-	-	DNC	-	
12	WDGDIS	SYS	o	OUT	1V8/3V3/5V	Allows disabling an external watchdog for flash programming*
13	nRDY	JTAG	o	IN	1V8/3V3/5V	Recommended for optimized performance*
14	nTRST/JCOMP	JTAG	(m)	OUT	1V8/3V3/5V	Mandatory if available on target microcontroller

Pin	Signal	Group	Need	Dir	I/O	Details
15	VREF	SYS	o	IN	-	Connected to pin 1 on POD side
16	nRSTOUT	SYS	o	IN	1V8/3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

Note
FTSH connectors are non-locking and therefore must be fixed if a vibration-proof connection is required.

Target (JTAG)

On the target microcontroller side a Samtec FTSH-108-01-L-DV-K-Pis required to directly connect the VX1544D Serial POD to the PCB with the target micro-controller. The pin assignment for JTAG is shown in the following table. The pin direction is shown as seen from the target.

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	OUT	-	Strongly recommended*
2	TMS	JTAG	m	IN	1V8/3V3/5V	
3	GND	-	m	PWR	-	

Pin	Signal	Group	Need	Dir	I/O	Details
4	TCK	JTAG	m	IN	1V8/3V3/5V	
5	GND	-	m	PWR	-	
6	TDO	JTAG	m	OUT	1V8/3V3/5V	
7	GND	-	m	PWR	-	
8	TDI	JTAG	m	IN	1V8/3V3/5V	
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	1V8/3V3/5V	
11	reserved	-	-	DNC	-	
12	WDGDIS	SYS	o	IN	1V8/3V3/5V	Allows disabling an external watchdog for flash programming*
13	nRDY	JTAG	o	OUT	1V8/3V3/5V	Recommended for optimized performance*
14	nTRST/JCOMP	JTAG	(m)	IN	1V8/3V3/5V	Mandatory if available on target microcontroller
15	VREF	SYS	o	OUT	-	Connected to pin 1 on POD side
16	nRSTOUT	SYS	o	OUT	1V8/3V3/5V	

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Pin direction	Description
IN	Input of the ECU
OUT	Output of the ECU
IN/OUT	Input/Output of the ECU
PWR	Power supply
DNC	Do not connect

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

5.15.5.3 Infineon HSCT, Renesas RHSIF, PowerPC Zipwire

ECU

The following table shows the pin assignment of the ECU connector of the VX1544D configured for HSCT/Zipwire/RHSIF mode. The functional details of the interfaces are described in the section Hardware Integration Guide The pin direction is shown as seen from the POD. For microcontrollers with HSCT/RHSIF/Zipwire, the following interface groups are provided:

- HSCT (HSCT/RHSIF/Zipwire) is mandatory for ECU access and provides the DAQ functionality
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	IN	-	Strongly recommended*
2	TXDp	HSCT	m	IN	LVDS	Pairwise required with pin 8 TXDn
3	GND	-	m	PWR	-	
4	REFCLK	HSCT	m	OUT	1V8/3V3/5V	
5	GND	-	m	PWR	-	
6	RXDp	HSCT	m	IN	LVDS	Pairwise required with pin 14 RXDn
7	GND	-	m	PWR	-	
8	TXDn	HSCT	m	IN	LVDS	Pairwise required with pin 2 TXDp
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	1V8/3V3/5V	
11	reserved	-	-	DNC	-	
12	WDGDIS	SYS	o	OUT	1V8/3V3/5V	Allows disabling an external watchdog for flash programming*
13	reserved	-	-	DNC	-	
14	RXDn	HSCT	m	OUT	LVDS	Pairwise required with pin 6 RXDp
15	VREF	SYS	o	IN	-	Connected to pin 1 on POD side
16	nRSTOUT	SYS	o	IN	1V8/3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD

Pin direction	Description
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

Note
FTSH connectors are non-locking and therefore must be fixed if a vibration-proof connection is required.

Target (HSCT/Zipwire)

On the target microcontroller side a Samtec FTSH-105-01-L-DV-K-P is required to directly connect the VX1544D Serial POD to the PCB with the target micro-controller.

5.15.5.4 SWD

ECU (SWD)

The following table shows the pin assignment of the ECU connector of the VX1544D configured for SWD. The functional details of the interfaces are described in the section Hardware Integration Guide The pin direction is shown as seen from the POD. For microcontrollers with SWD, the following interface groups are provided:

- SWD is mandatory for ECU access and provides the DAQ functionality
- SYS (System) signals are additionally provided for realization of different mea-surement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	IN	-	Strongly recommended*
2	SWDIO	SWD	m	IN/OUT	1V8/3V3/5V	
3	GND	-	m	PWR	-	
4	SWCLK	SWD	m	OUT	1V8/3V3/5V	
5	GND	-	m	PWR	-	
6	reserved	-	-	DNC	-	
7	GND	-	m	PWR	-	
8	reserved	-	-	DNC	-	
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	1V8/3V3/5V	
11	reserved	-	-	DNC	-	
12	WDGDIS	SYS	o	OUT	1V8/3V3/5V	Allows disabling an external watchdog for flash programming*
13	reserved	-	-	DNC	-	
14	reserved	-	-	DNC	-	

Pin	Signal	Group	Need	Dir	I/O	Details
15	VREF	SYS	o	IN	-	Connected to Pin 1 on POD side
16	nRSTOUT	SYS	o	IN	1V8/3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD
IN/OUT	Input/Output of the POD
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

Note
FTSH connectors are non-locking and therefore must be fixed if a vibration-proof connection is required. If the VX1544D is ordered with a sealed housing, all connectors on the VX1544D are fixed.

Target (SWD)

On the target microcontroller side a Samtec FTSH-105-01-L-DV-K-P is required to directly connect the VX1544D Serial POD to the PCB with the target microcontroller. The pin assignment for SWD is shown in the following table. The pin direction is shown as seen from the target.

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	OUT	-	Strongly recommended*
2	SWDIO	SWD	m	IN/OUT	1V8/3V3/5V	
3	GND	-	m	PWR	-	

Pin	Signal	Group	Need	Dir	I/O	Details
4	SWCLK	SWD	m	IN	1V8/3V3/5V	
5	GND	-	m	PWR	-	
6	reserved	-	-	DNC	-	
7	GND	-	m	PWR	-	
8	reserved	-	-	DNC	-	
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	1V8/3V3/5V	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Pin direction	Description
IN	Input of the ECU
OUT	Output of the ECU
IN/OUT	Input/Output of the ECU
PWR	Power supply
DNC	Do not connect

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

5.16 VX1621A XPOD

5.16.1 General Information

The VX1621A XPOD is a standalone Measurement and Calibration device with the following features:

- Directly connected to the target ECU.
- Small form factor eases ECU integration.
- 100BASE-T1 Host Interface for XCP on Ethernet.
- Support of Infineon TriCore Aurix devices via DAP.
- Standby mode, if 100BASE-T1 is inactive.
- Emulation Memory (EMEM) power supply (e.g. for calibration over reset).
- Optional: Aluminum mounting frame allowing the VX1621A to be installed in the ECU using an adhesive or screws.

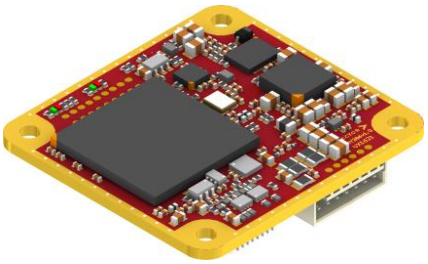


Figure 213: VX1621A XPOD

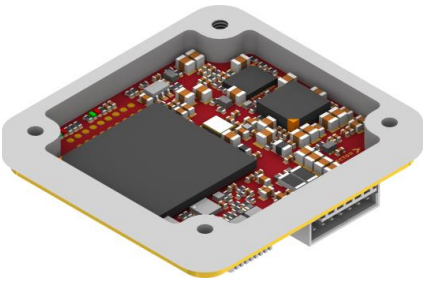


Figure 214: VX1621A with VX1710.01A Mounting Frame

WARNING!
VX1000 Hardware is not designed for high voltage (HV) environments, unless explicitly stated otherwise! If a solution for a HV environment is required, please contact VXsupport@vector.com for guidance.

Note
100BASE-T1 point-to-point connections always require a master node and a slave node. As the VX1621A XPOD is a 100BASE-T1 slave, the link partner must act as 100BASE-T1 master.

Note
The VX1621A XPOD standby mode is entered, if the BR host link is inactive for a defined amount of seconds (configurable via VXconfig). If the 100BASE-T1 host link is up again, the VX1621A XPOD will change from standby mode into normal operation mode.

As long as the VX1621A XPOD inputs are connected to power and if the VX1621A XPOD is configured to supply an attached target Emulation Memory (EMEM), the EMEM will be supplied regardless if the VX1621A XPOD is in standby mode or normal operation mode.

5.16.2 Connectors

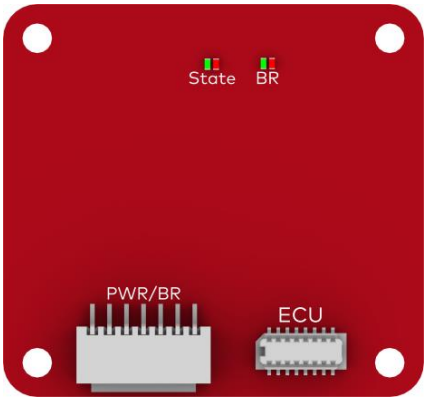


Figure 215: Top view connectors

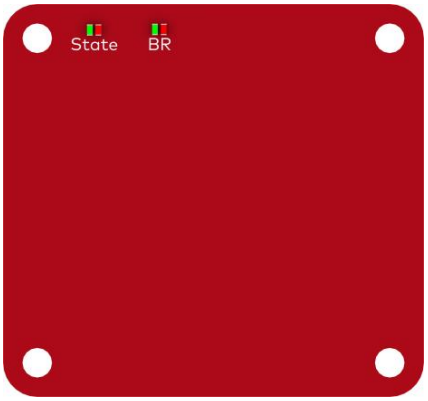


Figure 216: Bottom view connectors

- **PWR/BR**
  The PWR/BR connector (Molex type 87438-0743) has the following uses:
  - providing the VX1621 XPOD with power,
  - providing the EMEM of a target uC with power,
  - providing the CALWAKEUP signal for the target,
  - providing the 100BASE-T1 link.

  The appropriate VX1325.02A cable (length 22 cm, one side with open ends) is delivered together with the VX1621 XPOD. Alternatively, you can select the VX1325.02B cable (length 15 cm, with connectors on both sides).

Pin	Asignment	Color	Direction
1	CALWAKEUP	White	Output of VX1621 XPOD
2	VBAT	Red	-
3	GND	Black	-
4	VEMEM	Yellow	Output of VX1621 XPOD
5	GND	Black	-
6	BRp	Green	Input/Output of VX1621 XPOD
7	BRn	Blue	Input/Output of VX1621 XPOD

- **ECU**
  The ECU connector provides the interface to the target microcontroller of the ECU.

Port	Connector	Interface
ECU	52465-1671 (Molex)	DAP

Reference
Further information on the target connector pin assignments can be found in section Target Interface Connectors on page 562.

5.16.3 Technical Data

Dimensions (L/W/H)	37 x 34.5 x 8.5 mm
PCB Weight	10 g
Power supply	via ECU (5 V ... 32 V, unregulated VBat)
Power consumption (without EMEM supply)	3 W at max. operating temperature Standby: < 15 mW
Operating temperature range	-40 °C ... +120 °C (max. temperature at FPGA case)
Part number	22359

Note
Technical drawings can be found at vector.com/vx1000-drawings.

5.16.4 Target Interface Connectors

5.16.4.1 Infineon DAP

ECU (DAP-IFX)

The following table shows the pin assignment of the ECU connector of the VX1621A XPOD configured for DAP-IFX. The functional details of the interfaces are described in the section Hardware Integration Guide The pin direction is shown as seen from the POD. For microcontrollers with JTAG, the following interface groups are provided:

- DAP is mandatory for ECU access and provides the DAQ functionality
- SYS (System) signals are additionally provided for realization of different measurement and calibration features

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	IN	-	Strongly recommended*
2	nRSTOUT	SYS	m	IN	3V3/5V	
3	DAP1	DAP	m	IN/OUT	3V3/5V	
4	GND	-	m	PWR	-	
5	DAPEN	DAP	m	OUT	3V3/5V	4.75 kΩ pull-up to VREF on POD side
6	nRESET	SYS	m	IN/OUT	3V3/5V	
7	GND	-	m	PWR	-	
8	reserved	-	-	DNC	-	
9	DAP2	DAP	m	IN/OUT	3V3/5V	
10	GND	-	m	PWR	-	
11	reserved	-	-	DNC	-	
12	reserved	-	-	DNC	-	
13	GND	-	m	PWR	-	
14	WDGDIS	SYS	o	OUT	3V3/5V	Allows disabling a watchdog on the ECU for flash programming*
15	DAP0	DAP	m	OUT	3V3/5V	
16	reserved	-	-	DNC	-	

* for more information, please refer to the Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

Pin direction	Description
IN	Input of the POD
OUT	Output of the POD

Pin direction	Description
IN/OUT	Input/Output of the POD
PWR	Power supply
DNC	Do not connect

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig**! Else the POD can be damaged! This restriction does not apply to pins with a defined I/O standard of 5 V.

WARNING!
The VX1621A contains a pull-up resistor of 4.7 kΩ on the DAPEN (nTRST) pin to VREF. During power-up the ECU microcontroller needs a valid logic high level on the nTRST pin to enable the DAP Interface. This must be kept in mind when designing a pull-down resistor on nTRST of the target ECU.

Target cable

For the physical connection to the ECU a target cable is required (see sections VX1903.01A DAP Flex-PCB for XPOD on page 616 and VX1903.01B DAP Flex-PCB for XPOD on page 617).

6 Scripting Language (CASL)

In this chapter you find the following information:

6.1 Calculation and Scripting **565**

6.2 Script Execution Flow **567**

6.3 Configuring CASL Script Execution **569**

6.4 Device Specific Functions **570**

6.4.1 SetConfiguration 570
6.4.2 WriteIO (Standard Write Mode) 573
6.4.3 WriteIO (Read-modify Write Mode) 574
6.4.4 ReadIO 575
6.4.5 GetMicroseconds 576
6.4.6 WaitMicroseconds 576
6.4.7 ElementsOf 577
6.4.8 PrintString 577
6.4.9 PrintChar 578
6.4.10 PrintSignedDec 578
6.4.11 PrintDec 578
6.4.12 PrintHex 578
6.4.13 PrintBin 579

6.5 Script Templates and Examples **579**

6.1 Calculation and Scripting

General information

CASL (Calculation and Scripting Language) is a C-like language with a set of IO functions. CASL scripts can be used to configure deserializer, serializer and sensors in VX1161.51/.52 systems via I2C transactions. They cannot be used to control the VX1161.51/.52 card itself.

For every channel, up to three scripts can be configured. The execution of a script can be disabled while the source code is preserved. The intended use of the script system is that script 1 configures the local phy chip to gain access to the remote phy chip. Script 2 is intended to configure the remote phy chip to gain access to the remote sensor chip and script 3 is intended to configure the remote sensor if required. While all three scripts also could be combined in one script, three scripts allow a more modular approach. All three scripts of one channel are executed on the same I2C bus one after another.

CASL scripts are limited to 64.000 characters of source code per script and 24 kB of binary code per Channel Group (CH1/2 or CH3/4). If this limit is exceeded VXconfig will not allow to download the configuration.

A CASL script must contain exactly only one function with the following signature:

```
function AnyScriptName (unsigned int context)
{
   return 0;
}
```

The name of the script function does not matter. The input parameter must be of type `unsigned int` and must exist but is currently not used. The name of the parameter also does not matter.

In the function body, usual C syntax elements such as variable definition, `do`, `while`, `for`, `switch`, `if..then..else`, `break continue` and `return` elements can be used.

The function body must contain at least one return statement with a return value. If the return value is zero, the script execution is considered successful and complete. If the return value is not zero, the script execution is considered not successful (see section Script Execution Flow on page 567).

Variables can be defined as well as arrays. With VXtools 4.7 SP1 and later large arrays can be declared as CONST to avoid a shortage of RAM in the VX1161.5x firmware.

Structs and unions are not supported.

No data can be stored persistent (static) from one execution cycle to the other. If you need to, you have to store this data in a connected I2C device (deserializer, serializer, sensor, ..) and read it back.

To interact with the outside world, the functions in section Device Specific Functions on page 570 can be used in CASL scripts.

With the function `SetConfiguration()`, environment settings of the interpreter can be set. Some settings control the execution flow, some are intended to simplify the script code and some control the hardware. See section `SetConfiguration()` for details.

For debugging print functions can be used. Their output will be forwarded to the VXconsole tool. If the bandwidth of those prints exceeds the possible bandwidth limit, messages might be lost.

Note

Before VXtools 4.6 the execution of CASL scripts was done in sequence with parts of the VX1161.51 firmware. Blocking CASL code might have affected the LED status update and Link detection.

With VXtools 4.6 and later the execution of CASL scripts can no longer block the VX1161.51/.52 firmware.

6.2 Script Execution Flow

The script execution in the VX1161.51 is controlled by a state machine (FWSM). This state machine supervises the link states and controls the execution of CASL scripts.

The executions of scripts depend on:

- script enable setting
- script return values
- channel mode
- link state
- current state of the FWSM

When a VX1161.51/.52 starts, script execution starts immediately in order to maintain fast link up times. This execution is done before other important system tasks can be executed. If CASL script execution takes longer than 50ms the execution is aborted and retried after the system tasks. The abort will be noted in VXconsole.

Scripts are always executed in ascending order (script 1 first, then script 2, script 3 last). If a script is disabled, it will not be executed but the FWSM will considered it as “executed successfully”.

If a script returns a value of greater than zero, the FWSM will considerer it as “failed and repeat all”. In this case the script execution restarts with script 1.

The channel mode defines whether the FWSM will supervise only one channel (RX only, TX only) or two channels (TAP mode).

If the link state changes (e. g. cable disconnect, system reset, power loss), the FWSM considers the link as broken and the deserializer, serializer and sensor configuration as lost. In this case the scripts must be executed again to reconfigure all components.

Note
Note that any actions inside any CASL script that cause unlocks (e. g. deserializer reset) will cause the FWSM to restart with script 1. This will lead to an endless loop of resets by the CASL script. If required and possible, reset only the deserializer and serializer register set but keep the link stable.

In RXonly or TXonly mode, the FWSM powers and initializes the local deserializer or serializer with essential internal settings to adopt it to the VX1161.51 hardware. Those settings are e. g. clock sources, CSI settings and lock detection mechanisms. All enabled scripts are then executed once on that channel to configure the local deserializer or serializer to be able to detect a link with their remote link partners.

In TAP mode, the TAP deserializer is powered and initialized before the serializer is powered. All enabled scripts of the deserializer channel are then executed once to configure the local deserializer to be able to detect a link with its remote link partner. This behavior avoids indicating a stable link to the ECU before the link between the TAP and the sensor is stable. After the sensor link is stable and all deserializer CASL scripts were executed successfully, the TAP serializer is powered, initialized and its CASL scripts are executed.

As soon as the link is detected, all scripts are executed again to configure the next reachable chip. If one link unlocks or locks, all scripts are executed again to re-configure the next reachable chip. If one script of a channel fails, the execution of all scripts of that channel will be rescheduled after about 10 ms.

Note
Check the VX1161.51 section for your card to see which setting must not be overwritten by CASL scripts.

6.3 Configuring CASL Script Execution

In VX1161.51, CASL scripts are used to configure deserializer, serializer and sensors connected to the card. For every channel, up to three independent scripts can be configured in the **Config Scripts** dialog. In this dialog there are three tabs for selecting Script1..3 of the selected channel. The script source code can be edited in the script edit text box. To test the syntax of the script, the **[Compile script]** button can be used without downloading. To enable the execution of a script, the **Enable** check box in the dialog must be checked.

When the configuration is written to the VX device, the script will be compiled and the script source code will be stored on the VX1161.51. It can be read back via VXconfig via **Read from VX device...** and exported to an INI file. Via this INI file it is also possible to clone all settings of one card to another card.

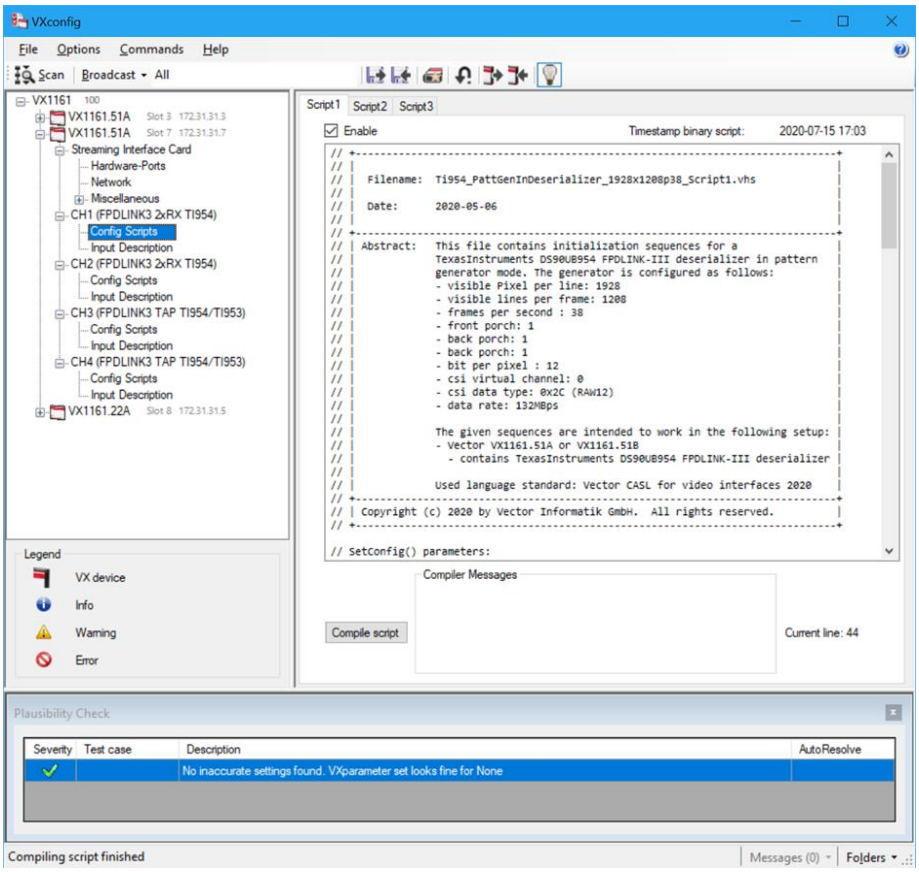


Figure 217: CASL script configuration dialog in VXconfig

6.4 Device Specific Functions

The following sections describe predefined external functions which can be called by CASL scripts.

6.4.1 SetConfiguration

Syntax

```
unsigned int SetConfiguration(
unsigned int ParameterNumber,
unsigned int ParameterValue);
```

Description

Set a configuration parameter to a given value. Configuration parameters define the general operation of the following script code. They can repeatedly be changed in the script if and where required.

Parameters

- **ParameterNumber**
  Defines which parameter to be changed.
- **ParameterValue**
  Value the parameter shall have.

Parameter Number	Parameter Value
0	Specifies how WriteIO() and ReadIO() access errors (e.g. I2C NACK) shall be handled. Both functions will return a value which indicates the success or failure of the access. Usually this return value must be checked by an IF statement which will bloat your code in case of many WriteIOs. When this parameter is set to a value of 1, the script will be terminated instantly with an error message if WriteIO or ReadIO fails. - 1: stop script execution on first error; print error message - 0: continue script execution; use return value to determine success of operations
1	Specifies the 8 bit I2C address to be used for all ReadIO() and WriteIO() accesses.
3	Specifies the mode how WriteIO() accesses shall be handled and with this how many parameters WriteIO() uses. - 0: simple write mode; WriteIO() uses two parameters: address and data - 1: read-modify-write mode; WriteIO() uses three parameters: address, mask and data
4	Specifies how WriteIO() and ReadIO() use its calling parameters to generate address and data transactions on I2C. Format: SetConfig(4, (AA << 8) | (DD << 0)); with AA=address and DD=data format specifier value as of

Width	Dir	AA / DD	Byte order on I2C of WriteIO()/ReadIO() dword parameters[1)]
8 bit	msByte first	VX_ACCESS_8BIT | VX_ACCESS_MSBYF	[7:0]

Parameter Number	Parameter Value			
	16 bit		VX_ACCESS_16BIT | VX_ACCESS_MSBYF	[15:8], [7:0]
	24 bit		VX_ACCESS_24BIT | VX_ACCESS_MSBYF	[23:16], [15:8], [7:0]
	32 bit		VX_ACCESS_32BIT | VX_ACCESS_MSBYF	[31:24], [23:16], [15:8], [7:0]

<table>
<tr><th>Parameter Number</th><th>Parameter Value</th></tr>
<tr><td></td><td>
<table>
<tr><th>Width</th><th>Dir</th><th>AA / DD</th><th>Byte order on I2C of WriteIO()/ReadIO() dword parameters[1)]</th></tr>
<tr><td>8 bit</td><td rowspan="4">lsByte first</td><td>VX_ACCESS_8BIT | VX_ACCESS_LSBYF</td><td>[7:0]</td></tr>
<tr><td>16 bit</td><td>VX_ACCESS_16BIT | VX_ACCESS_LSBYF</td><td>[7:0], [15:8]</td></tr>
<tr><td>24 bit</td><td>VX_ACCESS_24BIT | VX_ACCESS_LSBYF</td><td>[7:0], [15:8], [23:16]</td></tr>
<tr><td>32 bit</td><td>VX_ACCESS_32BIT | VX_ACCESS_LSBYF</td><td>[7:0], [15:8], [23:16], [31:24]</td></tr>
</table>
1) [7:0] means bits 7 to 0 in parameter (=low byte)
Example:

SetConfig(4, ((VX_ACCESS_32BIT | VX_ACCESS_LSBYF) << 8) |
((VX_ACCESS_24BIT | VX_ACCESS_MSBYF) << 0));
// 32 bit address width lsByte first; 24 bit data width msByte first

WriteIO(0xA1A2A3A4, 0xD1D2D3D4);
// outputs bytes SA 0xA4 0xA3 0xA2 0xA1 0xD2 0xD3 0xD4 (SA=SlaveAddress)
</td></tr>
<tr><td>5</td><td>Marks the script as "repeat endlessly". The repetition is executed when all other scripts finished. If several scripts are marked, they are executed one after another. This setting persists the end of the script until it is reset by the script itself or a power cycle.
If both SetConfiguration(5,1) and SetConfiguration(7,1) are used in parallel, ony "repeat endlessly" is used.

- 0: standard behavior; repeat script if it returns with a non-zero value.
- 1: repeat script endlessly independent of return value</td></tr>
<tr><td>6</td><td>Set default script return value for cases where the script is aborted due to access errors or timeouts.</td></tr>
<tr><td>7</td><td>Marks the script as "never repeat". If the script ends normally (not aborted) the script will never be repeated again, even in case of unlocks.
If both SetConfiguration(5,1) and SetConfiguration(7,1) are used in parallel, ony "repeat endlessly" is used.

- 0: standard behavior; repeat script if it returns with a non-zero value or in case of unlocks.
- 1: do not repeat script if it returned once with a zero value</td></tr>
<tr><td>8</td><td>Instructs the firmware to change the 8 bit I2C address of the VX1161.51/.52 channel serializer or deserializer chip with the given value. The firmware will change the address register in the chip and use this address for any further accesses. This overwrites the strapped I2C address configured in VXconfig.</td></tr>
<tr><td>other</td><td>Cause a script abort.</td></tr>
</table>

Return value

Type of unsigned int.

- ► 0: operation succeeded
- ► 1: operation failed

Example
Typical script parameter example:

```
// stop script execution on first IO error;
// 0: do not stop; 1:stop it
SetConfiguration(0,0);

// set i2c slave address for all following accesses
SetConfiguration(1,0x78);

// set write mode; 0:write; 1:read-modify-write;
SetConfiguration(3,0);

// set how address and data are output/input in I2C bus
// SetConfig(4, (AA << 8) | (DD << 0)); with AA=configuration for address
and DD=configuration for data parameters

// Predefined constants:
// VX_ACCESS_8BIT, VX_ACCESS_16BIT, VX_ACCESS_24BIT, VX_ACCESS_32BIT for
1..4 bytes

// VX_ACCESS_MSBYF, VX_ACCESS_LSBYF for outputting parameter value
most/least significant byte first

// e.g. (VX_ACCESS_16BIT|VX_ACCESS_MSBYF) sends a value of 0x12345678 on
I2C bus as 0x56, 0x78

// e.g. (VX_ACCESS_24BIT|VX_ACCESS_LSBYF) sends a value of 0x12345678 on
I2C bus as 0x78, 0x56, 0x34
SetConfig(4, ( (VX_ACCESS_16BIT|VX_ACCESS_MSBYF) << 8) | ( (VX_ACCESS_
8BIT|VX_ACCESS_LSBYF) << 0)); // 16 bit address width msByte first; 8 bit
data width
```

6.4.2 WriteIO (Standard Write Mode)

Syntax

```
unsigned int WriteIO(
unsigned int RegisterAddress,
unsigned int RegisterValue);
```

Description

Writes registers in an I2C slave with a given value.

Depending on the `SetConfiguration(3,x)` setting, this command executes standard register writes or read-modify-writes. The I2C slave address is defined by `SetConfiguration(1, SA)`.

Depending on the `SetConfiguration(4,x)` setting, this command executes register accesses with 8 or 16 bit register addresses and data. 16 bit accesses will be transferred as two I2C byte transfers with most significant byte first.

Parameters

- **RegisterAddress**
  Defines the register address.
- **RegisterValue**
  Defines the value to be written to the register.

Return value

Type of unsigned int.

- 0: operation succeeded
- 1: operation failed

Example

```
// set i2c slave address
SetConfiguration(1,0x78);

// set write mode to standard write;
SetConfiguration(3,1);

// set address and data width to 8 bit
SetConfiguration(4,0x0101);

// write 0x16 to register 0x1A
if (WriteIO(0x1A,0x16)) {
  PrintString("Write failed");
}

// transactions on I2C bus:
// S,0x78,0x1A,0x16,P
```

Example

```
// set i2c slave address
SetConfiguration(1,0x78);

// set write mode to standard write;
SetConfiguration(3,1);

// set address and data width to 16 bit
SetConfiguration(4,0x0202);

// write 0x1617 to register 0x1A1B
if (WriteIO(0x1A1B,0x1617)) {
    PrintString("Write failed\n");
}

// transactions on I2C bus:
// S,0x78,0x1A,0x1B,0x16,0x17,P
```

6.4.3 WriteIO (Read-modify Write Mode)

Syntax

```
unsigned int WriteIO (
unsigned int RegisterAddress,
unsigned int ValueMask,
unsigned int RegisterValue);
```

Description

Reads, masks and writes back the value of an I2C register

Depending on the `SetConfiguration(4,x)` setting, this command executes register accesses with 8 or 16 bit register addresses and data and thus via one or two I2C byte transfers per address and data. Only set bits in **ValueMask** will be modified.

This function executes the following steps:

1. Read from the given register address the register value.
2. Mask the read value where set ValueMask-bits are cleared from the read value.
3. Mask the RegisterValue parameter where non-set ValueMask-bits are cleared.
4. Do logic OR of the masked read value and the masked RegisterValue
5. Write the new register value back to the given register address.

Logic expression:
`Register = (Register & ~ValueMask) | (RegisterValue & ValueMask)`

Parameters

- **RegisterAddress**
  Defines the register address.
- **ValueMask**
  Defines which bits to be modified; set bits are modified.
- **RegisterValue**
  Defines the value of the bits to be modified.

Return value

Type of unsigned int.

- 0: operation succeeded
- 1: operation failed

Example

```
// set i2c slave address
SetConfiguration(1,0x78);

// set write mode to read-modify-write
SetConfiguration(3,1);

// set size of address and data
SetConfiguration(4,0x0101);

// read register 0x1A and set only bit 4, write result back
if (WriteIO(0x1A, 0x10, 0x16)) {
  PrintString("Write failed");
}

// transactions on I2C bus:
// S,0x78,0x1A -- write register address
// Sr,0x79,0x04,P -- read value; e.g.0x04
// S,0x78,0x1A,0x14,P -- write register value back; e. g. 0x14
```

6.4.4 ReadIO

Syntax

```
unsigned int ReadIO (
unsigned int RegisterAddress,
unsigned int &RegisterValue);
```

Description

Reads a value from an I2C register. The I2C slave address for this access is determined by `SetConfiguration(1, SA)`.

Depending on the `SetConfiguration(4,x)` setting, this command executes register accesses with 8 or 16 bit register addresses and data. 16 bit accesses will be transferred as two I2C byte transfers with most significant byte first.

Read transactions consist of two I2C transactions: Writing the register address and reading the register value. The read transaction is initiated via a repeated start condition to keep the bus occupied and enforce an uninterruptible sequence.

Parameters

- **RegisterAddress**
  Defines the register address.

- **RegisterValue**
  Variable to store read register value.

Return value

Type of unsigned int.

- 0: operation succeeded
- 1: operation failed

Example

```
// set address and data width to 8 bit
SetConfiguration(4,0x0101);

// set i2c slave address
SetConfiguration(1,0x78);

// define variable to receive read value
unsigned int readValue;

// read from register 0x1A
if (ReadIO(0x1A,readValue)) {
  PrintString("Reading failed\n");
}

// transactions on I2C bus:
// S,0x78,0x1A -- write register address
// Sr,0x79,0x04,P -- read value; e. g.0x04
```

6.4.5 GetMicroseconds

Syntax

```
unsigned int GetMicroseconds (unsigned int &now);
```

Description

Returns the script engine time in microseconds.

Parameters

- **now**
  Variable to store read time value.

Return value

Type of unsigned int.

- 0: operation succeeded
- 1: operation failed

Example

```
unsigned int timeInMicroseconds;
GetMicroseconds(timeInMicroseconds);
```

6.4.6 WaitMicroseconds

Syntax

```
unsigned int WaitMicroseconds (
unsigned int MicroSecondsToWait);
```

Description

Suspends the script execution for the specified number of microseconds.

Parameters

- **MicroSecondsToWait**
  Defines how long the script execution shall be suspended. The true granularity for suspensions is 1 ms = 1000 µs. Fractions of milliseconds are ignored.

Note
The execution of CASL scripts is done in sequence with parts of the VX1161.51 firmware. Blocking CASL code might affect the LED status update and Link detection.

Return value

Always 1.

Example

```
// wait 10 ms
WaitMicroseconds(10*1000);
```

6.4.7 ElementsOf

Syntax

```
unsigned int ElementsOf(Type Array);
```

Description

Returns the number of elements in an array.

Parameters

- **Type Array**
  Returns the elements contained in an array.

Return value

Type of unsigned int. Number elements in a given array.

Example

```
const unsigned int Array[] = {0,1,2,3,4,5};
unsigned int ArraySize;

ArraySize = ElementsOf(Array); // ArraySize is now 6.
```

Note
With VXtools 4.7 SP1 and later large arrays can be declared as **CONST** to avoid a shortage of RAM in the VX1161.5x firmware.

6.4.8 PrintString

Syntax

```
unsigned int PrintString(StringLiteral text);
```

Description

Prints a string literal to VxConsole.

Parameters

- **text**
  Defines the text to be sent to VxConsole.

Return value

Type of unsigned int.

- 0: operation succeeded
- 1: operation failed

Example

```
PrintString(„This is a string”);
```

6.4.9 PrintChar

Syntax

```
unsigned int PrintChar(unsigned int Value1,…);
```

Description

Prints a char to VXconsole.

Parameters

- **Value1,...**
  Chars to be printed.

Return value

Type of unsigned int.

- 0: operation succeeded
- 1: operation failed

6.4.10 PrintSignedDec

Syntax

```
unsigned int PrintSignedDec (
int Value,
unsigned int minDigits);
```

Description

Prints a signed decimal to VXconsole.

Parameters

- **Value**
  Value to be printed.
- **minDigits**
  Number of digits the print shall have at least.

Return value

Type of unsigned int.

- 0: operation succeeded
- 1: operation failed

6.4.11 PrintDec

Syntax

```
unsigned int PrintDec (
unsigned Value,
unsigned int minDigits);
```

Description

Prints a unsigned decimal to VXconsole.

Parameters

- **Value**
  Value to be printed.
- **minDigits**
  Number of digits the print shall have at least.

Return value

Type of unsigned int.

- 0: operation succeeded
- 1: operation failed

6.4.12 PrintHex

Syntax

```
unsigned int PrintHex (
unsigned Value,
```

```
unsigned int minDigits);
```

Description

Prints a hexadecimal to VXconsole.

Parameters

- **Value**
  Value to be printed.
- **minDigits**
  Number of digits the print shall have at least.

Return value

Type of unsigned int.

- 0: operation succeeded
- 1: operation failed

6.4.13 PrintBin

Syntax

```
unsigned int PrintBin (
unsigned Value,
unsigned int minDigits);
```

Description

Prints a binary to VXconsole.

Parameters

- **Value**
  Value to be printed.
- **minDigits**
  Number of digits the print shall have at least.

Return value

Type of unsigned int.

- 0: operation succeeded
- 1: operation failed

6.5 Script Templates and Examples

Reference

You can find CASL scripts in the VXtools installation folder in:
`\Doc\Examples\Streaming_CASL_scripts\example scripts`

Files with file name `*_Script1.vhs` usually contain deserializer settings.
Files with file name `*_Script2.vhs` usually contain serializer settings.
Files with file name `*_Script3.vhs` usually contain sensor settings.

7 Accessories

In this chapter you find the following information:

7.1 Adapters **582**
7.1.1 VX1912 PowerPC JTAG Adapter 582
7.1.2 VX1922 x850 JTAG Adapter 584
7.1.3 VX1923A XC2000 DAP OCDS-L1 Adapter 586
7.1.4 VX1923B TriCore DAP OCDS-L1 Adapter 588
7.1.5 VX1926 V850E2 JTAG 20-Pin Adapter 590
7.1.6 VX1936 V850E2 JTAG 16-Pin Adapter 592
7.1.7 VX1944 EMEM Power Supply 594
7.1.8 VX1927A JTAG-PT FlexPCB for VX154x 596
7.1.9 VX1927B DAP-PT FlexPCB for VX154x 599
7.1.10 VX1927C DAP-IFX FlexPCB for VX154x (obsolete) 602
7.1.11 VX1927D Zipwire/HSCT FlexPCB for VX154x 605
7.1.12 VX1927E DAP-IFX FlexPCB for VX154x (obsolete) 607
7.1.13 VX1927H DAP-IFX FlexPCB for VX154x 610
7.1.14 VX1927J DAP-IFX FlexPCB for VX154x 613
7.1.15 VX1903.01A DAP Flex-PCB for XPOD 616
7.1.16 VX1903.01B DAP Flex-PCB for XPOD 617

7.2 EEK Heads **618**
7.2.1 VX1902.02A EEK Head Aurora (Infineon) 618
7.2.2 VX1902.02B EEK Head AUR PCB (Infineon) 620
7.2.3 VX1902.03A EEK Head Aurora (NXP/Renesas) 622
7.2.4 VX1902.03B EEK Head AUR PCB (NXP/Renesas) 624
7.2.5 VX1902.04A EEK Head Aurora (ARM) 626
7.2.6 VX1902.04B EEK Head AUR PCB (ARM) 628

7.3 CAN Cables **630**
7.3.1 VX1361C CAN Cable LEMO/D-SUB 0.75m 630
7.3.2 VX1361D CAN Cable LEMO/D-SUB 0.75m 631
7.3.3 VX1362A CAN Cable LEMO 1.5m 632
7.3.4 VX1365A CAN Cable LEMO 5m 633
7.3.5 VX1362B CAN Cable LEMO/Banana Plugs 1.5m 634
7.3.6 CANcable 2Y 635

7.4 FAKRA Cables **636**
7.4.1 VX1382D/E/F/G FAKRA Cables 636
7.4.2 VX1382H FAKRA Extention 0.1m Cable 637
7.4.3 VX1382M.01 FAKRA Panelmount F0N100N0F (U.FL) 638

7.5 HSSL Cables **639**
7.5.1 VX137x HSSL Cables 639

7.6 HSSL2 Cables **642**
7.6.1 VX1381 HSSL2 Cables 642

7.7 Power/Sync Cables **645**
7.7.1 Vector Power Supply 12V/1.5A 645
7.7.2 Vector Power Supply ODU Mini-Snap 646
7.7.3 Desktop Type Power Supply 649
7.7.4 Cable Vehicle Input <> ODU Mini-Snap 650
7.7.5 Power Cable Molex Mini-Fit 4mm² 650
7.7.6 Cable Banana Plug <> Binder 3-pin 651
7.7.7 Power Sync Cable 2xBinder <> Banana Plugs 652
7.7.8 Cable Banana Plug <> ODU Mini-Snap 653
7.7.9 Car Power Supply Cable 12V with Binder 654
7.7.10 Cable Binder 3pol Connector with Pigtail 655
7.7.11 Vector SYNCcableXL 656

7.8 Serial Cables **657**
7.8.1 VX134x Serial Cables 657
7.8.2 VX1311 Serial2 Cables 659

7.9 Ethernet Cables **661**
7.9.1 AEcable 2Y Family 661
7.9.2 Ethernet Cable 10Gbps 0.5m 667
7.9.3 Ethernet Cable 10Gbps 2m 668
7.9.4 Ethernet Cable 10Gbps 7.5m 669
7.9.5 Ethernet Cable Robust 10Gbps 0.5m 670
7.9.6 Ethernet Cable Robust 10Gbps 2.0m 671
7.9.7 Ethernet Cable Robust 10Gbps 7.5m 672

7.10 FlexRay Cables **673**
7.10.1 FRcable AB 673
7.10.2 FR/CANcable 2Y 674

7.11 Mounting Equipment **675**
7.11.1 Fix Kit 32mm Device 675
7.11.2 Fix Kit Large Device 676
7.11.3 Serial POD Enclosure Gen2 677
7.11.4 VSH Cable Guard 216 678
7.11.5 VSH Cable Guard 216 Clip 681
7.11.6 VSH Connection Kit 19" 682
7.11.7 VSH Connection Kit Horizontal 684
7.11.8 VSH Connection Kit Vertical 685
7.11.9 VSH Equipment Foot Kit 686
7.11.10 VSH Mounting Flange Kit 687

7.1 Adapters

7.1.1 VX1912 PowerPC JTAG Adapter

7.1.1.1 General information

The VX1912 PowerPC EVB JTAG Adapter allows connecting the VX154x Serial POD JTAG Interface to NXP/ST PowerPC Evaluation Boards.

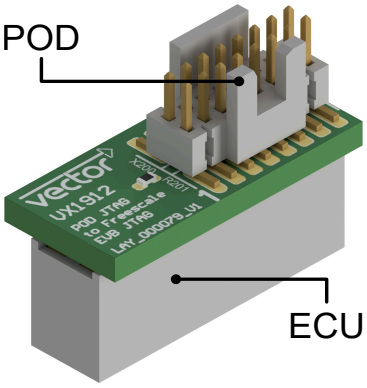


Figure 218: VX1912 PowerPC EVB JTAG Adapter

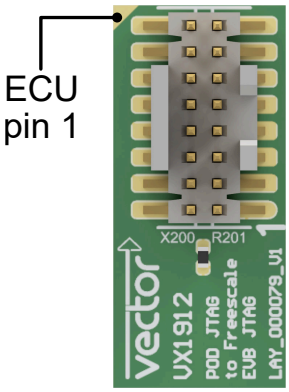


Figure 219: VX1912 side view

- **POD (Samtec)**
  16-pin connector for VX154x Serial POD with JTAG interface option.
  Pin 1 is indicated by a "1" next to the connector.
- **ECU**
  14-pin socket for JTAG interface of PowerPC Evaluation Boards.
  Pin 1 is indicated by a blank (golden) copper corner on the top layer of the adapter PCB.

Technical data

Dimensions (L/W/H)	18.5 x 8.5 x 15.5 mm
Weight	2 g
Power supply	none (passive adapter)
Temperature range	-40 °C ... +115 °C
Part number	22220

7.1.1.2 Target Interface Connectors

Port	Vendor	VX1543A connector type
POD	Samtec	FTSH-108-01-L-DV-K
ECU	Würth Elektronik	610 014 218 21

Note
The pin assignment of the POD connector can be found in section VX1543A Serial POD (Obsolete) on page 493.

ECU (JTAG)

The following table shows the pin assignment of the ECU connector of the VX1912 PowerPC EVB JTAG Adapter.

ECU (JTAG)			ECU (JTAG)		
Pin	Assignment	Direction	Pin	Assignment	Direction
1	TDI	IN	2	GND	PWR
3	TDO	OUT	4	GND	PWR
5	TCK	IN	6	GND	PWR
7	-	NC	8	-	NC
9	nRESET	IN/OUT	10	TMS	IN
11	VREF	OUT	12	GND	PWR
13	nRDY	OUT	14	JCOMP	IN

Pin direction

PWR	Power supply	IN	Input of µC
-	Reserved (input) pin	OUT	Output of µC
NC	Not connected	IN/OUT	Input/Output of µC
DNC	**Do not connect**		

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in VXconfig otherwise the POD may be damaged. This restriction does not apply to pins with the direction "NC" or listed as "5 V tolerant".

7.1.2 VX1922 x850 JTAG Adapter

7.1.2.1 General Information

The VX1922 allows connecting the VX154x Serial POD JTAG Interface to certain third-party Evaluation Boards with Renesas x850 microcontrollers.

WARNING!
The VX1922 requires a specific pin assignment on the Evaluation Board which is incompatible with the default pin assignment that is used by Renesas.

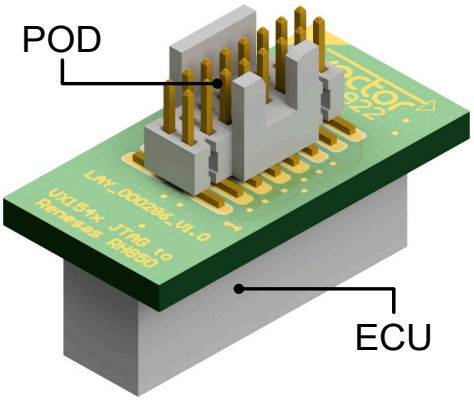


Figure 220: VX1922

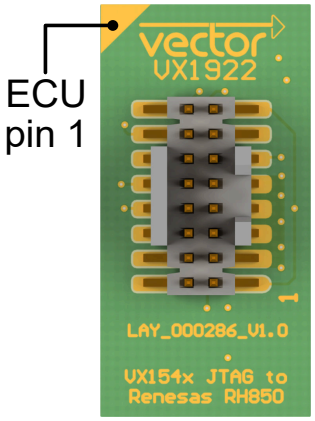


Figure 221: VX1922 top view

- **POD (Samtec)**
  16-Pin socket for VX154x Serial POD with JTAG interface option. Pin 1 of the POD connector is indicated by "1" on the PCB top side.
- **ECU**
  14-pin socket Renesas x850 Evaluation Board (JTAG).
  Pin 1 is indicated by a blank (golden) copper corner on the top layer of the adapter PCB.

Note
Technical drawings can be found at vector.com/vx1000-drawings.

Dimensions (L/W/H)	21.2 x 10.7 x 15.2 mm
Weight	2 g
Power supply	none (passive adapter)
Temperature range	-40 °C ... +115 °C
Part number	22326

7.1.2.2 Target Interface Connectors

Port	Vendor	VX1543A connector type
POD	Samtec	FTSH-108-01-L-DV-K-P
ECU	Würth Elektronik	610 014 218 21

Note
Both connectors are non-locking and therefore must be fixed if a vibration-proof connection is required.

Note
The pin assignment of the POD connector can be found in section VX1543A Serial POD (Obsolete) on page 493.

ECU (JTAG)

The following table shows the pin assignment of the ECU connector of the VX1922 x850 JTAG 14-Pin Adapter.

ECU (Renesas x850)			ECU (Renesas x850)		
Pin	Assignment	Direction	Pin	Assignment	Direction
1	TCK	IN	2	GND	PWR
3	nTRST	IN	4	FLMD0	IN
5	TDO	OUT	6	FLMD1	-
7	TDI	IN	8	VREF	OUT
9	TMS	IN	10	FLMD2	-
11	nRDY	OUT	12	GND	PWR
13	nRESET	IN/OUT	14	GND	PWR

Pin direction

PWR	Power supply	IN	Input of µC
-	Reserved (input) pin	OUT	Output of µC
NC	Not connected	IN/OUT	Input/Output of µC
DNC	**Do not connect**		

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig** otherwise the POD may be damaged. This restriction does not apply to pins with the direction "NC" or listed as "5 V tolerant".

7.1.3 VX1923A XC2000 DAP OCDS-L1 Adapter

7.1.3.1 General information

The VX1923A XC2000 DAP OCDS-L1 Adapter allows connecting the VX154x Serial POD DAP Interface to Infineon Evaluation Boards.

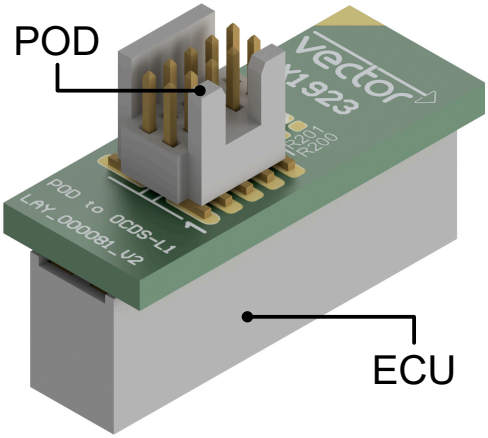


Figure 222: VX1923A XC2000 DAP OCDS-L1 Adapter

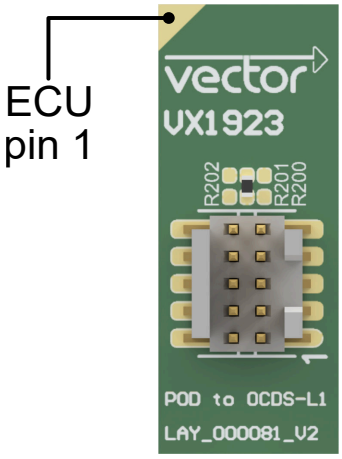


Figure 223: VX1923A top view

- **POD (Samtec)**
  10-pin connector for VX154x Serial POD with DAP (IFX) interface option.
  Pin 1 is indicated by a "1" next to the connector.
- **ECU**
  16-pin socket for OCDS-L1 interface of Infineon Evaluation Board.
  Pin 1 is indicated by a blank (golden) copper corner on the top layer of the adapter PCB.

Dimensions (L/W/H)	21 x 8.5 x 15.5 mm
Weight	3 g
Power supply	none (passive adapter)
Temperature range	-40 °C ... +115 °C
Part number	22339

7.1.3.2 Target Interface Connectors

Port	Vendor	VX1543A connector type
POD	Samtec	FTSH-108-01-L-DV-K
ECU	Würth Elektronik	610 014 218 21

Note
Both connectors are non-locking and therefore must be fixed if a vibration-proof connection is required.

Note
The pin assignment of the POD connector can be found in section VX1543A Serial POD (Obsolete) on page 493.

ECU XC2000 (OCDS-L1)

The following table shows the pin assignment of the ECU connector of the VX1923A XC2000 DAP OCDS-L1 Adapter for XC2000 microcontroller.

ECU XC2000 (OCDS Level 1)		
Pin	Assignment	Direction
1	-	NC
3	DAP1	IN/OUT
5	-	NC
7	-	NC
9	DAPEN/nTRST	IN
11	DAP0	IN
13	-	NC
15	-	NC

ECU XC2000 (OCDS Level 1)		
Pin	Assignment	Direction
2	VREF	OUT
4	GND	PWR
6	GND	PWR
8	nRESET	IN/OUT
10	-	NC
12	GND	PWR
14	-	NC
16	-	NC

Pin direction

PWR	Power supply
-	Reserved (input) pin
NC	Not connected
DNC	**Do not connect**

IN	Input of µC
OUT	Output of µC
IN/OUT	Input/Output of µC

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig** otherwise the POD may be damaged. This restriction does not apply to pins with the direction "NC" or listed as "5 V tolerant".

7.1.4 VX1923B TriCore DAP OCDS-L1 Adapter

7.1.4.1 General information

The VX1923B TriCore DAP OCDS-L1 Adapter allows connecting the VX154x Serial POD DAP Interface to Infineon Evaluation Boards.

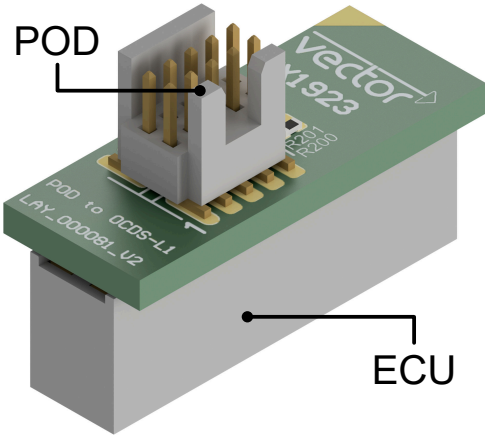


Figure 224: VX1923B TriCore DAP OCDS-L1 Adapter

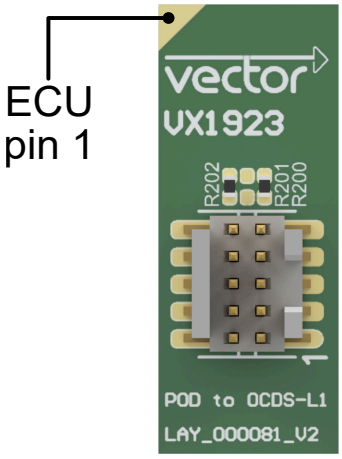


Figure 225: VX1923B top view

- **POD (Samtec)**
  10-pin connector for VX154x Serial POD with DAP (IFX) interface option.
  Pin 1 is indicated by a "1" next to the connector.
- **ECU**
  16-pin socket for OCDS-L1 interface of Infineon Evaluation Board.
  Pin 1 is indicated by a blank (golden) copper corner on the top layer of the adapter PCB.

Dimensions (L/W/H)	21 x 8.5 x 15.5 mm
Weight	3 g
Power supply	none (passive adapter)
Temperature range	-40 °C ... +115 °C
Part number	22340

7.1.4.2 Target Interface Connectors

Port	Vendor	VX1543A connector type
POD	Samtec	FTSH-108-01-L-DV-K
ECU	Würth Elektronik	610 014 218 21

Note
Both connectors are non-locking and therefore must be fixed if a vibration-proof connection is required.

Note
The pin assignment of the POD connector can be found in section VX1543A Serial POD (Obsolete) on page 493.

ECU TriCore (OCDS-L1)

The following table shows the pin assignment of the ECU connector of the VX1923 DAP OCDS-L1 Adapter for TriCore microcontroller.

The VX1923 DAP OCDS-L1 to VX154x DAP Adapter for TriCore microcontrollers has two assembled resistors on the top side.

ECU Tricore (OCDS Level1)		
Pin	Assignment	Direction
1	DAP1	IN/OUT
3	DAP2	IN/OUT
5	-	NC
7	-	NC
9	DAPEN/nTRST	IN
11	DAP0	IN
13	-	NC
15	-	NC

ECU TriCore (OCDS Level 1)		
Pin	Assignment	Direction
2	VREF	OUT
4	GND	PWR
6	GND	PWR
8	nRESET	IN/OUT
10	-	NC
12	GND	PWR
14	-	NC
16	-	NC

Pin direction

PWR	Power supply
-	Reserved (input) pin
NC	Not connected
DNC	**Do not connect**

IN	Input of µC
OUT	Output of µC
IN/OUT	Input/Output of µC

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig** otherwise the POD may be damaged. This restriction does not apply to pins with the direction "NC" or listed as "5 V tolerant".

7.1.5 VX1926 V850E2 JTAG 20-Pin Adapter

7.1.5.1 General information

The VX1926 V850E2 EVB JTAG-20 Adapter allows connecting the VX154x Serial POD JTAG Interface to Renesas/NEC V850E2 Evaluation Boards.

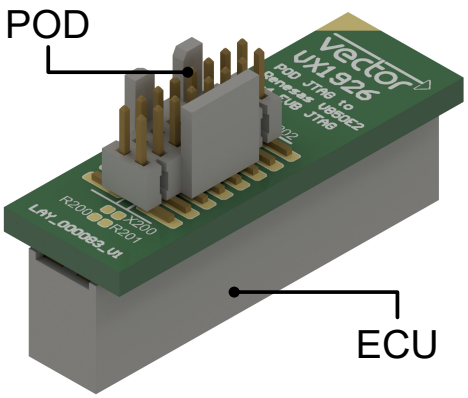


Figure 226: VX1926 V850E2 EVB JTAG-20 Adapter

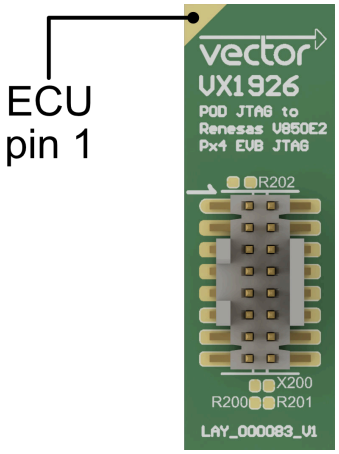


Figure 227: VX1926 top view

- **POD (Samtec)**
  16-pin conntector for VX154x Serial POD with JTAG interface option.
- **ECU**
  20-pin socket Renesas/NEC V850E2 Evaluation Board (JTAG).
  Pin 1 is indicated by a blank (golden) copper corner on the top layer of the adapter PCB.

Dimensions (L/W/H)	26 x 9 x 15.5 mm
Weight	3 g
Power supply	none (passive adapter)
Temperature range	-40 °C ... +115 °C
Part number	22230

7.1.5.2 Target Interface Connectors

Port	Vendor	VX1543A connector type
POD	Samtec	FTSH-108-01-L-DV-K
ECU	Würth Elektronik	610 014 218 21

Note
Both connectors are non-locking and therefore must be fixed if a vibration-proof connection is required.

Note
The pin assignment of the POD connector can be found in section VX1543A Serial POD (Obsolete) on page 493.

ECU (JTAG)

The following table shows the pin assignment of the ECU connector of the VX1926 V850E2 EVB JTAG-20 Adapter.

ECU (Renesas/NEC V850E2)			ECU (Renesas/NEC V850E2)		
Pin	Assignment	Direction	Pin	Assignment	Direction
1	GND	PWR	2	DCUTCK	IN
3	GND	PWR	4	DCUTMS	IN
5	GND	PWR	6	DCUTDI	IN
7	GND	PWR	8	nDCUTRST	IN
9	GND	PWR	10	-	NC
11	GND	PWR	12	nRESET	IN/OUT
13	GND	PWR	14	FLMD0	IN
15	GND	PWR	16	nDCURDY	OUT
17	GND	PWR	18	DCUTDO	OUT
19	GND	PWR	20	VREF	OUT

Pin direction

PWR	Power supply
-	Reserved (input) pin
NC	Not connected
DNC	**Do not connect**

IN	Input of µC
OUT	Output of µC
IN/OUT	Input/Output of µC

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig** otherwise the POD may be damaged. This restriction does not apply to pins with the direction "NC" or listed as "5 V tolerant".

7.1.6 VX1936 V850E2 JTAG 16-Pin Adapter

7.1.6.1 General Information

The VX1936 allows connecting the VX154x Serial POD JTAG Interface to certain third-party Evaluation boards with Renesas/NEC V850E2 microcontrollers.

WARNING!
The VX1936 requires a specific pin assignment on the Evaluation Board which is incompatible with the default pin assignment that is used by Renesas.

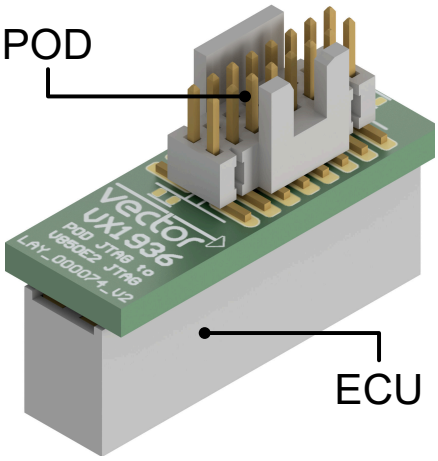


Figure 228: VX1936

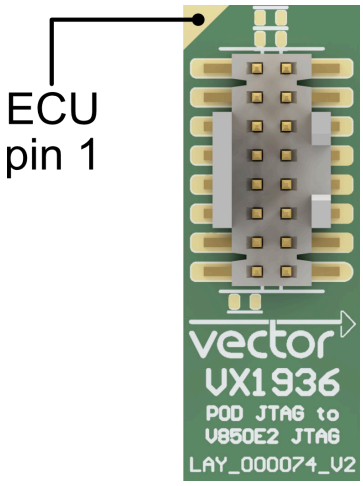


Figure 229: VX1936 top view

- **POD (Samtec)**
  16-pin conntector for VX154x Serial POD with JTAG interface option.
- **ECU**
  16-pin socket Renesas/NEC V850E2 Evaluation Board (JTAG).
  Pin 1 is indicated by a blank (golden) copper corner on the top layer of the adapter PCB.

Dimensions (L/W/H)	21 x 8 x 15.5 mm
Weight	3 g
Power supply	none (passive adapter)
Temperature range	-40 °C ... +115 °C
Part number	22246

7.1.6.2 Target Interface Connectors

Port	Vendor	VX1543A connector type
POD	Samtec	FTSH-108-01-L-DV-K
ECU	Würth Elektronik	610 016 218 21

Note
Both connectors are non-locking and therefore must be fixed if a vibration-proof connection is required.

Note
The pin assignment of the POD connector can be found in section VX1543A Serial POD (Obsolete) on page 493.

ECU (JTAG)

The following table shows the pin assignment of the ECU connector of the VX1936 V850E2 EVB JTAG-16 Adapter.

ECU (Renesas/NEC V850E2)			ECU (Renesas/NEC V850E2)		
Pin	Assignment	Direction	Pin	Assignment	Direction
1	GND	PWR	2	nRESET	IN/OUT
3	DCUTDO	OUT	4	VREF	OUT
5	DCUTDI	IN	6	-	NC
7	DCUTCK	IN	8	nDCURDY	OUT
9	nDCUTRST	IN	10	-	NC
11	-	NC	12	DCUTMS	IN
13	-	NC	14	FLMD0	IN
15	-	NC	16	-	NC

Pin direction

PWR	Power supply	IN	Input of µC
-	Reserved (input) pin	OUT	Output of µC
NC	Not connected	IN/OUT	Input/Output of µC
DNC	**Do not connect**		

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig** otherwise the POD may be damaged. This restriction does not apply to pins with the direction "NC" or listed as "5 V tolerant".

7.1.7 VX1944 EMEM Power Supply

7.1.7.1 Introduction

The VX1944 is able to provide a DC regulated power to an Emulation Memory (EMEM) and a microcontroller debug interface. The VX1944 itself is powered from the car battery and has the following features:

- Small footprint with one-sided parts assembly
- Two M2.5 mounting holes for ECU integration
- Wide input voltage range of 3.5 V ... 32 V
- Two power supply outputs powering EMEM and debug interface
- EMEM power monitor
- Calibration Wakeup signaling
- Reverse polarity protection for VBAT and GND_BAT
- Robust against Automotive Load Transients
- Intended use: Customer ECU integration and laboratory environment

7.1.7.2 Connectors

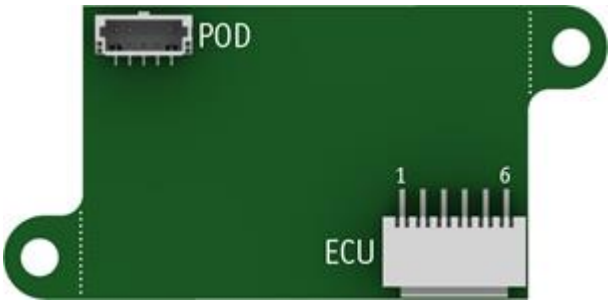


Figure 230: Top view

- **POD (Samtec)**
  The POD connector (Samtec type T1M-GF-S-V) provides an interface to the VX1544 Serial POD or the VX1452/VX1453 HSSL POD reporting EMEM Power Good status and looping through the CALWAKEUP signal.
- **ECU (Molex)**
  The ECU connector (Molex type 87438-0642) provides an interface to the customer ECU. The pin assignment is shown in the following table.

Pin	Assignment	Direction
1	VEMEM	PWR OUT
3	VCC	PWR OUT
5	VBAT	PWR IN

Pin	Assignment	Direction
2	GND	PWR OUT
4	GND_BAT	PWD IN
6	CALWAKEUP	OUT

Pin direction

PWR IN	Power supply input
PWR OUT	Power supply output

OUT	Power supply Output

7.1.7.3 Technical Data

Input Voltage range	3.5 V ... 32 V DC
Output Voltage	1.3 V DC (VX1944) 3.3 V DC (VX1944A)
Temperature range	-40 °C ... +125 °C
Humidity	noncondensing
ECU pins	
CALWAKEUP	The CALWAKEUP signal provides a positive output voltage derived from VBAT during a Calibration Wakeup and is tri-stated otherwise.
Output Voltage	0 V ... VBAT (VX1944A)
VBAT & GND_BAT	VBAT and GND_BAT is the power supply input of the VX1944 EMEM Power Supply.
Input Voltage (max.)	3.5 V ... 32 V DC
Input Current (max.)	600 mA (VBAT = 3.5 V, full load)
Reverse Polarity Protection	Yes (between VBAT and GND_BAT)
Transient Protection	ISO7637-2
VEMEM	The output voltage VEMEM (referenced to GND) is provided to power Emulation Memory of the Target Microcontroller.
Output Voltage	1.3 V DC ± 2.5% (VX1944A)
Output Current (max.)	800 mA (VBAT = 3.5 V ... 3.6 V) 1000 mA (VBAT > 3.6 V)
Dimensions (L/W/H)	43.4 x 21 x 7.6 mm
Weight	6 g
Part number	22259

7.1.8 VX1927A JTAG-PT FlexPCB for VX154x

7.1.8.1 General Information

The VX1927A flexible PCB adapter allows impedance-controlled connection of a VX154x Serial POD to the JTAG-PT interface of a target ECU.

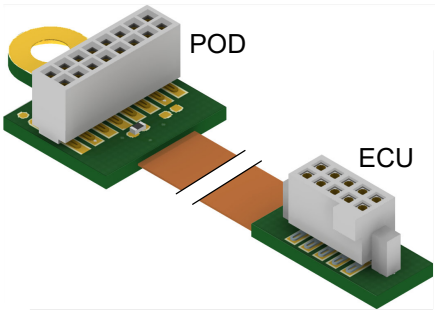


Figure 231: VX1927A JTAG-PT FlexPCB

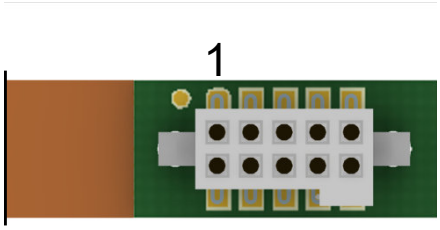


Figure 232: ECU connector with marking of pin 1

- **POD (Samtec)**
  16-pin socket for VX154x Serial POD with JTAG-PT interface option.
- **ECU**
  10-pin socket for µC interface (JTAG-PT).

Dimensions (L/W/H)	127 x 11 x 5.7 mm
Weight	2 g
Power supply	none (passive adapter)
Temperature range	-40 °C ... +115 °C
Part number	22308

WARNING!
All VX1927 FlexPCB adapters must be handled with care! They are not designed for frequent mating cycles! Avoid contact with sharp edges! Avoid sharp bending!

7.1.8.2 Target Interface Connectors

Port	Vendor	VX154x connector type
POD	Samtec	SFMC-108-02-L-D
ECU	Samtec	SFM-105-02-L-D-A

Note
Both connectors are non-locking and therefore must be fixed if a vibration-proof connection is required.

Note
The pin assignment of the POD connector can be found in section VX1543A Serial POD (Obsolete) on page 493.

ECU (JTAG)

The following table shows the pin assignment of the ECU connector of the VX1927A JTAG-PT FlexPCB for VX154x.

Pin	Signal	Group	Need	Dir	I/O	Details
1	GND	-	m	PWR	-	
2	TCK	JTAG	m	IN	3V3/(5V)	5 V support depends on VX154x Serial POD variant
3	nTRST/JCOMP	JTAG	(m)	IN	3V3/(5V)	Mandatory if available on target micro-controller; 5 V support depends on VX154x Serial POD variant
4	TDO	JTAG	m	OUT	3V3/(5V)	5 V support depends on VX154x Serial POD variant
5	TMS	JTAG	m	IN	3V3/(5V)	5 V support depends on VX154x Serial POD variant
6	TDI	JTAG	m	IN	3V3/(5V)	5 V support depends on VX154x Serial POD variant
7	WDGDIS	SYS	o	IN	3V3/(5V)	
8	VREF	SYS	o	OUT	3V3/(5V)	5 V support depends on VX154x Serial POD variant
9	nRSTOUT	SYS	o	OUT	3V3/(5V)	5 V support depends on VX154x Serial POD variant
10	nRESET	SYS	m	IN/OUT	3V3/(5V)	5 V support depends on VX154x Serial POD variant

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Pin direction	Description
IN	Input of the ECU
OUT	Output of the ECU
IN/OUT	Input/Output of the ECU
PWR	Power supply
DNC	Do not connect

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig** otherwise the POD may be damaged. This restriction does not apply to pins with the direction "NC" or listed as "5 V tolerant".

7.1.9 VX1927B DAP-PT FlexPCB for VX154x

7.1.9.1 General Information

The VX1927B flexible PCB adapter allows impedance-controlled connection of a VX154x Serial POD to the DAP-PT interface of a target ECU.

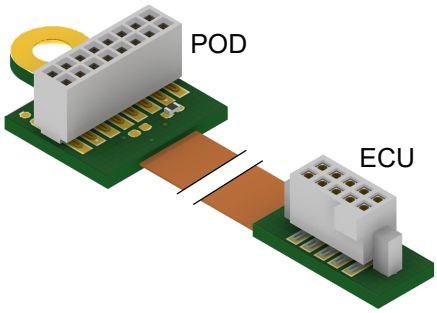


Figure 233: VX1927B DAP-PT FlexPCB

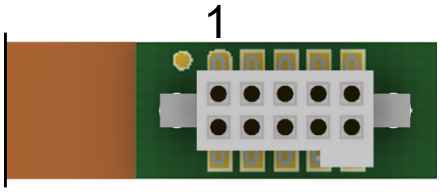


Figure 234: ECU connector with marking of pin 1

- **POD (Samtec)**
  16-Pin socket for VX154x Serial POD with DAP-PT interface option.
- **ECU**
  10-Pin socket for µC interface (DAP-PT).

Dimensions (L/W/H)	127 x 11 x 5.7 mm
Weight	2 g
Power supply	none (passive adapter)
Temperature range	-40 °C ... +115 °C
Part number	22309

WARNING!
All VX1927 FlexPCB adapters must be handled with care! They are not designed for frequent mating cycles! Avoid contact with sharp edges! Avoid sharp bending!

7.1.9.2 Target Interface Connectors

Port	Vendor	VX154x connector type
POD	Samtec	SFMC-108-02-L-D
ECU	Samtec	SFM-105-02-L-D-A

Note
Both connectors are non-locking and therefore must be fixed if a vibration-proof connection is required.

Note
The pin assignment of the POD connector can be found in section VX1543B Serial POD on page 502.

ECU (DAP-PT)

The following table shows the pin assignment of the ECU connector of the VX1927B DAP-PT FlexPCB for VX154x.

Pin	Signal	Group	Need	Dir	I/O	Details
1	GND	-	m	PWR	-	
2	DAP0	DAP	m	IN	3V3/(5V)	5 V support depends on VX154x Serial POD variant
3	DAPEN	DAP	m	IN	3V3/(5V)	5 V support depends on VX154x Serial POD variant
4	DAP2	DAP	(m)	IN/OUT	3V3/(5V)	Mandatory if using DAP Wide Mode; 5 V support depends on VX154x Serial POD variant
5	DAP1	DAP	m	IN/OUT	3V3/(5V)	5 V support depends on VX154x Serial POD variant
6	reserved	-	-	DNC	-	
7	WDGDIS	SYS	o	IN	3V3/(5V)	5 V support depends on VX154x Serial POD variant
8	VREF	SYS	o	OUT	3V3/(5V)	5 V support depends on VX154x Serial POD variant
9	nRSTOUT	SYS	o	OUT	3V3/(5V)	5 V support depends on VX154x Serial POD variant
10	nRESET	SYS	m	IN/OUT	3V3/(5V)	5 V support depends on VX154x Serial POD variant

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Pin direction	Description
IN	Input of the ECU
OUT	Output of the ECU
IN/OUT	Input/Output of the ECU
PWR	Power supply
DNC	Do not connect

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig** otherwise the POD may be damaged. This restriction does not apply to pins with the direction "NC" or listed as "5 V tolerant".

7.1.10 VX1927C DAP-IFX FlexPCB for VX154x (obsolete)

Reference
This accessory is obsolete. You can find its successor in section VX1927H DAP-IFX FlexPCB for VX154x on page 610.

7.1.10.1 General Information

The VX1927C flexible PCB adapter allows impedance-controlled connection of a VX154x Serial POD to the DAP-IFX interface of a target ECU.

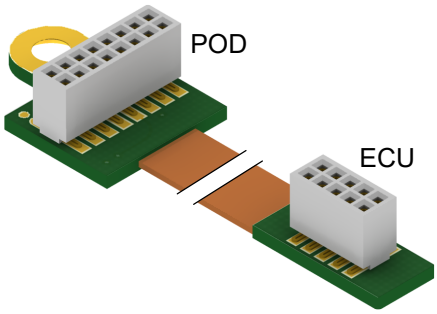


Figure 235: VX1927C DAP-IFX FlexPCB

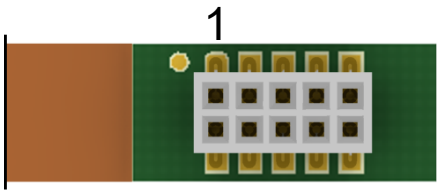


Figure 236: ECU connector with marking of pin 1

- **POD (Samtec)**
  16-Pin socket for VX154x Serial POD with DAP-IFX interface option.
- **ECU**
  10-Pin socket for µC interface (DAP-IFX).

Dimensions (L/W/H)	127 x 11 x 5.7 mm
Weight	2 g
Power supply	none (passive adapter)
Temperature range	-40 °C ... +115 °C
Part number	22310

WARNING!
All VX1927 FlexPCB adapters must be handled with care! They are not designed for frequent mating cycles! Avoid contact with sharp edges! Avoid sharp bending!

7.1.10.2 Target Interface Connectors

Port	Vendor	VX154x connector type
POD	Samtec	SFMC-108-02-L-D
ECU	Samtec	SFMC-105-02-L-D

Note
Both connectors are non-locking and therefore must be fixed if a vibration-proof connection is required.

Note
The pin assignment of the POD connector can be found in section VX1543B Serial POD on page 502.

ECU (DAP)

On the target microcontroller side a Samtec FTSH-105-01-L-DV-K-P is required. The pin assignment for Infineon DAP is shown in the following table. The pin direction is shown as seen from the target.

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	OUT	-	Strongly recommended*
2	DAP1	DAP	m	IN/OUT	3V3/(5V)	5 V support depends on VX154x Serial POD variant
3	GND	-	m	PWR	-	
4	DAP0	DAP	m	IN	3V3/(5V)	5 V support depends on VX154x Serial POD variant
5	GND	-	m	PWR	-	
6	DAP2	DAP	(m)	IN/OUT	3V3/(5V)	Mandatory if using DAP Wide Mode; 5 V support depends on VX154x Serial POD variant
7	GND	-	m	PWR	-	
8	DAPEN	DAP	m	IN	3V3/(5V)	4.75 kΩ pull-up to VREF on POD side; 5 V support depends on VX154x Serial POD variant
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	3V3/(5V)	5 V support depends on VX154x Serial POD variant

* for more Information see Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Pin direction	Description
IN	Input of the ECU
OUT	Output of the ECU
IN/OUT	Input/Output of the ECU
PWR	Power supply
DNC	Do not connect

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig** otherwise the POD may be damaged. This restriction does not apply to pins with the direction "NC" or listed as "5 V tolerant".

7.1.11 VX1927D Zipwire/HSCT FlexPCB for VX154x

7.1.11.1 General Information

The VX1927D flexible PCB adapter allows impedance-controlled connection of a VX154x Serial POD to the Zipwire/HSCT/RHSIF interface of a target ECU.

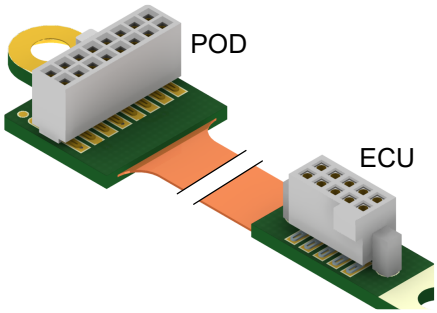


Figure 237: VX1927D Zipwire/HSCT FlexPCB

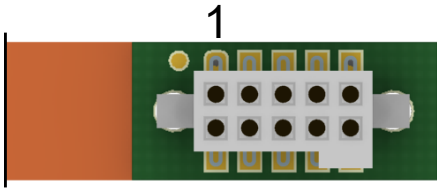


Figure 238: ECU connector with marking of pin 1

- **POD (Samtec)**
  16-Pin socket for VX154x Serial POD with Zipwire/HSCT interface option.
- **ECU**
  10-Pin socket for µC interface (Zipwire/HSCT).

Dimensions (L/W/H)	132 x 11 x 5.7 mm
Weight	2 g
Power supply	none (passive adapter)
Temperature range	-40 °C ... +115 °C
Part number	22433

WARNING!
All VX1927 FlexPCB adapters must be handled with care! They are not designed for frequent mating cycles! Avoid contact with sharp edges! Avoid sharp bending!

7.1.11.2 Target Interface Connectors

Port	Vendor	VX154x connector type
POD	Samtec	SFMC-108-02-L-D
ECU	Samtec	SFM-105-02-L-D-A

Note
Both connectors are non-locking and therefore must be fixed if a vibration-proof connection is required.

Note
The pin assignment of the POD connector can be found in section VX1543B Serial POD on page 502.

ECU

The following table shows the pin assignment of the ECU connector of the VX1927D Zipwire/HSCT/RHSIF FlexPCB for VX154x.

Pin	Signal	Group	Need	Dir	I/O	Details
1	nRESET	SYS	m	IN/OUT	3V3/5V	
2	reserved	-	-	DNC	-	
3	VREF	SYS	o	IN	3V3/5V	
4	reserved	-	-	DNC	-	
5	TXD_n	HSCT	m	OUT	LVDS	Differential HSCT data
6	TXD_p					
7	RXD_p	HSCT	m	IN	LVDS	Differential HSCT data
8	RXD_n					
9	REFCLK	HSCT	m	OUT	3V3/5V	
10	GND	-	m	PWR	-	

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig** otherwise the POD may be damaged. This restriction does not apply to pins with the direction "NC" or listed as "5 V tolerant".

7.1.12 VX1927E DAP-IFX FlexPCB for VX154x (obsolete)

Reference
This accessory is obsolete. You can find its successor in section VX1927J DAP-IFX FlexPCB for VX154x on page 613.

7.1.12.1 General Information

The VX1927E flexible PCB adapter allows impedance-controlled connection of a VX154x Serial POD to the DAP interface of a target ECU. While identical to the VX1927C, the VX1927E is 5 cm longer.

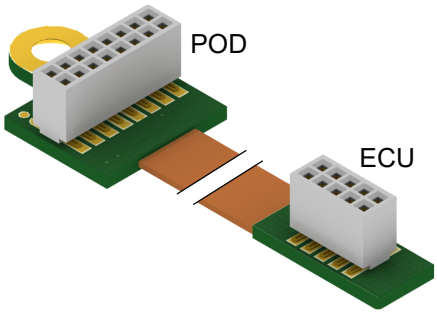


Figure 239: VX1927E DAP-IFX FlexPCB

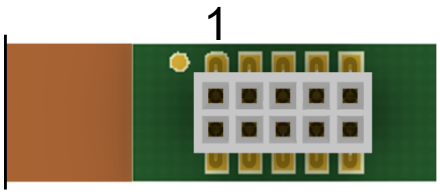


Figure 240: ECU connector with marking of pin 1

- **POD (Samtec)**
  16-Pin socket for VX154x Serial POD with DAP-IFX interface option.
- **ECU**
  10-Pin socket for µC interface (DAP-IFX).

Dimensions (L/W/H)	177 x 11 x 5.7mm
Weight	3 g
Power supply	none (passive adapter)
Temperature range	-40 °C ... +115 °C
Part number	22527

WARNING!
All VX1927 FlexPCB adapters must be handled with care! They are not designed for frequent mating cycles! Avoid contact with sharp edges! Avoid sharp bending!

7.1.12.2 Target Interface Connectors

Port	Venctor	VX154x connector type
POD	Samtec	SFMC-108-02-L-D
ECU	Samtec	SFMC-105-02-L-D

Note
Both connectors are non-locking and therefore must be fixed if a vibration-proof connection is required.

Note
The pin assignment of the POD connector can be found in section VX1543B Serial POD on page 502.

ECU (DAP-IFX)

On the target microcontroller side a Samtec FTSH-105-01-L-DV-K-P is required. The pin assignment for Infineon DAP is shown in the following table. The pin direction is shown as seen from the target.

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	OUT	-	Strongly recommended*
2	DAP1	DAP	m	IN/OUT	3V3/(5V)	5 V support depends on VX154x Serial POD variant
3	GND	-	m	PWR	-	
4	DAP0	DAP	m	IN	3V3/(5V)	5 V support depends on VX154x Serial POD variant
5	GND	-	m	PWR	-	
6	DAP2	DAP	(m)	IN/OUT	3V3/(5V)	Mandatory if using DAP Wide Mode; 5 V support depends on VX154x Serial POD variant
7	GND	-	m	PWR	-	
8	DAPEN	DAP	m	IN	3V3/(5V)	4.75 kΩ pull-up to VREF on POD side; 5 V support depends on VX154x Serial POD variant
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	3V3/(5V)	5 V support depends on VX154x Serial POD variant

* for more Information see Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Pin direction	Description
IN	Input of the ECU
OUT	Output of the ECU
IN/OUT	Input/Output of the ECU
PWR	Power supply
DNC	Do not connect

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig** otherwise the POD may be damaged. This restriction does not apply to pins with the direction "NC" or listed as "5 V tolerant".

7.1.13 VX1927H DAP-IFX FlexPCB for VX154x

7.1.13.1 General Information

The VX1927H flexible PCB adapter allows impedance-controlled connection of a VX154x Serial POD to the DAP-IFX interface of a target ECU.

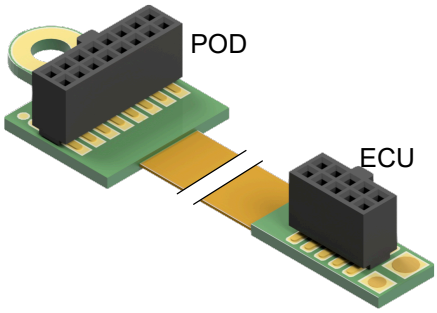


Figure 241: VX1927H DAP-IFX FlexPCB

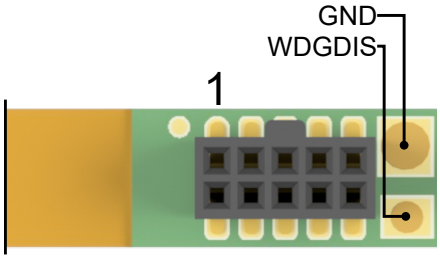


Figure 242: ECU connector with marking of pin 1

- **POD (Samtec)**
  16-Pin socket for VX154x Serial POD with DAP-IFX interface option.
- **ECU**
  10-Pin socket for µC interface (DAP-IFX).

Dimensions (L/W/H)	127 x 11 x 5.7 mm
Weight	2 g
Power supply	none (passive adapter)
Temperature range	-40 °C ... +115 °C
Part number	22636

WARNING!
All VX1927 FlexPCB adapters must be handled with care! They are not designed for frequent mating cycles! Avoid contact with sharp edges! Avoid sharp bending!

7.1.13.2 Target Interface Connectors

Port	Vendor	VX154x connector type
POD	Samtec	ASP-188638-02
ECU	Samtec	ASP-188638-01

Note
Both connectors are non-locking and therefore must be fixed if a vibration-proof connection is required.

Note
The pin assignment of the POD connector can be found in section VX1543B Serial POD on page 502.

ECU (DAP)

On the target microcontroller side a Samtec FTSH-105-01-L-DV-K-P is required. The pin assignment for Infineon DAP is shown in the following table. The pin direction is shown as seen from the target.

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	OUT	-	Strongly recommended*
2	DAP1	DAP	m	IN/OUT	3V3/(5V)	5 V support depends on VX154x Serial POD variant
3	GND	-	m	PWR	-	
4	DAP0	DAP	m	IN	3V3/(5V)	5 V support depends on VX154x Serial POD variant
5	GND	-	m	PWR	-	
6	DAP2	DAP	(m)	IN/OUT	3V3/(5V)	Mandatory if using DAP Wide Mode; 5 V support depends on VX154x Serial POD variant
7	GND	-	m	PWR	-	
8	DAPEN	DAP	m	IN	3V3/(5V)	4.75 kΩ pull-up to VREF on POD side; 5 V support depends on VX154x Serial POD variant
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	3V3/(5V)	5 V support depends on VX154x Serial POD variant

* for more Information see Hardware Integration Guide

WDGDIS

This solderpad allows a connection to the watchdog disable pin of the target. The pin direction is shown as seen from the target.

Pin	Signal	Group	Need	Dir	I/O	Details
Pad	WDGDIS	SYS	o	IN	3V3/(5V)	Allows disabling an external watchdog for flash programming*; 5 V support depends on VX154x Serial POD variant

* for more Information see Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional

Legend

Column	Description
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Pin direction	Description
IN	Input of the ECU
OUT	Output of the ECU
IN/OUT	Input/Output of the ECU
PWR	Power supply
DNC	Do not connect

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig** otherwise the POD may be damaged. This restriction does not apply to pins with the direction "NC" or listed as "5 V tolerant".

7.1.14 VX1927J DAP-IFX FlexPCB for VX154x

7.1.14.1 General Information

The VX1927J flexible PCB adapter allows impedance-controlled connection of a VX154x Serial POD to the DAP interface of a target ECU. While identical to the VX1927H, the VX1927J is 5 cm longer.

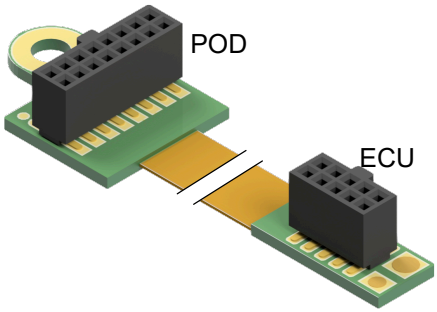


Figure 243: VX1927H DAP-IFX FlexPCB

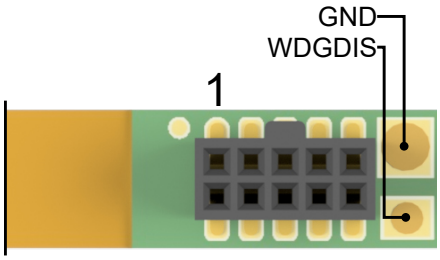


Figure 244: ECU connector with marking of pin 1

- **POD (Samtec)**
  16-Pin socket for VX154x Serial POD with DAP-IFX interface option.
- **ECU**
  10-Pin socket for µC interface (DAP-IFX).

Dimensions (L/W/H)	177 x 11 x 5.7mm
Weight	3 g
Power supply	none (passive adapter)
Temperature range	-40 °C ... +115 °C
Part number	22527

WARNING!
All VX1927 FlexPCB adapters must be handled with care! They are not designed for frequent mating cycles! Avoid contact with sharp edges! Avoid sharp bending!

7.1.14.2 Target Interface Connectors

Port	Venctor	VX154x connector type
POD	Samtec	ASP-188638-02
ECU	Samtec	ASP-188638-01

Note
Both connectors are non-locking and therefore must be fixed if a vibration-proof connection is required.

Note
The pin assignment of the POD connector can be found in section VX1543B Serial POD on page 502.

ECU (DAP-IFX)

On the target microcontroller side a Samtec FTSH-105-01-L-DV-K-P is required. The pin assignment for Infineon DAP is shown in the following table. The pin direction is shown as seen from the target.

Pin	Signal	Group	Need	Dir	I/O	Details
1	VREF	SYS	o	OUT	-	Strongly recommended*
2	DAP1	DAP	m	IN/OUT	3V3/(5V)	5 V support depends on VX154x Serial POD variant
3	GND	-	m	PWR	-	
4	DAP0	DAP	m	IN	3V3/(5V)	5 V support depends on VX154x Serial POD variant
5	GND	-	m	PWR	-	
6	DAP2	DAP	(m)	IN/OUT	3V3/(5V)	Mandatory if using DAP Wide Mode; 5 V support depends on VX154x Serial POD variant
7	GND	-	m	PWR	-	
8	DAPEN	DAP	m	IN	3V3/(5V)	4.75 kΩ pull-up to VREF on POD side; 5 V support depends on VX154x Serial POD variant
9	GND	-	m	PWR	-	
10	nRESET	SYS	m	IN/OUT	3V3/(5V)	5 V support depends on VX154x Serial POD variant

* for more Information see Hardware Integration Guide

WDGDIS

This solderpad allows a connection to the watchdog disable pin of the target. The pin direction is shown as seen from the target.

Pin	Signal	Group	Need	Dir	I/O	Details
Pad	WDGDIS	SYS	o	IN	3V3/(5V)	Allows disabling an external watchdog for flash programming*; 5 V support depends on VX154x Serial POD variant

* for more Information see Hardware Integration Guide

Legend

Column	Description
Pin	Pin number
Signal	Signal name
Group	Interface group
Need	Need for the signal within the Interface group (see table below for details) - please note the group itself might be optional

Legend

Column	Description
Dir	Pin direction (see table below for details)
I/O	I/O standard (see table below for details)

Pin direction	Description
IN	Input of the ECU
OUT	Output of the ECU
IN/OUT	Input/Output of the ECU
PWR	Power supply
DNC	Do not connect

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

I/O Standard	Description
LVDS	Low voltage differential signalling
1.5-V PCML	Pseudo common mode logic (differential signalling)
1V8	compatible with 1.8 V TTL
3V3	compatible with 3.3 V LVTTL and LVCMOS33
5V	compatible with 5 V TTL

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig** otherwise the POD may be damaged. This restriction does not apply to pins with the direction "NC" or listed as "5 V tolerant".

7.1.15 VX1903.01A DAP Flex-PCB for XPOD

VX1903.01A

The VX1903.01A is a rigid flex target cable for the VX1621A with a length of 12 cm. On the target side it is equipped with a Samtec SFM-105-02-L-D-A connector.

Figure 245: VX1903.01A DAP Flex-PCB for XPOD

The following table shows the pinout of the compatible ECU connector from type Samtec FTSH-105-01-L-DV-K-P. Pin direction is shown from target side.

Target (DAP)			Target (DAP)		
Pin	Assignment	Direction	Pin	Assignment	Direction
1	VREF	OUT	2	DAP0	IN/OUT
3	GND	PWR	4	DAP1	IN
5	GND	PRW	6	DAP2	IN/OUT
7	GND	PWR	8	DAPEN	IN
9	GND	PRW	10	mReset	IN/OUT

Pin direction

PWR	Power supply	IN	Input of VX1621
-	Reserved (input) pin	OUT	Output of VX1621
NC	Not connected	IN/OUT	Input/Output of VX1621
DNC	**Do not connect**		

Dimensions (L/W/H)	
Weight	
Power supply	
Temperature range	
Part number	

7.1.16 VX1903.01B DAP Flex-PCB for XPOD

VX1903.01B

The VX1903.01B is a rigid flex target cable for the VX1621A with a length of 12 cm. On the target side it is equipped with a Samtec SFM-105-02-L-D-A connector.

Figure 246: VX1903.01B DAP Flex-PCB for XPOD

VX1903.01B

The VX1903.01B is a rigid flex target cable for the VX1621 with a length of 12 cm. On the target side it is equipped with a Samtec SFM-105-02-L-D-A connector.

The following table shows the pinout of the compatible ECU connector from type Samtec TFM-105-02-L-DH. Pin direction is shown from target side.

Target (DAP)		
Pin	Assignment	Direction
1	GND	PWR
3	DAPEN	I
5	DAP1	I/O
7	WDGDIS	I
9	nRSTOUT	O

Target (DAP)		
Pin	Assignment	Direction
2	DAP0	I
4	DAP2	I/O
6	reserved	-
8	VREF	O
10	nPORST	I/O

Pin direction

PWR	Power supply
-	Reserved (input) pin
NC	Not connected
DNC	**Do not connect**

IN	Input of VX1621
OUT	Output of VX1621
IN/OUT	Input/Output of VX1621

Dimensions (L/W/H)	
Weight	
Power supply	
Temperature range	
Part number	

7.2 EEK Heads

7.2.1 VX1902.02A EEK Head Aurora (Infineon)

7.2.1.1 General Information

The VX1902.02A EEK Head Aurora (Infineon) is a frontend adapter required when using an Evalboard Extension Kit for a VX145x HSSL POD. It is designed to interface to Infineon Evalboards with a 22-pin header for Aurora Data Trace. It provides an Aurora clock frequency of 100 MHz towards the target. While the VX1902.02A has a housing, the VX1902.02B is a PCB-only variant.

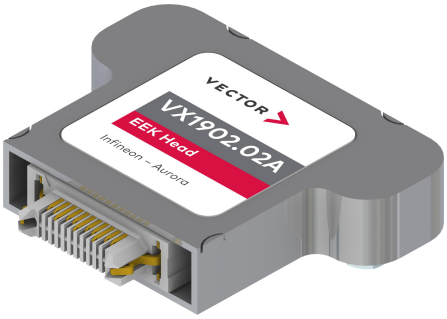


Figure 247: VX1902.02A top view

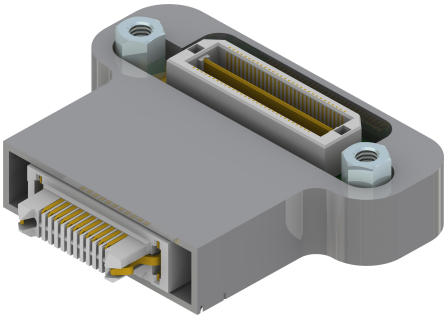


Figure 248: VX1902.02A bottom view

- **Flex-PCB**
  Connection to the Flex-PCB of the Evalboard Extension Kit.
- **Evalboard**
  22-pin connector for Aurora interface (Infineon).

Dimensions (L/W/H)	37.2 x 34 x 9 mm
Weight	8 g
Power supply	1.5 W
Temperature range	-40 °C ... +85 °C
Part number	22346

7.2.1.2 Target Interface Connectors

Port	Vendor	VX145x connector type
Flex-PCB	Samtec	QTH-030-01-L-D-A
ECU	Samtec	ASP-137971-02

The following table shows the pin assignment of the adapter:

ECU, Infineon (Aurora/DAP)			ECU, Infineon (Aurora/DAP)		
Pin	Assignment	Direction	Pin	Assignment	Direction
1	AUR TX0p	IN	2	VREF	IN
3	AUR TX0n	IN	4	DAP0	IN
5	GND	PWR	6	DAP1	IN/OUT
7	AUR TX1p	IN	8	reserved	-
9	AUR TX1n	IN	10	DAP2	IN/OUT
11	GND	PWR	12	DAPEN	IN
13	AUR TX2p	IN	14	AUR CLKp	IN
15	AUR TX2n	IN	16	AUR CLKn	IN
17	GND	PWR	18	reserved	-
19	AUR TX3p	IN	20	AUR CRC ERR	IN
21	AUR TX3n	IN	22	nRESET (5 V tolerant)	IN/OUT

Pin direction

PWR	Power supply	IN	Input of µC
-	Reserved (input) pin	OUT	Output of µC
NC	Not connected	IN/OUT	Input/Output of µC
DNC	**Do not connect**		

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig** otherwise the POD may be damaged. This restriction does not apply to pins with the direction "NC" or listed as "5 V tolerant".

7.2.2 VX1902.02B EEK Head AUR PCB (Infineon)

7.2.2.1 General Information

The VX1902.02B EEK Head Aurora (Infineon) is a frontend adapter required when using an Evalboard Extension Kit for a VX145x HSSL POD. It is designed to interface to Infineon Evalboards with 22-pin header for Aurora Data Trace. It provides an Aurora clock frequency of 100 MHz towards the target. While the VX1902.02A has a housing, the VX1902.02B is a PCB-only variant.

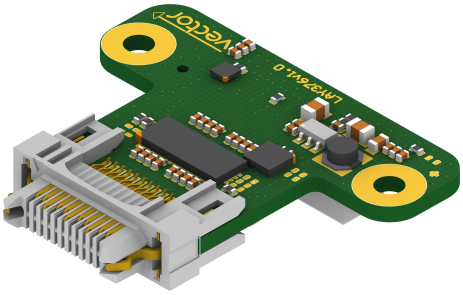


Figure 249: VX1902.02B top view

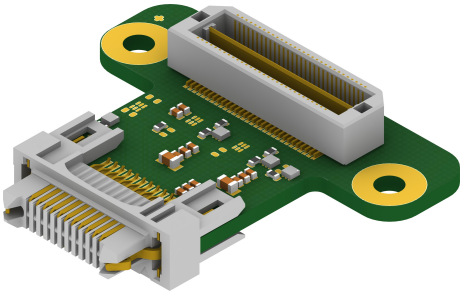


Figure 250: VX1902.02B bottom view

- **Flex-PCB**
  Connection to the Flex-PCB of the Evalboard Extension Kit.
- **Evalboard**
  22-pin connector for Aurora interface (Infineon).

Dimensions (L/W/H)	34.7 x 31 x 7 mm
Weight	5 g
Power supply	1.5 W
Temperature range	-40 °C ... +85 °C
Part number	120094

7.2.2.2 Target Interface Connectors

Port	Vendor	VX145x connector type
Flex-PCB	Samtec	QTH-030-01-L-D-A
ECU	Samtec	ASP-137971-02

The following table shows the pin assignment of the adapter:

ECU, Infineon (Aurora/DAP)			ECU, Infineon (Aurora/DAP)		
Pin	Assignment	Direction	Pin	Assignment	Direction
1	AUR TX0p	IN	2	VREF	IN
3	AUR TX0n	IN	4	DAP0	IN
5	GND	PWR	6	DAP1	IN/OUT
7	AUR TX1p	IN	8	reserved	-
9	AUR TX1n	IN	10	DAP2	IN/OUT
11	GND	PWR	12	DAPEN	IN
13	AUR TX2p	IN	14	AUR CLKp	IN
15	AUR TX2n	IN	16	AUR CLKn	IN
17	GND	PWR	18	reserved	-
19	AUR TX3p	IN	20	AUR CRC ERR	IN
21	AUR TX3n	IN	22	nRESET (5 V tolerant)	IN/OUT

Pin direction

PWR	Power supply	IN	Input of µC
-	Reserved (input) pin	OUT	Output of µC
NC	Not connected	IN/OUT	Input/Output of µC
DNC	**Do not connect**		

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig** otherwise the POD may be damaged. This restriction does not apply to pins with the direction "NC" or listed as "5 V tolerant".

7.2.3 VX1902.03A EEK Head Aurora (NXP/Renesas)

7.2.3.1 General Information

The VX1902.03A EEK Head Aurora (NXP/Renesas) is a frontend adapter required when using an Evalboard Extension Kit for a VX145x HSSL POD. It is designed to interface to NXP and Renesas Evalboards with 34-pin header for Nexus Aurora Data Trace. It provides an Aurora clock frequency from 0.2 MHz up to 1.5 GHz towards the target. While the VX1902.03A has a housing, the VX1902.03B is a PCB-only variant.

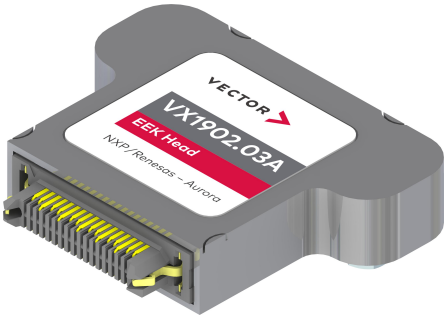


Figure 251: VX1902.03A top view

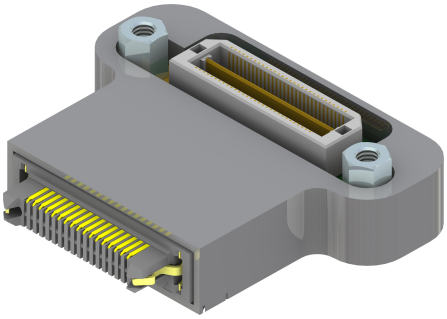


Figure 252: VX1902.03A bottom view

- **Flex-PCB**
  Connection to the Flex-PCB of the Evalboard Extension Kit.
- **Evalboard**
  38-pin connector for Aurora interface (NXP/Renesas).

Dimensions (L/W/H)	37.2 x 34 x 9 mm
Weight	8 g
Power supply	1.5 W
Temperature range	-40 °C ... +85 °C
Part number	180095

7.2.3.2 Target Interface Connectors

Port	Vendor	VX145x connector type
Flex-PCB	Samtec	QTH-030-01-L-D-A
ECU	Samtec	ASP-137974-02

The following table shows the pin assignment of the adapter:

ECU, NXP/Renesas (Aurora/JTAG)			ECU, NXP/Renesas (Aurora/JTAG)		
Pin	Assignment	Direction	Pin	Assignment	Direction
1	AUR TX0p	IN	2	VREF	IN
3	AUR TX0n	IN	4	TCK	OUT
5	GND	PWR	6	TMS	OUT
7	AUR TX1p	IN	8	TDI	OUT
9	AUR TX1n	IN	10	TDO	IN
11	GND	PWR	12	JCOMP	OUT
13	AUR TX2p	IN	14	nEVTI1	-
15	AUR TX2n	IN	16	nEVTI0	-
17	GND	PWR	18	nEVTO0	-
19	AUR TX3p	IN	20	RSTOUT	IN
21	AUR TX3n	IN	22	nRESET (5 V tolerant)	IN/OUT
23	GND	PWR	24	GND	PWR
25	-	NC	26	AUR CLKp	OUT
27	-	NC	28	AUR CLKn	OUT
29	GND	PWR	30	GND	PWR
31	-	NC	32	nRDY	-
33	-	NC	34	WDT	-

Pin direction

PWR	Power supply	IN	Input of µC
-	Reserved (input) pin	OUT	Output of µC
NC	Not connected	IN/OUT	Input/Output of µC
DNC	**Do not connect**		

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig** otherwise the POD may be damaged. This restriction does not apply to pins with the direction "NC" or listed as "5 V tolerant".

7.2.4 VX1902.03B EEK Head AUR PCB (NXP/Renesas)

7.2.4.1 General Information

The VX1902.03B EEK Head Aurora (NXP/Renesas) is a frontend adapter required when using an Evalboard Extension Kit for a VX145x HSSL POD. It is designed to interface to NXP and Renesas Evalboards with 34-pin Nexus Aurora Data Trace. It provides an Aurora clock frequency from 0.2 MHz up to 1.5 GHz towards the target. While the VX1902.03A has a housing, the VX1902.03B is a PCB-only variant.

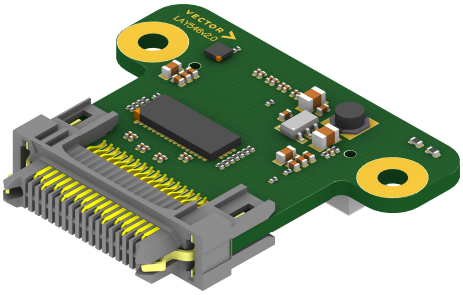


Figure 253: VX1902.03B top view

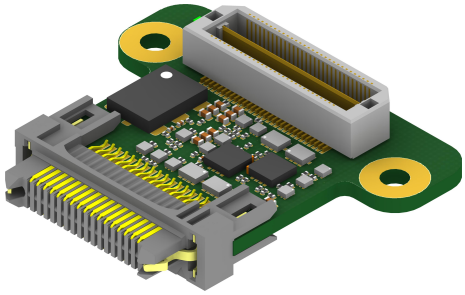


Figure 254: VX1902.03B bottom view

- **Flex-PCB**
  Connection to the Flex-PCB of the Evalboard Extension Kit.
- **Evalboard**
  38-pin connector for Aurora interface (NXP/Renesas).

Dimensions (L/W/H)	34.7 x 31 x 7 mm
Weight	5 g
Power supply	1.5 W
Temperature range	-40 °C ... +85 °C
Part number	120231

7.2.4.2 Target Interface Connectors

Port	Vendor	VX145x connector type
Flex-PCB	Samtec	QTH-030-01-L-D-A
ECU	Samtec	ASP-137974-02

The following table shows the pin assignment of the adapter:

ECU, NXP/Renesas (Aurora/JTAG)			ECU, NXP/Renesas (Aurora/JTAG)		
Pin	Assignment	Direction	Pin	Assignment	Direction
1	AUR TX0p	IN	2	VREF	IN
3	AUR TX0n	IN	4	TCK	OUT
5	GND	PWR	6	TMS	OUT
7	AUR TX1p	IN	8	TDI	OUT
9	AUR TX1n	IN	10	TDO	IN
11	GND	PWR	12	JCOMP	OUT
13	AUR TX2p	IN	14	nEVTI1	-
15	AUR TX2n	IN	16	nEVTI0	-
17	GND	PWR	18	nEVTO0	-
19	AUR TX3p	IN	20	RSTOUT	IN
21	AUR TX3n	IN	22	nRESET (5 V tolerant)	IN/OUT
23	GND	PWR	24	GND	PWR
25	-	NC	26	AUR CLKp	OUT
27	-	NC	28	AUR CLKn	OUT
29	GND	PWR	30	GND	PWR
31	-	NC	32	nRDY	-
33	-	NC	34	WDT	-

Pin direction

PWR	Power supply	IN	Input of µC
-	Reserved (input) pin	OUT	Output of µC
NC	Not connected	IN/OUT	Input/Output of µC
DNC	**Do not connect**		

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig** otherwise the POD may be damaged. This restriction does not apply to pins with the direction "NC" or listed as "5 V tolerant".

7.2.5 VX1902.04A EEK Head Aurora (ARM)

7.2.5.1 General Information

The VX1902.04A EEK Head Aurora (ARM) is a frontend adapter required when using an Evalboard Extension Kit for a VX145x HSSL POD. It is designed to interface to ARM Evalboards with 40-pin header for Aurora Data Trace. It provides an Aurora clock frequency from 0.2 MHz up to 1.5 GHz, the ARM SWD Interface and an additional JTAG connector towards the Evalboard. While the VX1902.04A has a housing, the VX1902.04B is a PCB-only variant.

Figure 255: VX1902.04A top view

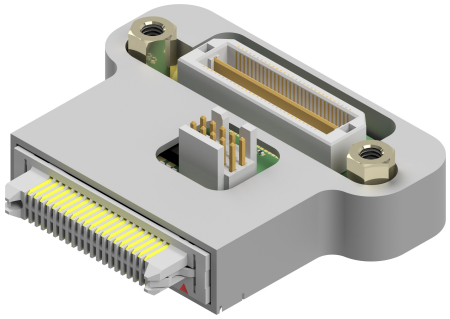


Figure 256: VX1902.04A bottom view

- **Flex-PCB**
  Connection to the Flex-PCB of the Evalboard Extension Kit.
- **Evalboard**
  40-pin connector for Nexus Aurora and SWD interfaces (ARM).
- **JTAG**
  JTAG interface connector.

Dimensions (L/W/H)	37.2 x 34 x 9 mm
Weight	8 g
Power supply	1.5 W
Temperature range	-40 °C ... +85 °C
Part number	180031

7.2.5.2 Target Interface Connectors

Port	Vendor	VX145x connector type
Flex-PCB	Samtec	QTH-030-01-L-D-A
ECU	Samtec	ASP-133811-03
JTAG	Samtec	FTSH-105-01-L-DV-K

The following table shows the pin assignment of the adapter:

ECU, ARM (Aurora/JTAG/SWD)			ECU, ARM (Aurora/JTAG/SWD)		
Pin	Assignment	Direction	Pin	Assignment	Direction
1	-	NC	2	VREF	PWR
3	-	NC	4	TCK/SWCLK	OUT
5	GND	PWR	6	GND	PWR
7	AUR TX2p	IN	8	TMS/SWDIO	IN/OUT
9	AUR TX2n	IN	10	nTRST	IN
11	GND	PWR	12	GND	PWR
13	AUR TX0p	IN	14	TDI	IN
15	AUR TX0n	IN	16	TDO/SWO	OUT
17	GND	PWR	18	GND	PWR
19	AUR CLKp	IN	20	nRESET (5 V tolerant)	IN/OUT
21	AUR CLKn	IN	22	DBG REQ	IN
23	GND	PWR	24	GND	PWR
25	AUR TX1p	IN	26	DBG ACK	OUT
27	AUR TX1n	IN	28	-	NC
29	GND	PWR	30	GND	PWR
31	AUR TX3p	NC	32	TRIG IN	IN
33	AUR TX3n	NC	34	TRIG OUT	OUT
35	GND	PWR	36	GND	PWR
37	-	NC	38	GND	PWR
39	-	NC	40	GND	PWR

Pin direction

PWR	Power supply	IN	Input of µC
-	Reserved (input) pin	OUT	Output of µC
NC	Not connected	IN/OUT	Input/Output of µC
DNC	**Do not connect**		

WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig** otherwise the POD may be damaged. This restriction does not apply to pins with the direction "NC" or listed as "5 V tolerant".

7.2.6 VX1902.04B EEK Head AUR PCB (ARM)

7.2.6.1 General Information

The VX1902.04B EEK Head Aurora (ARM) is a frontend adapter required when using an Evalboard Extension Kit for a VX145x HSSL POD. It is designed to interface to ARM Evalboards with 40-pin header for Aurora Data Trace. It provides an Aurora clock frequency from 0.2 MHz up to 1.5 GHz, the ARM SWD Interface and an additional JTAG connector towards the Evalboard. While the VX1902.04A has a housing, the VX1902.04B is a PCB-only variant.

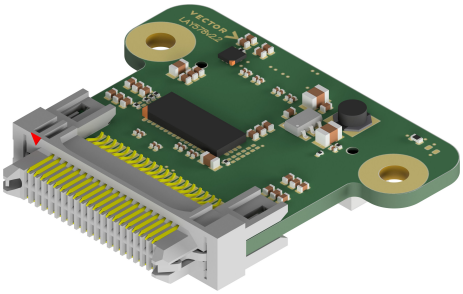


Figure 257: VX1902.04B top view

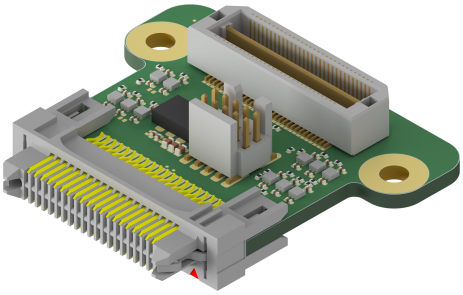


Figure 258: VX1902.04B bottom view

- **Flex-PCB**
  Connection to the Flex-PCB of the Evalboard Extension Kit.
- **Evalboard**
  40-pin connector for Nexus Aurora and SWD interfaces (ARM).
- **JTAG**
  JTAG interface connector.

Dimensions (L/W/H)	34.7 x 31 x 7 mm
Weight	5 g
Power supply	1.5 W
Temperature range	-40 °C ... +85 °C
Part number	120243

7.2.6.2 Target Interface Connectors

Port	Vendor	VX145x connector type
Flex-PCB	Samtec	QTH-030-01-L-D-A
ECU	Samtec	ASP-133811-03
JTAG	Samtec	FTSH-105-01-L-DV-K

The following table shows the pin assignment of the adapter:

ECU, ARM (Aurora/JTAG/SWD)			ECU, ARM (Aurora/JTAG/SWD)		
Pin	Assignment	Direction	Pin	Assignment	Direction
1	-	NC	2	VREF	PWR
3	-	NC	4	TCK/SWCLK	OUT
5	GND	PWR	6	GND	PWR
7	AUR TX2p	IN	8	TMS/SWDIO	IN/OUT
9	AUR TX2n	IN	10	nTRST	IN
11	GND	PWR	12	GND	PWR
13	AUR TX0p	IN	14	TDI	IN
15	AUR TX0n	IN	16	TDO/SWO	OUT
17	GND	PWR	18	GND	PWR
19	AUR CLKp	IN	20	nRESET (5 V tolerant)	IN/OUT
21	AUR CLKn	IN	22	DBG REQ	IN
23	GND	PWR	24	GND	PWR
25	AUR TX1p	IN	26	DBG ACK	OUT
27	AUR TX1n	IN	28	-	NC
29	GND	PWR	30	GND	PWR
31	AUR TX3p	NC	32	TRIG IN	IN
33	AUR TX3n	NC	34	TRIG OUT	OUT
35	GND	PWR	36	GND	PWR
37	-	NC	38	GND	PWR
39	-	NC	40	GND	PWR

Pin direction

PWR	Power supply	IN	Input of µC
-	Reserved (input) pin	OUT	Output of µC
NC	Not connected	IN/OUT	Input/Output of µC
DNC	**Do not connect**		

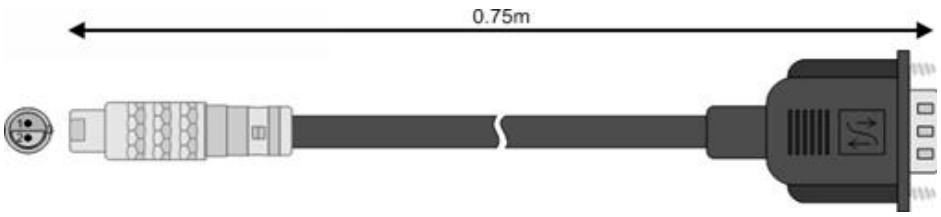
WARNING!
The voltage on each individual pin must not exceed the Target Interface Voltage set in **VXconfig** otherwise the POD may be damaged. This restriction does not apply to pins with the direction "NC" or listed as "5 V tolerant".

7.3 CAN Cables

7.3.1 VX1361C CAN Cable LEMO/D-SUB 0.75m

CAN cable

Description	CAN cable
Length	75 cm
Connectors	1x LEMO 1x D-SUB9
Part number	22332



LEMO	
Pin	**Assignment**
1	CAN High
2	CAN Low
3	Shield

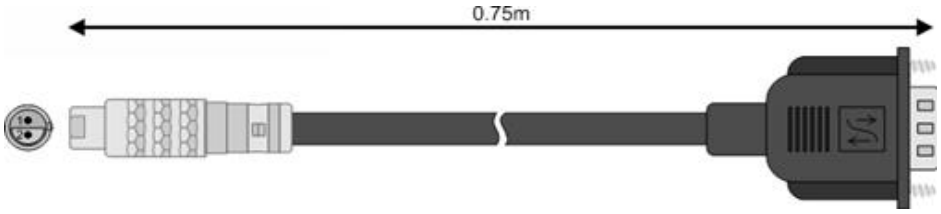
D-SUB9	
Pin	**Assignment**
1	Not connected
2	CAN Low
3	GND
4	Not connected
5	Shield
6	Not connected
7	CAN High
8	Not connected
9	Not connected

Note
In comparison to the VX1361D the D-SUB9 connector of the VX1361C has fixing screws.

7.3.2 VX1361D CAN Cable LEMO/D-SUB 0.75m

CAN cable

Description	CAN cable
Length	75 cm
Connectors	1x LEMO 1x D-SUB9
Part number	22225



LEMO	
Pin	**Assignment**
1	CAN High
2	CAN Low
3	Shield

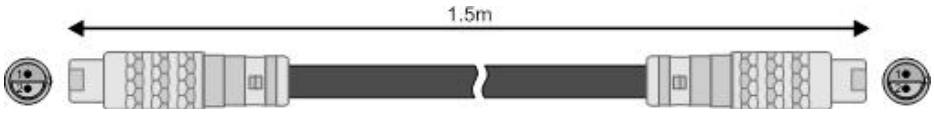
D-SUB9	
Pin	**Assignment**
1	Not connected
2	CAN Low
3	GND
4	Not connected
5	Shield
6	Not connected
7	CAN High
8	Not connected
9	Not connected

Note
In comparison to the VX1361C the D-SUB9 connector of the VX1361D has fixing bolts.

7.3.3 VX1362A CAN Cable LEMO 1.5m

CAN cable

Description	CAN cable
Length	1.5 m
Connectors	2x LEMO
Part number	22154



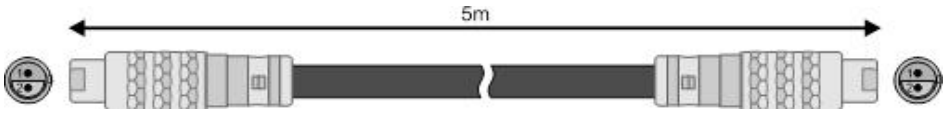
LEMO	
Pin	**Assignment**
1	CAN High
2	CAN Low
3	Shield

LEMO	
Pin	**Assignment**
1	CAN High
2	CAN Low
3	Shield

7.3.4 VX1365A CAN Cable LEMO 5m

CAN cable

Description	CAN cable
Length	5 m
Connectors	2x LEMO
Part number	22155



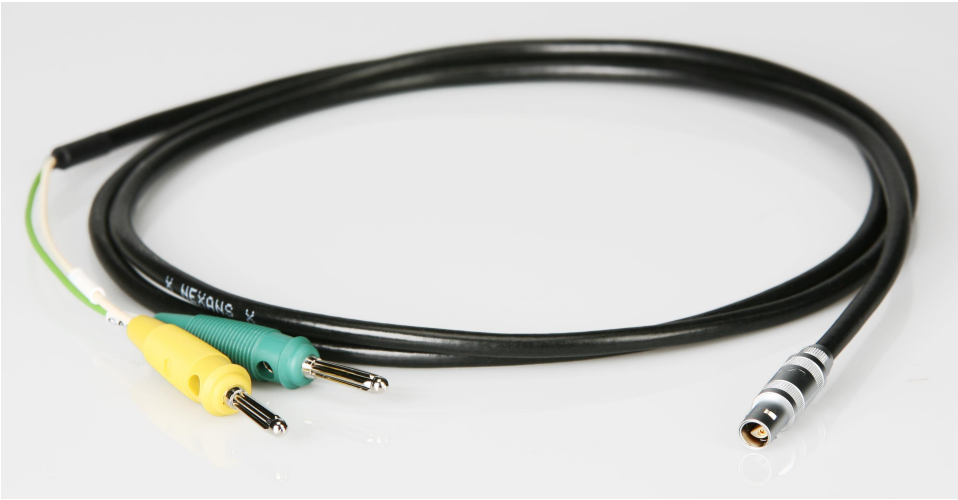
LEMO	
Pin	**Assignment**
1	CAN High
2	CAN Low
3	Shield

LEMO	
Pin	**Assignment**
1	CAN High
2	CAN Low
3	Shield

7.3.5 VX1362B CAN Cable LEMO/Banana Plugs 1.5m

CAN cable

Description	CAN cable
Length	1.5 m
Connectors	1x LEMO connector 2x banana plugs (4 mm)
Part number	22258



LEMO	
Pin	**Assignment**
1	CAN High
2	CAN Low
3	Shield

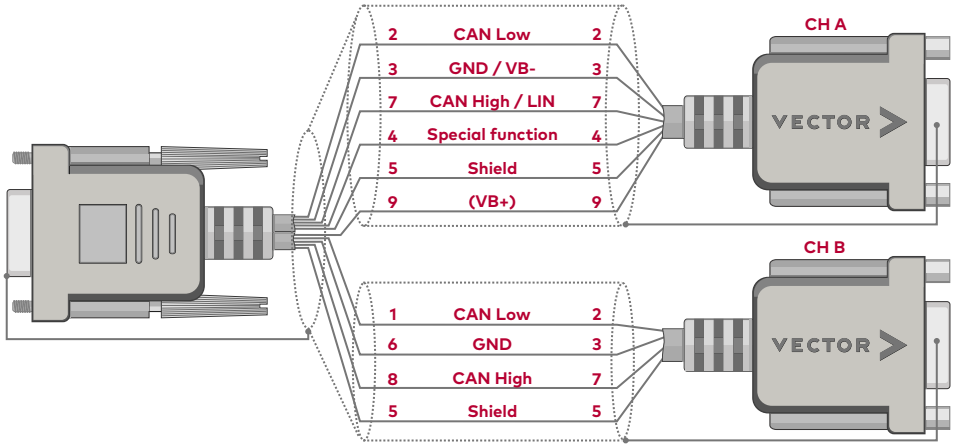
Banana Plugs	
Color	**Assignment**
Yellow	CAN High
Green	CAN Low
-	-

7.3.6 CANcable 2Y

Y cable

Description	Y cable for Vector CAN/LIN interfaces with D-SUB9 double assignment (VN1600 interface family, VN8970 etc.). Splits the double assignment into two separate D-SUB9 connectors (CH A and CH B). Examples: ► **VN1610** - CH1/2 → Channel 1 (CH A) and 2 (CH B) ► **VN1630** - CH1/3 → Channel 1 (CH A) and 3 (CH B) - CH2/4 → Channel 2 (CH A) and 4 (CH B) ► **VN8970** - CH1/5 → Channel 1 (CH A) and 5 (CH B) - CH2/6 → Channel 2 (CH A) and 6 (CH B) - CH3/7 → Channel 3 (CH A) and 7 (CH B) - CH4/8 → Channel 4 (CH A) and 8 (CH B) The pin assignments of the D-SUB9 connectors CH A and CH B depend on the used bus transceivers inside the interface .
Length	0.3 m
Connectors	1x female D-SUB9 connector 2x male D-SUB9 connectors
Temp. range	Fixed installation: -30 °C ...+80 °C Flexible use: -10 °C ...+80 °C
Min. bending radius	Fixed installation: 27.5 mm Flexible use: 55.0 mm
Part number	05075

Setup



7.4 FAKRA Cables

7.4.1 VX1382D/E/F/G FAKRA Cables

Description	The VX1382 FAKRA Cable for the VX1000 system is designed to resist the environmental stress in a vehicle setup. These cables are waterproof.
Details	▶ Temperature range: -40 °C ... +105 °C ▶ Mating cycles: max. 5 ▶ Minimum bending radius: 5.2 cm ▶ Vibration-proof due to latching mechanism in the connectors ▶ Waterproof ▶ While connected, compliant to protection class IPx6 ▶ Cable type: Leoni DACAR 302 85120380D
Part number	▶ **VX1382D FAKRA Cable 2m** 22562 ▶ **VX1382E FAKRA Cable 5m** 22563 ▶ **VX1382F FAKRA Cable 8m** 22564

WARNING!
The maximum mating cycles of waterproof FAKRA connectors must be considered. For increasing the number of mating cycles, VX1382H FAKRA Extension 0.1m can be used.

WARNING!
The cable must not be bent with a radius of less than 5.2 cm!
Only in the case, when the cable is permanently installed in a fixed position, a bending radius of down to 1.8 cm is possible.

Note
Technical drawings can be found at vector.com/vx1000-drawings.

7.4.2 VX1382H FAKRA Extention 0.1m Cable

Description	The VX1382H FAKRA Extention 0.1 Cable is designed to resist the environmental stress in a vehicle setup. It is intended for conserving connectors on ECUs, measurement tools or cables which need to be plugged frequently. This cable is waterproof.
Details	► Temperature range: -40 °C ... +105 °C ► Mating cycles: max. 5 guaranteed by connector manufacturer ► Minimum bending radius: 5.2 cm ► Vibration-proof due to latching mechanism in the connectors ► Waterproof ► While connected, compliant to protection class IPx6 ► Cable type: Leoni DACAR 302 85120380D
Part number	► **VX1382H FAKRA Extension 0.1m** 22572

WARNING!
The maximum mating cycles of waterproof FAKRA connectors must be considered. For increasing the number of mating cycles,
VX1382H FAKRA Extension 0.1m can be used.

WARNING!
The cable must not be bent with a radius of less than 5.2 cm!
Only in the case, when the cable is permanently installed in a fixed position, a bending radius of down to 1.8 cm is possible.

Note
Technical drawings can be found at vector.com/vx1000-drawings.

7.4.3 VX1382M.01 FAKRA Panelmount F0N100N0F (U.FL)

Description	The VX1382M.01 FAKRA Panelmount F0N100N0F is a coax adapter cable for small PODs with U.FL coax connectors such as the VX1522.01. It is designed to lead the coax link from the POD to and through the ECU housing with low mechanical force to the POD. The FAKRA connector can be mount waterproof into the ECU housing.
Details	► Cable+U.FL-assemby: Hirose U.FL-LPHF6-088N1T-A-150 ► Mating cycles: FAKRA: max. 5; U.FL: max.30 (extraction tool Hirose U.FL-LP-N-2 needed!) ► Impedance: 50 Ohm ► Rated frequency: 0 GHz to 8 GHz ► Operating temperature: -40 °C ... +105 °C ► Minimum bending radius: 14 mm in a movable installation ► Minimum bending radius: 7 mm in a fixed installation
Part number	► **VX1382M.01 FAKRA Panelmount F0N100N0F** 22574

WARNING!
Violating the minimum bending radii can negatively affect the electical properties.

WARNING!
Unplugging the U.FL coaxial connector without extraction tool Hirose U.FL-LP-N-2 might damage either the connector on the PCB or on the cable!

Note
Technical drawings can be found at vector.com/vx1000-drawings.

7.5 HSSL Cables

7.5.1 VX137x HSSL Cables

The VX137x HSSL Cables for the VX1000 system are designed to resist the environmental stress inside the engine compartment.

- **VX1371F HSSL Panelmount D0N150A0A**
  HSSL Panelmount for HSSL POD
  HSSL Pigtail with HSSL socket to be mounted into a front panel.
  Connectors: HSSL socket to POD connector.
  Nominal length: 0.15 m
  Part number: 22290

- **VX1372 HSSL Cable 2m**
  HSSL Cable to connect a HSSL POD with Pigtail to a Base Module.
  Connectors: plugs on both ends
  Part number: 22197

- **VX1375 HSSL Cable 5m**
  HSSL Cable to connect a HSSL POD with Pigtail to a Base Module.
  Connectors: plugs on both ends.
  Part number: 22199

- **VX1376A HSSL Cable 5m 3H 45°**
  HSSL Cable to connect a HSSL POD with Pigtail to a Base Module.
  Connectors: plugs on both ends, one-sided 3H 45°
  Part number: 22201

- **VX1378 HSSL to HSSL2 Adapter Cable 5m**
  Cable to connect a HSSL POD with Pigtail to a Base Module.
  Adaption from Base Module with HSSL-Socket to Pigtail with HSSL2 socket.
  Connectors: HSSL plug to HSSL2 plug.
  Part number: 22599

Note
Operation of the VX1378 in reverse direction is not possible!
Overall system performance is limited to HSSL and not to HSSL2!

- **VX1379B HSSL Pigtail A500E13A40A**
  HSSL Pigtail for HSSL POD.
  Connectors: HSSL socket to POD connector
  Nominal length: 0.5 m
  Part number: 22237

- **VX1379C HSSL Pigtail A485A35A40A**
  HSSL Pigtail for HSSL POD.
  Connectors: HSSL socket to POD connector
  Nominal length: 0.5 m
  Part number: 22271

- **VX1379D HSSL Panelmount D0N300A40A**
  HSSL Panelmount for HSSL POD.
  HSSL Pigtail with HSSL socket to be mounted into a front panel.
  Connectors: HSSL socket to POD connector for HSSL POD.
  Nominal length: 0.34 m
  Part number: 22277

Technical data

Temperature range	-40 °C ... +130 °C (plugging and unplugging only allowed above 0 °C)
Minimum bending radius	7 cm
Mechanical stress	Vibration-proof due to latching mechanism in the connectors
IP classification	While connected, compliant to protection class IP67 (i. e. complete dust protection and waterproof up to 2 meter of submersion for 1 hour)

WARNING!
The instructions for handling the HSSL cable must be followed. Else the cable might get damaged beyond functionality!

Note
Technical drawings can be found at vector.com/vx1000-drawings.

Mechanical limitations

The VX137x HSSL cables are sensitive to both torsion and bending.

WARNING!
The cable must not be bent with a radius of less than 7 cm!
Only in the case, when the cable is permanently installed in a fixed position, a bending radius of down to 1.5 cm is possible.

WARNING!
Torsion of 90° must be distributed over a loose length of 25 cm!

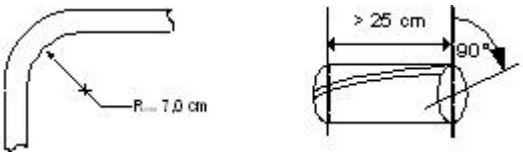


Figure 259: Limitations

Connector plug and sockets

The connector plug has a slightly moveable part which controls the latching mechanism (see fig. below). The socket has no moveable parts. It is either directly built into the POD or VX1000 Base Module or in some PODs located at the end of a short pigtail.

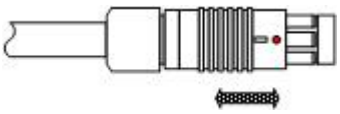


Figure 260: HSSL cable with connector plug

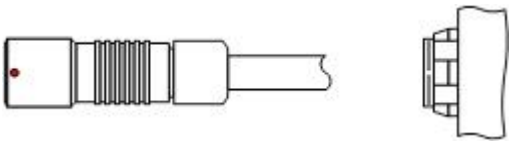


Figure 261: HSSL pigtail with cable socket (left) Built-in connector socket (right)

Plugging and unplugging

For connecting plug and socket, the red marking on both have to be aligned. Twisting misaligned connectors while pressing them together will not damage the contacts as they are mechanically protected. When twisting please mind the maximum allowed torsion of 90° per loose 25 cm of the VX137x HSSL Cable.
To disconnect the plug and socket, grab the plug at the moveable part and pull to unlock the latching mechanism and separate the plug and socket. If the latching mechanism is blocked, first push the plug and socket against each other before pulling them apart.

WARNING!
Never pull directly at the cable to unplug! The plug must be seized at the metal part which is marked in the figure below and then pulled out of the socket. It is essential to pull at this part of the plug to open the latching mechanism.

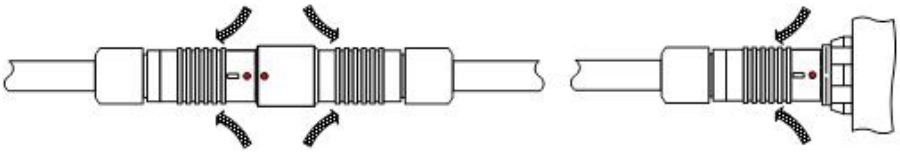


Figure 262: Metal parts

7.6 HSSL2 Cables

7.6.1 VX1381 HSSL2 Cables

The VX1381 HSSL2 Cables for the VX1000 system are designed to resist the environmental stress inside the engine compartment.

- **VX1381A HSSL2 Cable 2m**
  HSSL2 Cable to connect a HSSL POD with Pigtail to a Base Module.
  Connectors: plugs on both ends
  Part number: 22187

- **VX1381B HSSL2 Cable 5m**
  HSSL2 Cable to connect a HSSL POD with Pigtail to a Base Module.
  Connectors: plugs on both ends
  Part number: 22189

- **VX1381C HSSL2 Cable 8m**
  HSSL2 Cable to connect a HSSL POD with Pigtail to a Base Module.
  Connectors: plugs on both ends
  Part number: 22191

- **VX1381E HSSL2 Pigtail A500A20A76A**
  HSSL2 Pigtail for HSSL POD.
  Connectors: HSSL2 socket to POD connector
  Nominal length: 0.60 m
  Part number: 22193

- **VX1381K HSSL2 Panelmount D0N300N40A**
  HSSL2 Panelmount for HSSL POD.
  HSSL Pigtail with HSSL socket to be mounted into a front panel.
  Connectors: HSSL socket to POD connector for HSSL POD.
  Nominal length: 0.34 m
  Part number: 22493

Technical data

Temperature range	-40 °C ... +130 °C (plugging and unplugging only allowed above 0 °C)
Minimum bending radius	7 cm
Mechanical stress	Vibration-proof due to latching mechanism in the connectors
IP classification	While connected, compliant to protection class IP67 (i. e. complete dust protection and waterproof up to 2 meter of submersion for 1 hour)

WARNING!
The instructions for handling the HSSL2 cable must be followed. Else the cable might get damaged beyond functionality!

Note
Technical drawings can be found at vector.com/vx1000-drawings.

Mechanical limitations

The VX1381 HSSL2 cables are sensitive to both torsion and bending.

WARNING!
The cable must not be bent with a radius of less than 7 cm!
Only in the case, when the cable is permanently installed in a fixed position, a bending radius of down to 1.5 cm is possible.

WARNING!
Torsion of 90° must be distributed over a loose length of 25 cm!

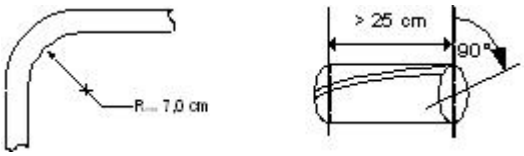


Figure 263: Limitations

Connector plug and sockets

The connector plug has a slightly moveable part which controls the latching mechanism (see fig. below). The socket has no moveable parts. It is either directly built into the POD or VX1000 Base Module or in some PODs located at the end of a short pig- tail

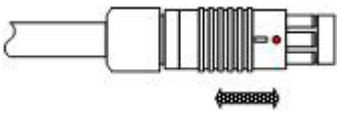


Figure 264: HSSL2 cable with connector plug

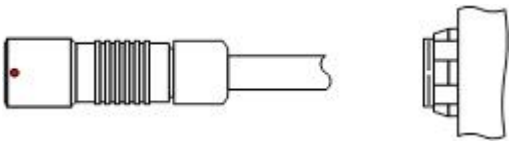


Figure 265: HSSL2 pigtail with cable socket (left) Built-in connector socket (right)

Plugging and unplugging

For connecting plug and socket, the red marking on both have to be aligned. Twisting misaligned connectors while pressing them together will not damage the contacts as they are mechanically protected. When twisting please mind the maximum allowed torsion of 90° per loose 25 cm of the VX1381 HSSL2 Cable.

To disconnect the plug and socket, grab the plug at the moveable part and pull to unlock the latching mechanism and separate the plug and socket. If the latching mechanism is blocked, first push the plug and socket against each other before pulling them apart.

WARNING!
Never pull directly at the cable to unplug! The plug must be seized at the metal part which is marked in the figure below and then pulled out of the socket. It is essential to pull at this part of the plug to open the latching mechanism.

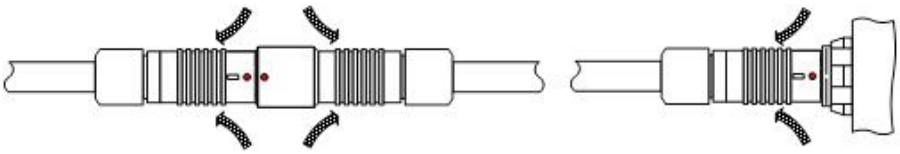


Figure 266: Metal parts

7.7 Power/Sync Cables

7.7.1 Vector Power Supply 12V/1.5A

Power supply unit for Vector devices

Description	Power supply unit with 12 V and 1.5 A, with changeable adapters for USA and Europe (GB adapter on request)
Length	2 m
Input voltage	100 V ... 240 V AC (+/-10 %), 50 Hz ... 60 Hz
Connectors	1x adapter for USA 1x adapter for Europe 1x 3-pin Binder connector (type 711)
Part number	05024

7.7.2 Vector Power Supply ODU Mini-Snap

Power supply unit for Vector devices

Description	Power supply unit with 24 V and 2.5 A. Power cord not included.
Input voltage	100 V ... 240 V AC (+/-10 %), 47 Hz ... 63 Hz
Connectors	1x Connector for Europe IEC60320 C8 1x ODU S11L0C-P02NPL0-5200
Dimensions	Housing: 114.5 mm x 49.5 mm x 33 mm (LxWxH) Cable: approx. 2 m
Part number	05068

Connection cable

Description	Connection cable for power supply unit (part number 05068)
Connectors	Type-C (CEE7/16) to IEC 60320 C7
Length	Approx. 2 m
Part number	05251

Connection cable

Description	Connection cable for power supply unit (part number 05068)
Connectors	Type-A to IEC 60320 C7
Length	Approx. 2 m
Part number	05071

Connection cable

Description	Connection cable for power supply unit (part number 05068)
Connectors	Type-G („UK”, BS1363) to IEC60320 C7
Length	Approx. 2 m
Part number	05070

Connection cable

Description	Connection cable for power supply unit (part number 05068)
Connectors	"PRC2" (Type A) to IEC60320 C7
Length	Approx. 2 m
Part number	05249

Connection cable

Description	Connection cable for power supply unit (part number 05068)
Connectors	Type-D to IEC60320 C7
Length	Approx. 1.8 m
Part number	05250

7.7.3 Desktop Type Power Supply

Technical data

Description	Power supply - AC 110...240 V - 50/60 Hz - 220 W - 24 V - 9.5 A - incl. power cord EURO CEE7/7
Part number	05116

Connection cable Europe

Description	Connection cable for power supply unit
Connectors	Connector for Europe
Part number	05244

Connection cable USA/Japan

Description	Connection cable for power supply unit
Connectors	Connector for USA/Japan
Part number	05115

Connection cable UK

Description	Connection cable for power supply unit
Connectors	Connector for UK
Part number	05125

Connection cable China

Description	Connection cable for power supply unit
Connectors	Connector for China
Part number	05114

7.7.4 Cable Vehicle Input <> ODU Mini-Snap

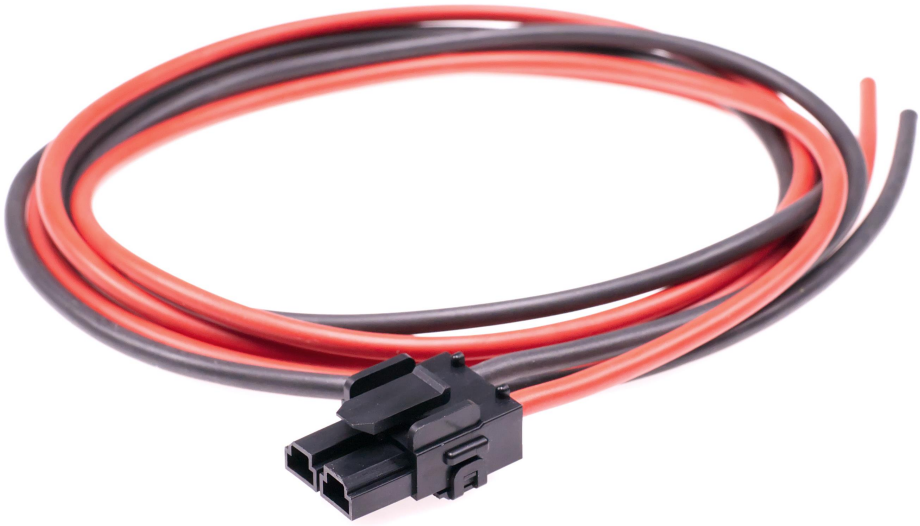
On-board power supply

Description	On-board power supply cable
Connectors	1x ODU S11L0C-P02NPL0-5200 1x 12 V plug (DIN ISO 4165)
Temp. range	Fixed installation: -20 °C ... +70 °C Flexible use: -20 °C ... +70 °C
Part number	05076

7.7.5 Power Cable Molex Mini-Fit 4mm²

Technical data

Description	Red/black 4 mm² power cable for Vector devices with Molex Mini-Fit connector.
Length	1.5 m
Connectors	1x Molex Mini-Fit connector 2x open end
Part number	22515

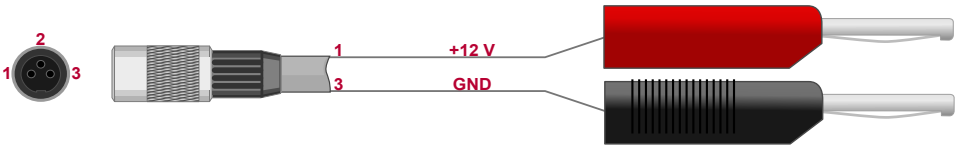


7.7.6 Cable Banana Plug <> Binder 3-pin

Power cable

Description	Power cable for Vector devices
Length	1.5 m
Connectors	1x female 3-pin Binder connector (type 711) 2x banana plugs (4 mm)
Temp. range	Fixed installation: -25 °C ...+70 °C Flexible use: -5 °C ...+70 °C
Min. bending radius	Fixed installation: 16.8 mm Flexible use: 42.0 mm
Part number	30012

Setup



WARNING!
To protect against hazardous situations, the equipment (cable and Vector device) must be connected only to power supplies providing limited energy. This can be achieved by using slow blow fuses or equivalent. The specific voltage and current ranges of the connected Vector device must be taken into account.

7.7.7 Power Sync Cable 2xBinder <> Banana Plugs

Power cable

Description	Power cable with hardware synchronization capability for Vector devices (requires Vector devices with hardware synchronization support)
Length	1.5 m
Connectors	2x female 3-pin Binder connector (type 711) 2x banana plugs (4 mm)
Part number	05123

Setup

WARNING!
To protect against hazardous situations, the equipment (cable and Vector device) must be connected only to power supplies providing limited energy. This can be achieved by using slow blow fuses or equivalent. The specific voltage and current ranges of the connected Vector device must be taken into account.

7.7.8 Cable Banana Plug <> ODU Mini-Snap

ODU

Description	Power cord for Vector devices with ODU MINI-SNAP connector
Length	approx. 2 m
Connectors	1x ODU connector (type S11L0C-P02NPL0-6200) 2x bunch plugs (power supply)
Temp. range	Fixed installation: -40 °C ...+75 °C Flexible use: -40 °C ...+75 °C
Min. bending radius	Fixed installation: 15.6 mm Flexible use: 26.0 mm
Part number	05069

Cable setup

WARNING!
To protect against hazardous situations, the equipment (cable and Vector device) must be connected only to power supplies providing limited energy. This can be achieved by using slow blow fuses or equivalent. The specific voltage and current ranges of the connected Vector device must be taken into account.

7.7.9 Car Power Supply Cable 12V with Binder

On-board power supply

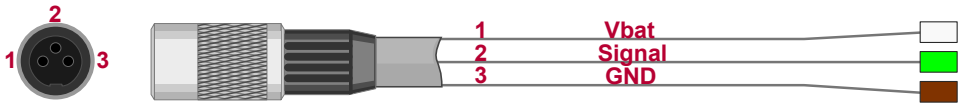
Description	On-board power supply cable
Connectors	1x 3-pin connector (Binder type 711) 1x 12 V plug (DIN ISO 4165)
Temp. range	Fixed installation: -20 °C ...+70 °C Flexible use: -20 °C ...+70 °C
Part number	15023

7.7.10 Cable Binder 3pol Connector with Pigtail

All-purpose cable

Description	Connection cable for time synchronization at Vector devices
Length	1.5 m
Connectors	1x female 3-pin Binder connector (type 711) 3x stripped wires
Temp. range	Fixed installation: -25 °C ...+70 °C Flexible use: -5 °C ...+70 °C
Min. bending radius	Fixed installation: 16.8 mm Flexible use: 42.0 mm
Part number	30011

Setup



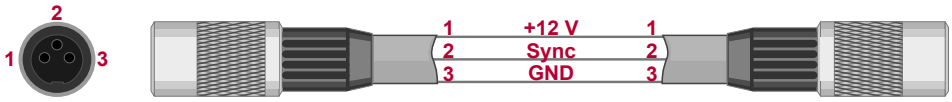
WARNING!
To protect against hazardous situations, the equipment (cable and Vector device) must be connected only to power supplies providing limited energy. This can be achieved by using slow blow fuses or equivalent. The specific voltage and current ranges of the connected Vector device must be taken into account.

7.7.11 Vector SYNCcableXL

Synchronization cable

Description	Connection cable for time synchronization of Vector devices
Length	2 m
Connectors	2x female 3-pin Binder connectors (type 711)
Temp. range	Fixed installation: -30 °C ...+80 °C Flexible use: -5 °C ...+70 °C
Min. bending radius	Fixed installation: 16 mm Flexible use: 40 mm
Part number	05018

Setup



7.8 Serial Cables

7.8.1 VX134x Serial Cables

The VX134x Serial Cables for the VX1000 system are designed to resist the environmental stress inside the engine compartment.

- **VX1341F Serial Panelmount D0N0E160A**
  Serial Panelmount for Serial POD.
  Serial Pigtail with Serial socket to be mounted into a front panel.
  Connectors: Serial socket to POD connector for VX1543A
  Nominal length: 0.15 m
  Part number: 22317

- **VX1341G Serial Panelmount D0N0E150D**
  Serial Panelmount for Serial POD.
  Serial Pigtail with Serial socket to be mounted into a front panel.
  Connectors: Serial socket to POD connector for VX1544x/VX1543B
  Nominal length: 0.15 m
  Part number: 22318

- **VX1342 Serial Cable 2m**
  Serial Cable to connect a Serial POD with Panelmount to a Base Module.
  Connectors: plugs on both ends
  Part number: 22184

- **VX1345 Serial Cable 5m**
  Serial Cable to connect a Serial POD with Panelmount to a Base Module.
  Connectors: plugs on both ends
  Part number: 22186

- **VX1345A Serial Extension 5m**
  Serial Extension cable for Serial Pigtail.
  Connectors: plug to socket
  This cable can only be combined with a Pigtail of 0.7 m length!
  Including dust covers for both sides.
  Yellow mark on female socket side.
  Part number: 22228

- **VX1346A Serial Extension 8m**
  Serial Extension cable for Serial Pigtail.
  Connectors: plug to socket
  This cable can only be combined with a Pigtail of 0.7 m length!
  Including dust covers for both sides.
  Yellow mark on female socket side.
  Part number: 22270

- **VX1348A Serial Pigtail B2465E0E35A**
  Serial Pigtail for Serial POD.
  Connectors: Serial plug to POD connector for VX1543A.
  Nominal length: 2.5 m
  Part number: 22213

- **VX1348B Serial Pigtail B2465N0F35D**
  Serial Pigtail for Serial POD.
  Connectors: Serial plug to POD connector for VX1544x/VX1543B
  Nominal length: 2.5 m
  Part number: 22273

- **VX1349A.01 Serial Pigtail B715E0E35A**
  Serial Pigtail for Serial POD.
  Connectors: Serial plug to POD connector for VX1543A.
  Nominal length: 0.75 m
  Part number: 120070

- **VX1349B Serial Pigtail B715N0F35D**
  Serial Pigtail for Serial POD.
  Connectors: Serial plug to POD connector for VX1544x/VX1543B
  Nominal length: 0.75 m
  Part number: 22274

Technical data

Temperature range	-40 °C ... +125 °C (plugging and unplugging only allowed above 0 °C)
Minimum bending radius	6 cm
Mechanical stress	Vibration-proof due to latching mechanism in the connectors
IP classification	While connected, compliant to protection class IP60 (i. e. complete dust protection and splash water)

WARNING!
The instructions for handling the Serial cable must be followed. Else the cable might get damaged beyond functionality!

Note
The locking and unlocking mechanism of the VX134x Serial Cable is similar to the VX137x HSSL-Cable. The handling instructions can be found in section VX137x HSSL Cables on page 639.

WARNING!
The cable must not be bent with a radius of less than 6 cm!
Only in the case, when the cable is permanently installed in a fixed position, a bending radius of down to 1.5 cm is possible.

Note
Technical drawings can be found at vector.com/vx1000-drawings.

7.8.2 VX1311 Serial2 Cables

The VX1311x Serial2 Cables for the VX1000 system are designed to resist the environmental stress inside the engine compartment.

- **VX1311A Serial2 Extension 2m**
  Serial2 Extension cable for Serial or Serial2 Pigtail.
  Connectors: plug to socket.
  The Extension Cable can only be combined with a Pigtail of 0.7 m length!
  Part number: 22352

- **VX1311B Serial2 Extension 5m**
  Serial2 Extension cable for Serial or Serial2 Pigtail.
  Connectors: plug to socket.
  The Extension Cable can only be combined with a Pigtail of 0.7 m length!
  Part number: 22353

- **VX1311C Serial2 Extension 8m**
  Serial2 Extension cable for Serial or Serial2 Pigtail.
  Connectors: plug to socket.
  The Extension Cable can only be combined with a Pigtail of 0.7 m length!
  Part number: 22354

- **VX1311D Serial2 Cable 2m**
  Serial2 Cable to connect a Serial POD with Panelmount to a Base Module.
  Connectors: plugs on both ends
  Part number: 22355

- **VX1311E Serial2 Cable 5m**
  Serial2 Cable to connect a Serial POD with Panelmount to a Base Module.
  Connectors: plugs on both ends
  Part number: 22356

- **VX1311F Serial2 Cable 8m**
  Serial2 Cable to connect a Serial POD with Panelmount to a Base Module.
  Connectors: plugs on both ends
  Part number: 22357

- **VX1311G Serial2 Pigtail B750A6E40D**
  Serial2 Pigtail for Serial POD.
  Connectors: Serial plug to POD connector for VX1544x/VX1543B
  Nominal length: 0.75 m
  Can be directly plugged into a Base Module.
  Part number: 22499

- **VX1311H Serial2 Pigtail B2500A6E40D**
  Serial2 Pigtail for Serial POD.
  Connectors: Serial plug to POD connector for VX1544x/VX1543B
  Nominal length: 2.5 m
  Can be directly plugged into a Base Module.
  Part number: 22498

Technical data

Temperature range	-40 °C ... +125 °C (plugging and unplugging only allowed above 0 °C)
Minimum bending radius	7 cm
Mechanical stress	Vibration-proof due to latching mechanism in the connectors
IP classification	While connected, compliant to protection class IP67 (i. e. complete dust protection and splash water)

WARNING!
The instructions for handling the Serial2 cable must be followed. Else the cable might get damaged beyond functionality!

Note
The locking and unlocking mechanism of the VX1311x Serial2 Cable is similar to the VX137x HSSL-Cable. The handling instructions can be found in section VX137x HSSL Cables on page 639.

WARNING!
The cable must not be bent with a radius of less than 7 cm!
Only in the case, when the cable is permanently installed in a fixed position, a bending radius of down to 2 cm is possible.

Note
Technical drawings can be found at vector.com/vx1000-drawings.

7.9 Ethernet Cables

7.9.1 AEcable 2Y Family

7.9.1.1 General Information

Connection cable

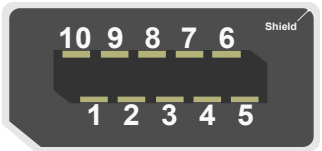
Description	Connection cable for 1000BASE-T1 and 100BASE-T1 Ethernet network interfaces with an ix connector, a splitter box and different ends for multiple use cases (see section Connector Options on page 662). The splitter box separates the two Ethernet channels into two separate cables. The shieldings of CHA and CHB are connected to each other, but are electrically isolated from the shield of the network interface via capacitors.
Length	approx. 1.8 m*
Rated voltage	30 V
HV test	300 V (0.01 s)
Connectors	1x ix connector (type A, 10 pole, male) Harting: 09 45 181 2561 XL / Hirose IX30G-A-10S-CV(7.0)

* exceptions see section Connector Options on page 662

AEcable 2Y with H-MTD Z connectors

Pin assignment

Pin	Assignment	Color
1	CHB MDI+	white/orange
2	CHB MDI-	orange
3	Not connected	-
4	Not connected	blue
5	Not connected	white/blue
6	CHA MDI+	white/green
7	CHA MDI-	green
8	Not connected	-
9	Not connected	white/brown
10	Not connected	brown
Shield	Shield	-



7.9.1.2 Connector Options

Overview

The following table shows an overview of the available connectors for the AEcable 2Y:

AEcable 2Y	Connectors	Part no.
EVA	2x open end - Single strand cross section 0.14 mm² (seven tinned copper wires) - Outer diameter one conductor: about 1.23 mm - Outer diameter cable: about 3.8 mm - Similar to LEONI Dacar 686-3 or Gebauer&Griller 2Speed 251 Technical data - Rated voltage: 30 V - Min. bending radius: 37.2 mm - Max. operating temp.: 75 °C - Cable: white (MDI+) / green (MDI-)	05124
EVA 10m	2x open end - Cable length: 10 m - Single strand cross section 0.14 mm² (seven tinned copper wires) - Outer diameter one conductor: about 1.23 mm - Outer diameter cable: about 3.8 mm - Similar to LEONI Dacar 686-3 or Gebauer&Griller 2Speed 251 Technical data - Rated voltage: 30 V - Min. bending radius: 37.2 mm - Max. operating temp.: 75 °C - Cable: white (MDI+) / green (MDI-)	05211
H-MTD Zjp	1x H-MTD, coding Z, type jack Manufacturer part number: Rosenberger E6K10A-1CAZ5-Z 1x H-MTD, coding Z, type plug Manufacturer part number: Rosenberger E6S10A-1CAZ5-Z Technical data - Rated voltage: 30 V - Min. bending radius: 37.2 mm - Max. operating temp.: 75 °C	05134
H-MTD Zjj	2x H-MTD, coding Z, type jack Manufacturer part number: Rosenberger E6K10A-1CAZ5-Z Technical data - Rated voltage: 30 V - Min. bending radius: 37.2 mm - Max. operating temp.: 75 °C	05133
H-MTD Zpp	2x H-MTD, coding Z, type plug Manufacturer part number: Rosenberger E6S10A-1CAZ5-Z Technical data - Rated voltage: 30 V - Min. bending radius: 37.2 mm - Max. operating temp.: 75 °C	05135
HSD Zjp	1x HSD, coding Z, type jack Manufacturer part number: Rosenberger D4K10A-1D5A5-Z 1x HSD, coding Z, type plug Manufacturer part number: Rosenberger D4S10A-1D5A5-Z	05126
HSD Zjj	2x HSD, coding Z, type jack Manufacturer part number: Rosenberger D4K10A-1D5A5-Z Technical data	05128

AEcable 2Y	Connectors	Part no.
	- Rated voltage: 30 V - Min. bending radius: 37.2 mm - Max. operating temp.: 75 °C	
HSD Zpp	2x HSD, coding Z, type plug Manufacturer part number: Rosenberger D4S10A-1D5A5-Z Technical data - Rated voltage: 30 V - Min. bending radius: 37.2 mm - Max. operating temp.: 75 °C	05129
MATEnet Zfh	1x MATEnet, coding Z, type frame Manufacturer part number: TE (Tyco Electronics, TE Connectivity) 1-1703930-1 MATEnet NanoMQS Receptacle Terminal 1-2302450-1 MATEnet female insert modul 2302454-9 MATEnet 1-port Frame UTP Z-Coding 1x MATEnet, coding Z, type header Manufacturer part number: TE (Tyco Electronics, TE Connectivity) 2236905-1 MATEnet Nano TAB 0.5x0.4 Terminal 2302451-1 MATEnet male insert modul 4-2302461-9 MATEnet 1-port Inline Coupler Z-Coding Technical data - Rated voltage: 30 V - Min. bending radius: 37.2 mm - Max. operating temp.: 75 °C	05137
MATEnet Zff	2x MATEnet, coding Z, type frame Manufacturer part number: TE (Tyco Electronics, TE Connectivity) 1-1703930-1 MATEnet NanoMQS Receptacle Terminal 1-2302450-1 MATEnet female insert modul 2302454-9 MATEnet 1-port Frame UTP Z-Coding Technical data - Rated voltage: 30 V - Min. bending radius: 37.2 mm - Max. operating temp.: 75 °C	05138
MATEnet Zhh	2x MATEnet, coding Z, type header Manufacturer part number: TE (Tyco Electronics, TE Connectivity) 2236905-1 MATEnet Nano TAB 0.5x0.4 Terminal 2302451-1 MATEnet male insert modul 4-2302461-9 MATEnet 1-port Inline Coupler Z-Coding Technical data - Rated voltage: 30 V - Min. bending radius: 37.2 mm - Max. operating temp.: 75 °C	05136
DSUB9 male	2x D-SUB9, type male Technical data - Rated voltage: 30 V - Min. bending radius: 37.2 mm - Max. operating temp.: 75 °C	05119
RJ45pp	2x RJ45, type plug Technical data - Rated voltage: 30 V - Min. bending radius: 37.2 mm - Max. operating temp.: 75 °C	05131
RJ45jj	2x RJ45, type jack Technical data	05130

AEcable 2Y	Connectors	Part no.
	- Rated voltage: 30 V - Min. bending radius: 37.2 mm - Max. operating temp.: 75 °C	
RJ45jp	1x RJ45, type plug 1x RJ45, type jack Technical data - Rated voltage: 30 V - Min. bending radius: 37.2 mm - Max. operating temp.: 75 °C	05132

H-MTD Z connectors

Figure 267: Jack

Figure 268: Plug

HSD Z connectors

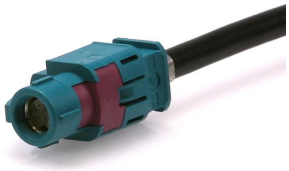


Figure 269: Jack

Figure 270: Plug

MATEnet Z connectors

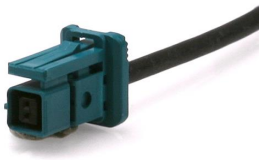


Figure 271: Frame

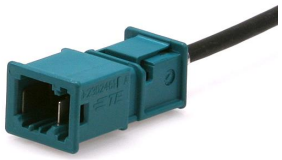


Figure 272: Header

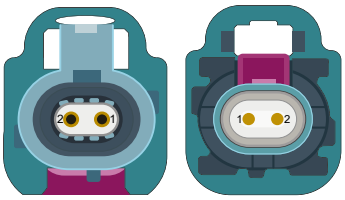
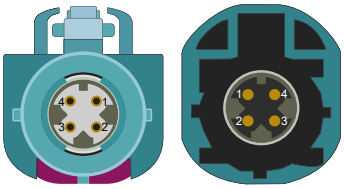
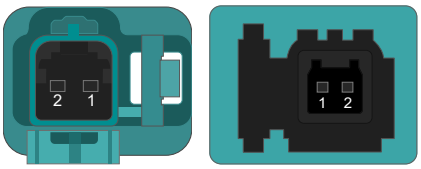
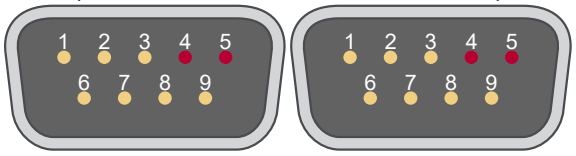
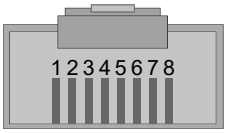
D-SUB9 / RJ45 connectors

Figure 273: D-SUB 9 (plug)

Figure 274: RJ45 (plug)

Pin assignment

The following table shows the pin assignment depending on the connector type:

Connector Type	Ethernet MDI+	Ethernet MDI-
H-MTD Z  jack / plug	Pin 1	Pin 2
HSD Z  jack / plug	Pin 2	Pin 3
MATEnet Z  frame / header	Pin 1	Pin 2
D-SUB9 (2x) (one MDI channel at each connector) 	Pin 4	Pin 5
RJ45 	Pin 1	Pin 2

EVA

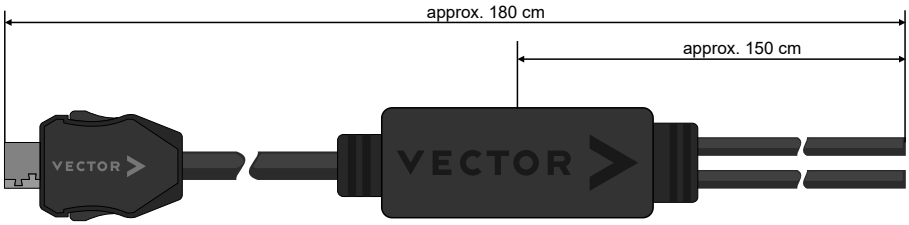
Color	Assignment
White	MDI+
Green	MDI-

Note
Please note that the AEcable 2Y DSUB9 male is intended for use with 100BASE-T1 only. All other cables are intended for 1000BASE-T1 and 100BASE-T1.

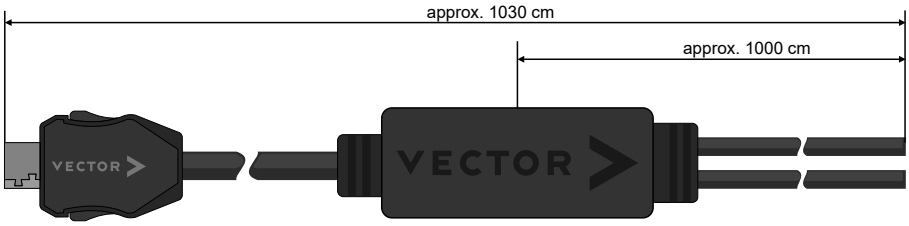
Cable length

The cable lengths vary depending on the cable option. The length of the H-MTD, HSD, MATEnet, D-SUB or RJ45 connector will increase the overall length accordingly.

AEcable 2Y EVA



AEcable 2Y EVA 10m



7.9.2 Ethernet Cable 10Gbps 0.5m

Connection cable

Description	Cable for Ethernet network interfaces. Designed for vehicle interiors.
Length	approx. 0.5 m
Connectors	2x RJ45 (LxWxH: 41.9 x 12.1 x 15.6 mm)
Category	Cat 6A
Temp. range	Fixed installation: -20 °C ...+60 °C
Min. bending radius	Fixed installation: 87 mm
Outer diameter	6.2 mm
Part number	30017

7.9.3 Ethernet Cable 10Gbps 2m

Connection cable

Description	Cable for Ethernet network interfaces. Designed for vehicle interiors.
Length	approx. 2 m
Connectors	2x RJ45 (LxWxH: 41.9 x 12.1 x 15.6 mm)
Category	Cat 6A
Rated voltage	250 V
Temp. range	Fixed installation: -20 °C ...+60 °C
Min. bending radius	Fixed installation: 87 mm
Outer diameter	6.2 mm
Part number	30016

7.9.4 Ethernet Cable 10Gbps 7.5m

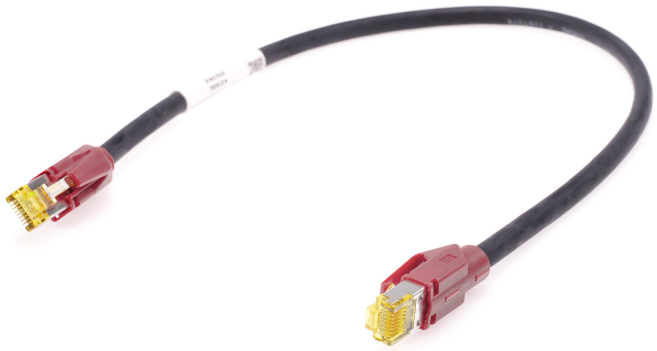
Connection cable

Description	Cable for Ethernet network interfaces. Designed for vehicle interiors.
Length	approx. 7.5 m
Connectors	2x RJ45 (LxWxH: 41.9 x 12.1 x 15.6 mm)
Category	Cat 6A
Temp. range	-20 °C ...+60 °C
Min. bending radius	60 mm
Outer diameter	6.2 mm
Part number	30018

7.9.5 Ethernet Cable Robust 10Gbps 0.5m

Connection cable

Description	Robust cable for Ethernet network interfaces. Designed for vehicle interiors during demanding long-term vehicle testing, with extended temperature range and higher cable robustness against oil, fuel and ozone.
Length	approx. 0.5 m
Connectors	2x RJ45 (LxWxH: 44.7 x 13.4 x 16.1 mm)
Temp. range	-40 °C ...+80 °C
Category	Cat 6A
Min. bending radius	Fixed installation: 40 mm
Outer diameter	6.6 mm
Part number	30024



7.9.6 Ethernet Cable Robust 10Gbps 2.0m

Connection cable

Description	Robust cable for Ethernet network interfaces. Designed for vehicle interiors during demanding long-term vehicle testing, with extended temperature range and higher cable robustness against oil, fuel and ozone.
Length	approx. 2 m
Connectors	2x RJ45 (LxWxH: 44.7 x 13.4 x 16.1 mm)
Temp. range	-40 °C ...+80 °C
Category	Cat 6A
Min. bending radius	Fixed installation: 40 mm
Outer diameter	6.6 mm
Part number	30025

7.9.7 Ethernet Cable Robust 10Gbps 7.5m

Connection cable

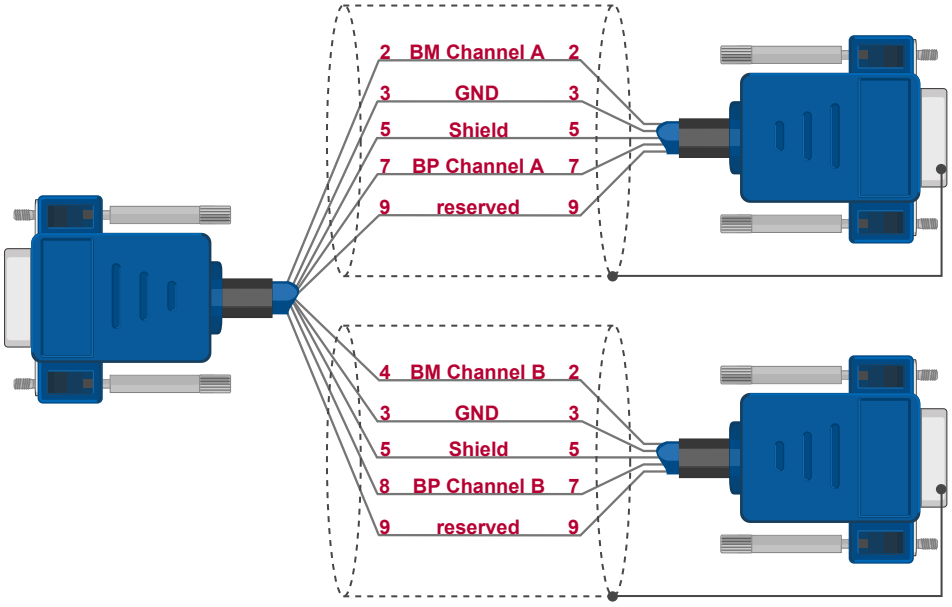
Description	Robust cable for Ethernet network interfaces. Designed for vehicle interiors during demanding long-term vehicle testing, with extended temperature range and higher cable robustness against oil, fuel and ozone.
Length	approx. 7.5 m
Connectors	2x RJ45 (LxWxH: 44.7 x 13.4 x 16.1 mm)
Temp. range	-40 °C ...+80 °C
Category	Cat 6A
Min. bending radius	Fixed installation: 40 mm
Outer diameter	6.6 mm
Part number	30026

7.10 FlexRay Cables

7.10.1 FRcable AB

Description	Cable for connection of a FlexRay Interface to the FlexRay bus (Channel A and B)
Length	1 m
Connectors	3x D-SUB9 connectors (female)
Temp. range	Fixed installation: -20 °C ...+60 °C Flexible use: 0 °C ...+50 °C
Min. bending radius	Fixed installation: 20 mm
Properties	Provides channel A and B at the FlexRay interface. The pin assignment of both single ended connectors is identical and suitable for replacement of an existing FlexCard configuration.
Part number	Included in FRcable Set (part number 05062)

Setup

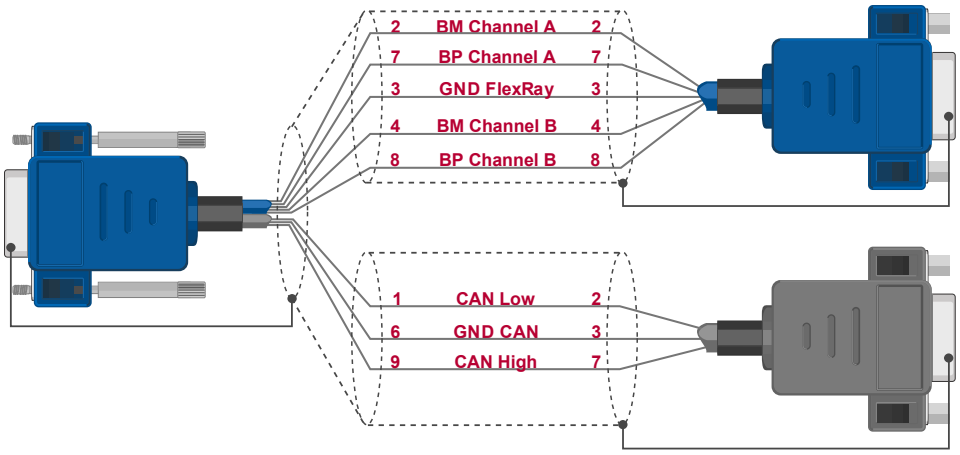


7.10.2 FR/CANcable 2Y

Y cable

Description	Y cable for Vector FR/CAN interfaces with D-SUB9 double assignment (e. g. VN7610). Splits the double assignment into two separate D-SUB9 connectors.
Length	0.3 m
Connectors	1x female D-SUB9 connector 2x male D-SUB9 connectors
Temp. range	Fixed installation: -20 °C ...+60 °C Flexible use: 0 °C ...+50 °C
Min. bending radius	Fixed installation: 20 mm
Part number	05099

Setup



7.11 Mounting Equipment

7.11.1 Fix Kit 32mm Device

Technical data

Description	Fixing kit containing 4 pcs. of wall brackets for ALUbos housing incl. seals and screws. Suitable for VN1630A or VX1060 housing height.
Part number	07139

Note

- The Fix Kit may only be used for Vector devices.
- Always use all 4 brackets incl. screws for mounting.
- For devices up to max. 5 kg.
- For fixing the mounting brackets to the floor, wall or ceiling, suitable screws must be selected to ensure permanent fixing according to the expected load (not included in the scope of delivery).
- In case of uncertainty and questions, please contact qualified personnel.

7.11.2 Fix Kit Large Device

Technical data

Description	Fixing kit containing 4 pcs. of wall brackets for ALUbos housing 1600/1680 incl. seals and screws. Suitable for VN8900 or VX113x housing height.
Part number	07189

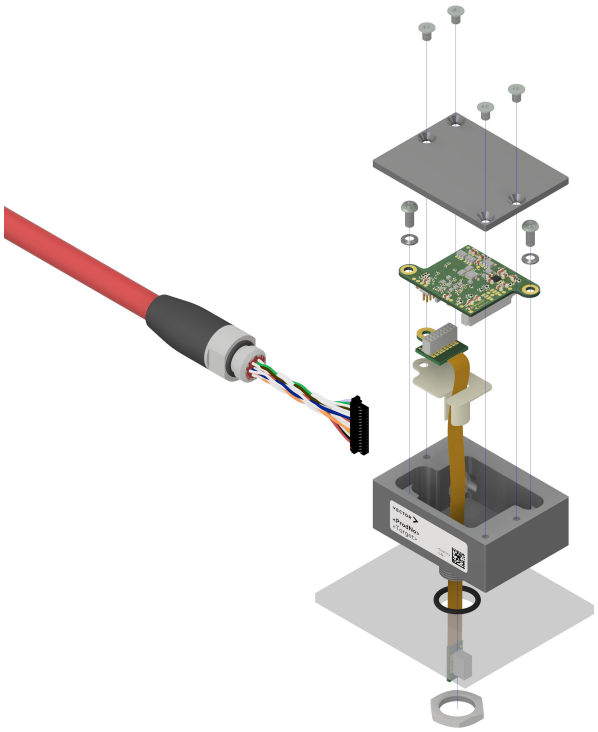
Note

- The Fix Kit may only be used for Vector devices.
- Always use all 4 brackets incl. screws for mounting.
- For devices up to max. 5 kg.
- For fixing the mounting brackets to the floor, wall or ceiling, suitable screws must be selected to ensure permanent fixing according to the expected load (not included in the scope of delivery).
- In case of uncertainty and questions, please contact qualified personnel.

7.11.3 Serial POD Enclosure Gen2

Technical data

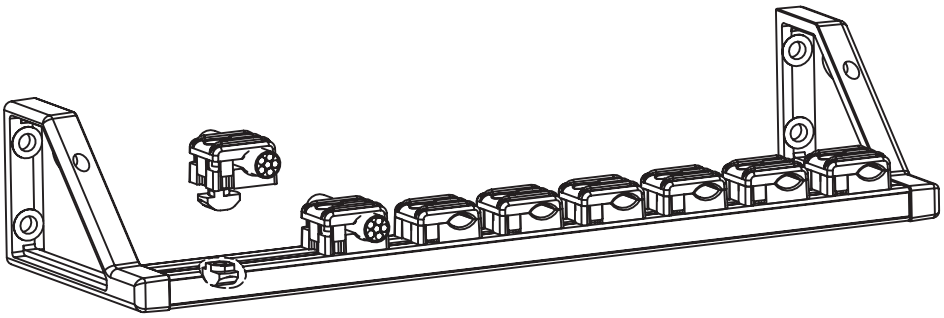
Description	This metallic enclosure is designed for VX154x serial PODs to protect against environmental and electromagnetic interference (EMC). The enclosure protects the POD from electromagnetic interference (EMC). The GND potential of the target connector is therefore routed through the adapter cable. Since the POD is fixed in the enclosure, the GND potential of the ECU is also available outside the enclosure. For a detailed description of the ground potential see section Galvanic Isolation and Ground Concept on page 713.
Ratings	► IP60 in combination with VX134x Serial Pigtails ► IP67 in combination with VX1311 Serial2 Pigtails
Dimensions	44.0 x 32.5 x 18.0 mm
Part number	120498

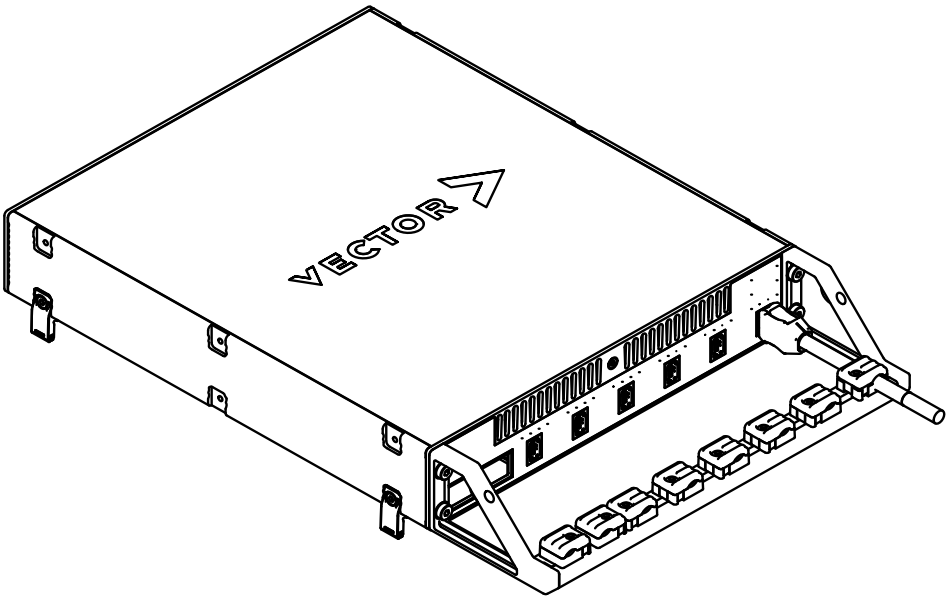


7.11.4 VSH Cable Guard 216

Technical data

Description	Cable support system including eight clips for clamping single cables. For cable diameters (clamping range) from Ø 3.8 mm to Ø 6.8 mm.
Part number	05149



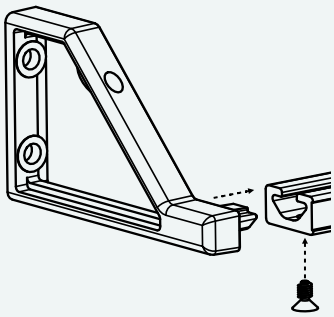


Note
The VSH Cable Guard 216 and the clips are not designed for use with VNcable 3Y.

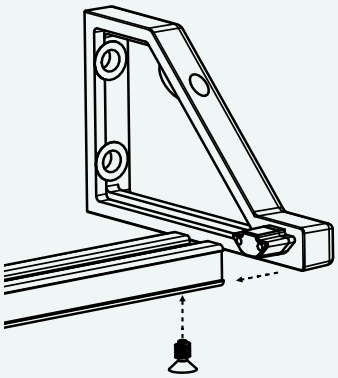
Step by Step Procedure

The following steps describe the mounting of the VSH Cable Guard 216:

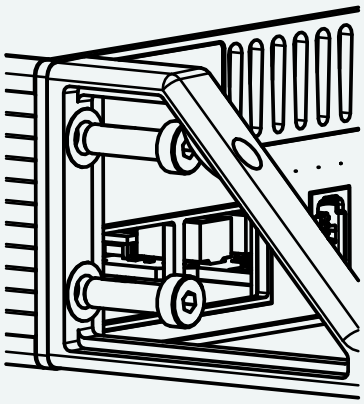
1. Insert the left frame into the cable rail and tighten the screw (Torx 8) with a torque of 1.4 Nm.



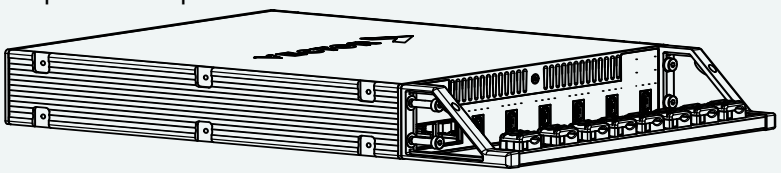
2. Repeat the steps above with the right frame.



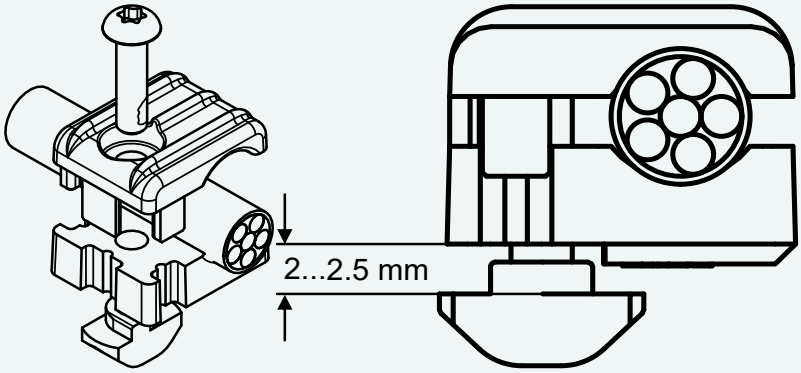
3. Attach the complete Cable Guard to the VN device and tighten it with the enclosed hexagonal screws (3 mm) and with a torque of 1.7 Nm.



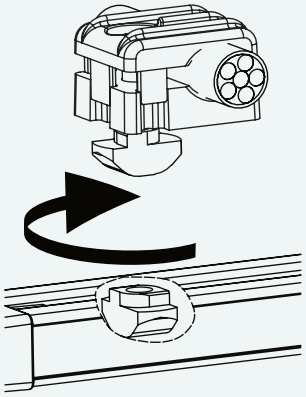
4. Repeat the steps on the other side.



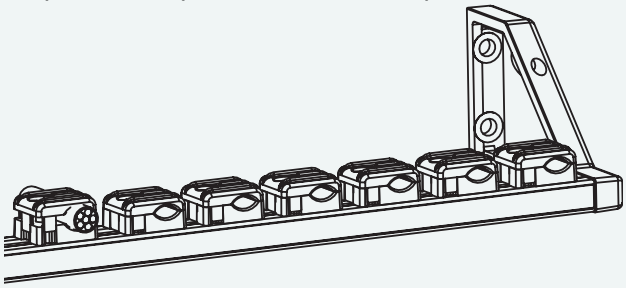
5. Lead the individual cables through the clamps and tighten the individual clamps depending on the cable (Torx 10). Tighten the screw only enough to leave a gap of 2 to 2.5 mm between the slot nut and the cable clamp.



6. Now insert the individual clamps into the Cable Guard and tighten the screw completely. The slot nut will rotate 90° inside the rail and thus fix the entire clamp.



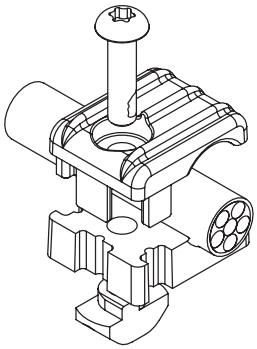
7. Repeat the steps with all other clamps.

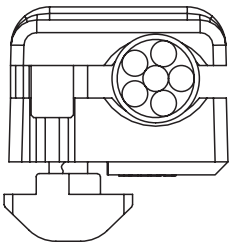


7.11.5 VSH Cable Guard 216 Clip

Technical data

Description	Set of four single clips for VSH Cable Guard 216 (see section VSH Cable Guard 216 on page 678). For single cables.
Part number	05148

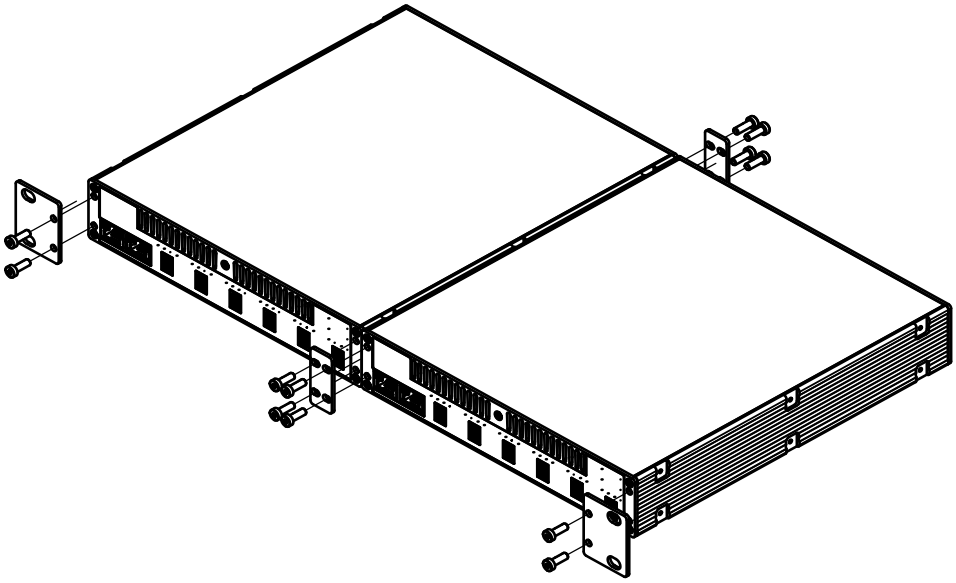


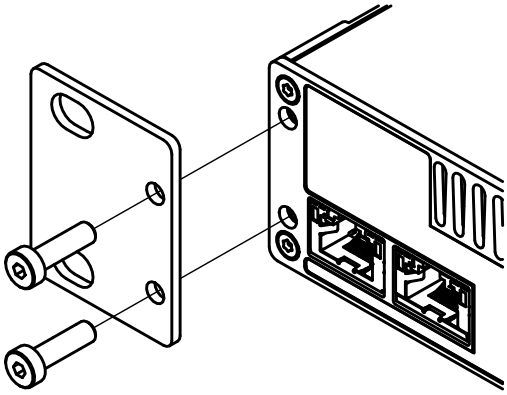


7.11.6 VSH Connection Kit 19"

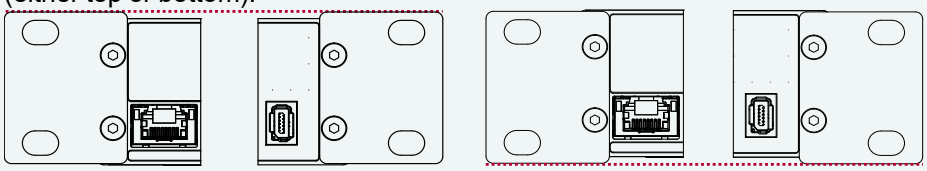
Technical data

Description	Brackets for 19" rack mounting (front and back mounting plates required, see VSH Connection Kit Horizontal on page 684).
Part number	05151



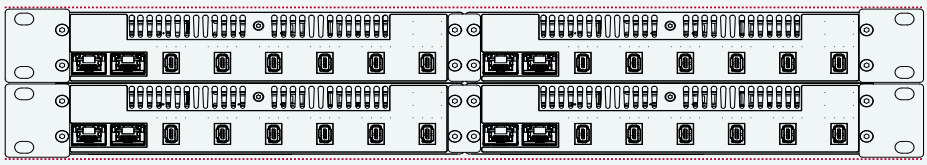


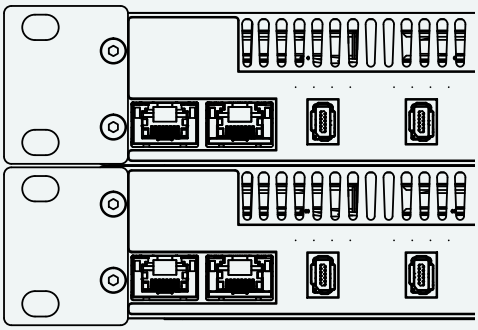
WARNING!
To avoid mechanical damage to the accessories, the unit or the installation, make sure that the two mounting brackets are always mounted in alignment (either top or bottom):



Note

For mounting two devices on top of each other, the rack brackets must be aligned accordingly. The rack brackets must not overlap between the devices!





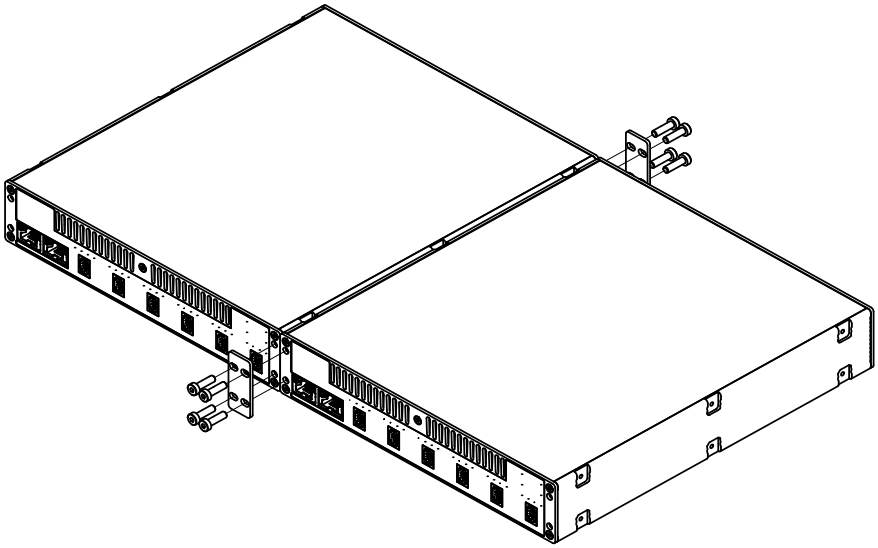
Note

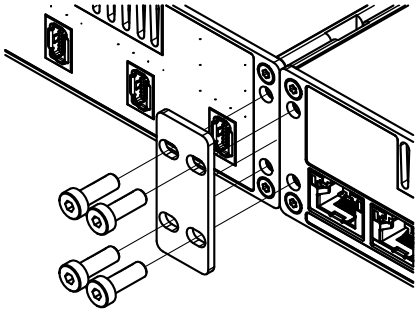
- Elements may only be used for Vector devices in the Vector System Housing form factor.
- Always use both supplied elements and associated screws for mounting.
- For systems up to max. 20 kg.
- For mounting on a 19" rack, suitable screws must be selected to ensure permanent fixing according to the expected load (not included in the scope of delivery).
- In case of uncertainty and questions, please contact qualified personnel.
- Tightening torque: hexagon socket 3 mm, 1.7 Nm.

7.11.7 VSH Connection Kit Horizontal

Technical data

Description	Mounting plates for connecting two devices side by side (horizontally). The mounting is done on the front and back side of the devices.
Part number	05153





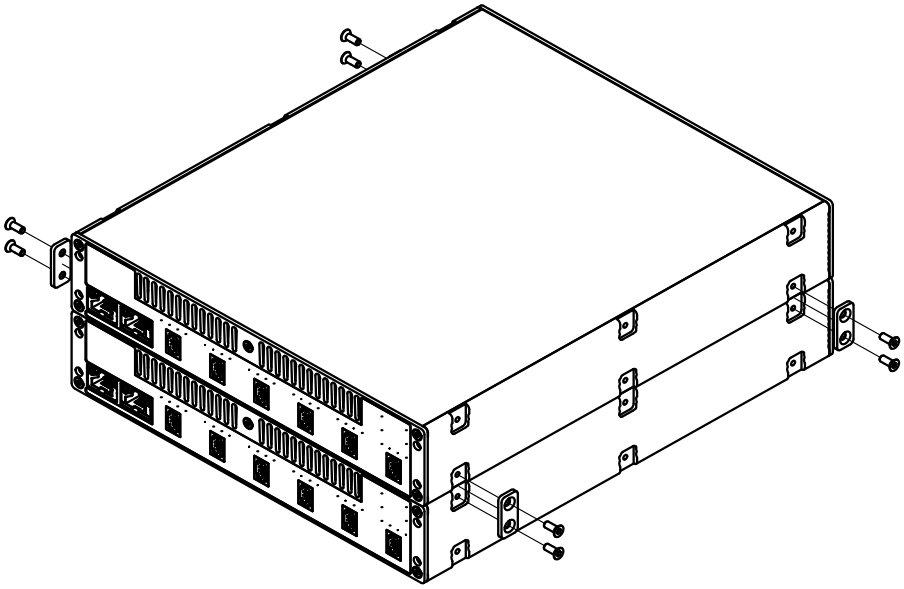
Note

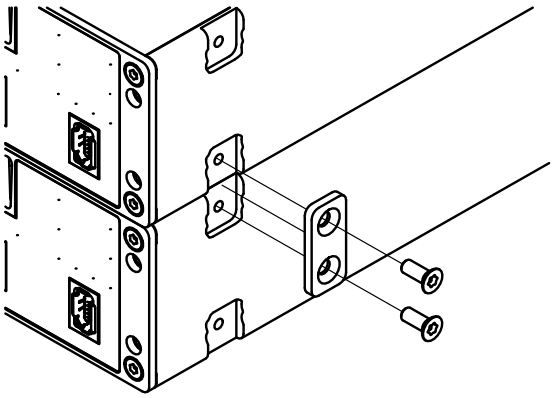
- Elements may only be used for Vector devices in the Vector System Housing form factor.
- Always use both supplied elements and associated screws for mounting.
- For systems up to max. 20 kg.
- In case of uncertainty and questions, please contact qualified personnel.
- Tightening torque: hexagon socket 3 mm, 1.7 Nm.

7.11.8 VSH Connection Kit Vertical

Technical data

Description	Four small mounting plates for connecting two devices on top of each other (vertically). Mounting is done on the left and right side of the housing.
Part number	05152





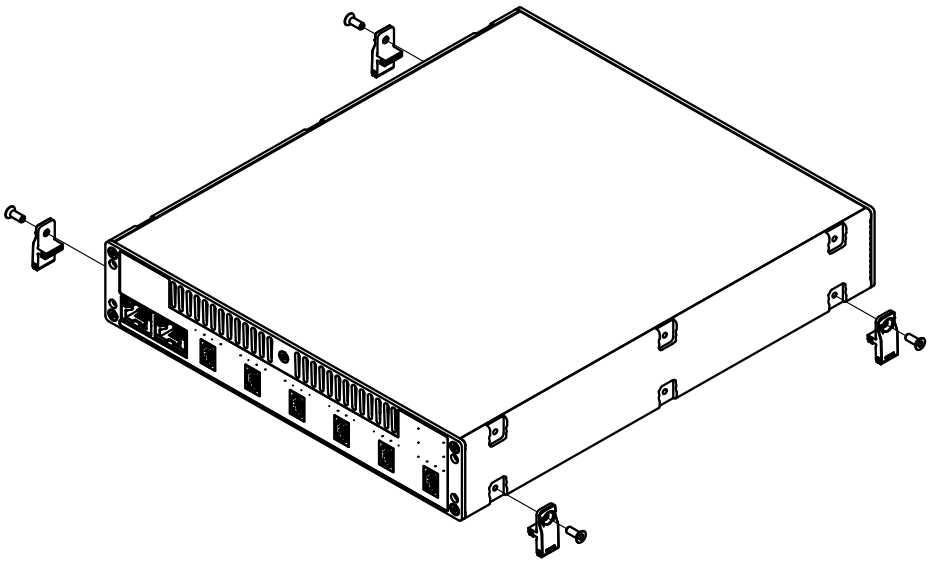
Note

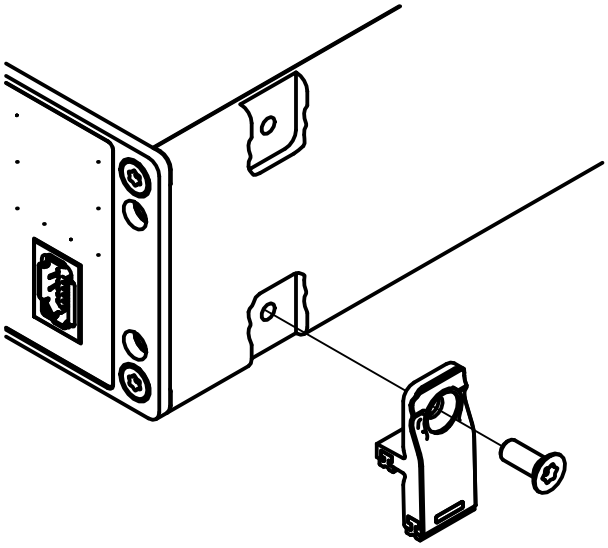
- Elements may only be used for Vector devices in the Vector System Housing form factor.
- Always use all supplied elements and associated screws for mounting.
- For systems up to max. 20 kg.
- In case of uncertainty and questions, please contact qualified personnel.
- Tightening torque: Torx T10, 1.2 Nm.

7.11.9 VSH Equipment Foot Kit

Technical data

Description	Four spare mounting feet.
Part number	05150





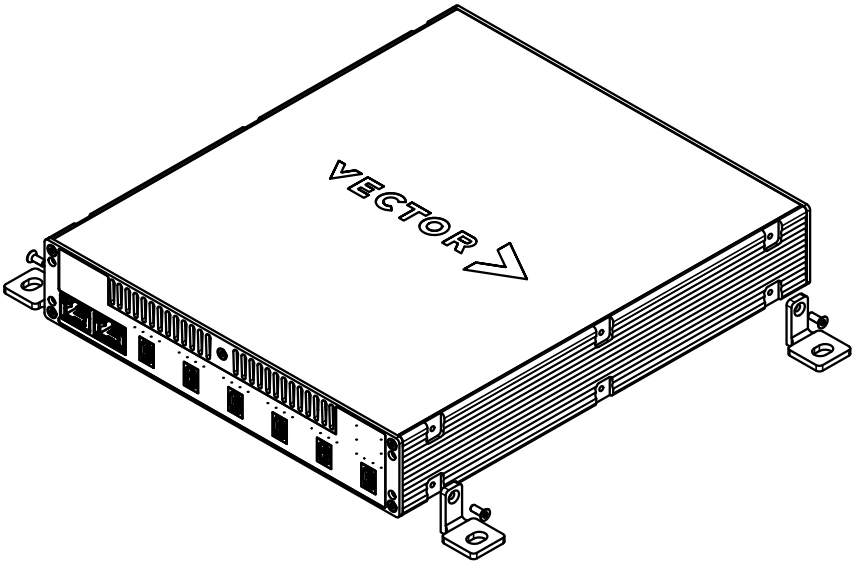
Note

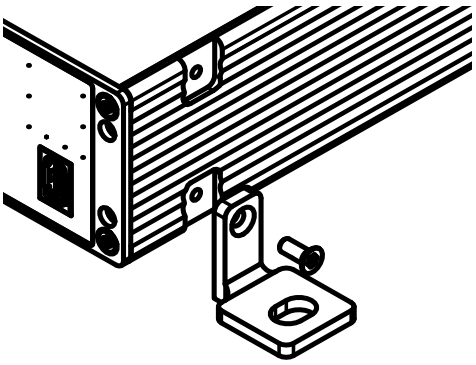
- Elements may only be used for Vector devices in the Vector System Housing form factor.
- Always use all supplied elements and associated screws for mounting.
- For systems up to max. 20 kg.
- In case of uncertainty and questions, please contact qualified personnel.
- Tightening torque: Torx T10, 0.5...0.9 Nm.

7.11.10 VSH Mounting Flange Kit

Technical data

Description	Four mounting brackets. Allows fixing the device to a suitable surface.
Part number	05147





Note

- Mounting brackets may only be used for Vector devices in the Vector System Housing form factor.
- Always use all 4 brackets incl. screws for mounting.
- For systems up to max. 20 kg.
- For fixing the mounting brackets to the floor, wall or ceiling, suitable screws must be selected to ensure permanent fixing according to the expected load (not included in the scope of delivery).
- In case of uncertainty and questions, please contact qualified personnel.
- Tightening torque: Torx T10, 1.2 Nm.

8 Hardware Integration Guide

In this chapter you find the following information:

8.1 Introduction **689**
8.1.1 Getting Started 689

8.2 POD Usage Guideline **690**

8.3 Interfaces and Bus Systems **691**
8.3.1 Introduction 691
8.3.2 AUR (Aurora) 692
8.3.3 DAP 694
8.3.4 HSCT/RHSIF/Zipwire 696
8.3.5 HSSL 698
8.3.6 JTAG 699
8.3.7 RIF (LVDS Radar Interface) 701
8.3.8 SWD 703
8.3.9 SYSTEM 705

8.4 Layout Recommendations **707**
8.4.1 Introduction 707
8.4.2 PCB Layer Stack-Up 707
8.4.3 Impedance Matching 708
8.4.4 Routing of Traces 709
8.4.5 Reference Planes 711
8.4.6 Differential Signals 712

8.5 Galvanic Isolation and Ground Concept **713**
8.5.1 Understanding the VX1000 Ground Concept 713
8.5.2 Grounds of a VX1000 Setup 714
8.5.3 Grounds inside a VX Base Module 718
8.5.4 Grounds Inside a POD 719
8.5.5 Disconnecting an ECU from its Power Supply 720

8.6 Mechanical Integration **724**
8.6.1 Introduction 724
8.6.2 Basic Thoughts about Integrating a POD 725
8.6.3 Heat Dissipation of a POD 726

8.7 Wiring **728**
8.7.1 Cable Pigtail 728
8.7.2 VX1381 HSSL2 Pigtail Installation 729

8.8 Advanced Topics **732**
8.8.1 Aurora Reference Clock 732

8.1 Introduction

The Hardware Integration Guide shares supplemental information about hardware related topics and guides through main aspects of integrating VX1000 components into an ECU.

As the technical solutions covered by the VX1000 system vary in many aspects, this guide can only give an introduction to the topics to be considered. Details in the context of a specific setup should be discussed with your Vector contact.

Note
Before starting any development activity in this context, get in contact with Vector (via VXsupport@vector.com) to find the appropriate solution for your use case and select the suitable VX1000 components.

8.1.1 Getting Started

This guide covers following aspects to be considered when integrating VX1000 components into an ECU:

- Read the POD usage guideline to avoid damage when handling PODs.
- Find information about the electrical interfaces used to communicate between the target ECU and the VX1000 System.
- Follow our PCB layout recommendations to ensure signal integrity and to design a robust interface.
- Get familiar with the VX1000 ground concept and avoid damages due to typical errors.
- Develop a mechanical concept to integrate VX1000 components.
- Find information about wiring the target ECU with the VX1000 system.
- Get help on advanced topics for specific use cases.

8.2 POD Usage Guideline

WARNING!
VX1000 Hardware is not designed for high voltage (HV) environments, unless explicitly stated otherwise! If a solution for a HV environment is required, please contact VXsupport@vector.com for guidance.

Note
Exposed POD PCBs are sensitive to ESD and mechanical stress and should always be handled with care.

Please mind the following hints to protect PODs from damage and increase their lifetime.

- Do not plug or unplug the electrical connection between POD and ECU while either is powered! The interface is not designed to be hot-pluggable.
- Always ensure that the POD target interface is correctly configured before physically connecting POD and ECU! Wrong settings like the target interface voltage can damage either side.
- Always treat Adapter Flex-PCBs and PODs carefully during assembly of measurement setups, like when attaching to or detaching from an ECU.
- Neither Flex-PCB nor POD are designed for frequent mating cycles with the ECU connector.
- When handling the Flex-PCB ...
  - Avoid contact with sharp edges!
  - Avoid sharp bending!
  - Do not pull the flexible part for unplugging – use the rigid part instead!
- When handling exposed PCBs like the POD PCB, always work in an ESD-protected environment.

Reference
For further questions contact VXsupport@vector.com.

8.3 Interfaces and Bus Systems

8.3.1 Introduction

This chapter gives general information about interfaces and bus systems that are supported by the VX1000 PODs to communicate with a target MCU.

Besides debug and trace interfaces, additional pins for common control and status information are described in the SYSTEM section.

The following sections comprise short descriptions of the interface with main electrical aspects. Especially for high-speed interfaces there are layout recommendations given that should be followed in the target design.

Note
Vector strongly recommends being involved in a partial HW design review when developing an interface to VX1000 components for an ECU. For more information, please contact VXsupport@vector.com.

WARNING!
The information and recommendations about interfaces provided in this chapter are kept common for all supported microcontroller families.

For dedicated facts and figures the microcontroller's datasheet must be referenced.

The pin tables given in the next sections are using following abbreviations:

Legend

Column	Description
Signal	Signal name
Dir	Pin direction (see table below for details)
Need	Need for the signal within the Interface group (see table below for details)

Pin direction	Description
IN	Input of the ECU
OUT	Output of the ECU
IN/OUT	Input/Output of the ECU
PWR	Power supply
DNC	Do not connect

Need	Description
m	mandatory
(m)	mandatory under certain conditions (see Details)
o	optional

8.3.2 AUR (Aurora)

8.3.2.1 Description

The Aurora (AUR) interface is a Gigabit trace interface that can be used by the VX1000 for high performance DAQ over data trace with minimum impact on ECU runtime. It is implemented by unidirectional differential data lanes in combination with a differential reference clock lane. The Aurora interface is supported by various microcontroller families and vendors, and may require emulation devices.

- Target microcontroller: Dedicated devices of Renesas RH850, Infineon Aurix, NXP/ST PowerPC family.
- Data rate: up to 5 Gbit/s per lane on up to 4 lanes.
- Data frequency: up to 2.5 GHz
- Reference clock frequency: dependent on target PLL (e.g. 100MHz, 125 MHz, 156.25 MHz, 1.25 GHz)
- Voltage level: 1.5-V PCML

8.3.2.2 Physical Interface

The Aurora electrical interface is implemented by unidirectional differential lanes using the 1.5V-PCML standard. It consists of up to four data lanes transferring data from the target to the VX1000 whereas the reference clock lane is provided by the VX1000 to the target. For Infineon targets, the optional CRC_ERR signal can be used by the VX1000 to request a retransmit for faulty data. The maximum lane number and speed and the reference clock frequency is dependent on the target microcontroller. The signals below are given as seen from the target:

Signal	Dir	Need	Description
AUR TX[0..3] p	OUT	m	Differential Aurora data output, 1, 2 or 4 data lanes.
AUR TX[0..3] n			
AUR CLK p	IN	m	Differential Aurora reference clock input
AUR CLK n			

Dedicated target microcontrollers provide additional signals for handling the Aurora interface. These signals may be mandatory for the VX1000 if they exist on a specific target. Following table summarizes the signals as seen from the target:

Signal	Dir	Need	Description
CRC_ERR	IN	o	CRC-Error input, triggers the target to retransmit data if the VX1000 has seen a fault during reception. Only available on Infineon TriCore targets.
nMSYN	IN	m	Aurora synchronization request. Only available on Renesas RH850 targets.
nAUR_RST1	IN	m	Aurora domain reset signal. Only available on Renesas RH850 targets.
nAUR_RST2	IN	m	Aurora domain signal isolation. Only available on Renesas RH850 targets.

8.3.2.3 VX1000 Specific

- VX1000 supports 8b10b Aurora coding, whereas 64b66b coding is not supported
- The number of lanes and maximum data rate may be restricted by the target microcontroller and the used POD. This has to be checked within the dedicated VX1000 Setup.
- In VX1000 POD designs the Aurora data and clock lanes are AC coupled
- In dedicated VX1000 Setups it is necessary to use an EEK head for providing the reference clock, or to implement a separate reference clock in the target design. See chapter Aurora Reference Clock for details.

8.3.2.4 Design Recommendations

- Common rules for high-speed designs apply.
  See section Layout Recommendations for more information.
- Route the Aurora lanes impedance matched at 100 Ω differential
- Route the traces within one lane length matched (<1 mm)
- Route the data lanes length matched (<1 mm)
- There is no need to match the length of the Aurora clock lane with the data lanes
- Avoid gaps in the reference plane at all costs
- Minimize the number of vias in the Aurora lanes
- Avoid stubs in the Aurora lanes
- Avoid long parallel traces to minimize crosstalk
- Follow the layout recommendations of the microcontroller vendor

Note
More detailed information about the Aurora interface and its implementation can be obtained from the microcontroller vendor.

WARNING!
Since the Aurora interface is a high-speed interface with frequencies in the GHz range, the hardware design has a high impact on signal quality and link stability. Special care must be taken when implementing this interface in a target design.

Note
Vector strongly recommends being involved in a partial HW design review when developing an interface to VX1000 components for an ECU. For more information, please contact VXsupport@vector.com.

8.3.3 DAP

8.3.3.1 Description

Infineon TriCore microcontrollers provide the device access port (DAP) as a debugging interface. Usually it shares the pins with the JTAG interface but offers a higher bandwidth and other benefits regarding reliability and performance.

Different modes at various speeds and pin count are supported by the DAP interface. For more information please refer to Infineon application notes on DAP interface usage.

- Target microcontroller: Infineon TriCore family.
- Data rate: up to 160 Mbit/s (realistic measure, dependent on implementation and DAP-mode)
- Bus frequency: up to 160 MHz
- Several modes and signal count supported.
- Voltage level: standard I/O voltage (e.g. 3V3, 5V0)

Note
This section exclusively describes the DAP interface of Infineon microcontrollers. It should not be confused with the debug access port of ARM devices based on other physical interfaces. For more details on ARM debug interfaces, see section SWD.

8.3.3.2 Physical Interface

The physical DAP interface has several single-ended signals for clock and data, driven at the target standard I/O voltage. The clock is always provided from the VX1000 to the target. Depending on the used microcontroller the DAP interface must be enabled by pulling DAPEN high.

The DAP interface consists of the following signals as seen from the target, showing both the standard mode (2-pin mode) and the wide mode (3-pin mode):

Signal	Dir	Need in Mode		Description
		2-Pin	3-Pin	
DAP0	IN	m	m	Clock input
DAP1	IN/OUT	m	m	Bidirectional data input/output
DAP2	IN/OUT	-	m	Bidirectional data input/output (only used in 3-pin mode)
DAP3	-	-	-	Not used
DAPEN	IN	(m)	(m)	Enable input. On dedicated devices, the DAP is enabled by default. Other devices need this signal to enable DAP in the target.

8.3.3.3 VX1000 Specific

- The VX1000 supports DAP interfaces of Infineon Aurix generations TC1x, TC2x, TC3x and TC4x
- Depending on the target and the VX1000 Setup, either the “Standard Mode" (2-pin Mode) or the “Wide Mode" (3-pin Mode) can be used
- The "Unidirectional Wide Mode" (UWM) is not supported by the VX1000
- The VX1000 PODs implement a pullup resistor at DAPEN to VREF with a value of 4.75 kΩ.
  Hence, in order to use the DAPEN pin, connecting the VREF pin is mandatory.

8.3.3.4 Design Recommendations

- See section Layout Recommendations for general notes
- Route the DAP clock and data signals impedance matched at 50 Ω
- Keep the DAP traces as short as possible (overall length <100 mm)
- Keep the DAP traces equal in length (difference <3 mm)
- Route DAP0 with highest priority and avoid layer changes
- If possible, separate DAP0 using a GND-guard from other signals to avoid crosstalk. Make sure the GND-guard is well connected to the signal ground over the entire length
- Do not apply pullup/pulldown resistors on the DAP clock and data signals
- Terminate the DAP data signals (DAP1, DAP2) with a series resistor near the target microcontroller to 50 Ohm source impedance
- Minimize the number of vias in the DAP traces
- Avoid stubs in the DAP traces
- Avoid long parallel traces to minimize crosstalk
- Follow the layout recommendations of the microcontroller vendor

WARNING!
A design can be optimized for either DAP or JTAG. Compromises may result in a non-functioning DAP interface or performance degradation. To achieve high DAP clock rates, an optimized design and terminated data signals are essential.

Note
More detailed information about the DAP interface and its implementation can be obtained from Infineon.

Note
Vector strongly recommends being involved in a partial HW design review when developing an interface to VX1000 components for an ECU. For more information, please contact VXsupport@vector.com.

8.3.4 HSCT/RHSIF/Zipwire

8.3.4.1 Description

HSCT/RHSIF/Zipwire are high-speed serial interfaces for chip-to-chip comunication, which are largely compatible with each other:

- HSCT (High-Speed Communication Tunnel) of Infineon TriCore MCUs
- Zipwire (also called the LFAST/SIPI) of ST/NXP MCUs
- RHSIF of Renesas MCUs

It is designed for communication between two devices via a high-speed serial interface.

The throughput of this interface can reach up to 320 Mbit/s and thus offers a considerably higher data rate than for example JTAG.

- Target microcontroller: Infineon TriCore family, ST/NXP microcontroller.
- Data rate: up to 320 Mbit/s.
- Data frequency: 160 MHz.
- Reference clock frequency: 10 MHz, 20 MHz.
- Voltage level: standard I/O voltage single-ended clock, LVDS data lanes.

8.3.4.2 Physical Interface

The data signals are implemented as unidirectional full duplex TX/RX LVDS lanes for supporting stable high-speed data. The clock of this interface is a single-ended reference clock with a relatively low frequency (10 MHz / 20 MHz).

The HSCT/Zipwire/RHSIF interface consists of the following signals as seen from the target:

Signal		Dir	Need	Description
HSCT	**Zipwire**			
SYSCLK_OUT	REFCLK	IN/OUT	m	Single-ended reference clock, either input or output (depending on setup; preferred/recommended: OUT)
RXDP	SIPI_RXP	IN	m	Differential serial data received from the VX1000
RXDN	SIPI_RXN			
TXDP	SIPI_TXP	OUT	m	Differential serial data transmitted to the VX1000
TXDN	SIPI_TXN			

8.3.4.3 VX1000 Specific

- The VX1000 is always acting as the HSCT master, owning the link.
- In order to support a target presence detection, the target should provide the clock to the VX1000.

8.3.4.4 Design Recommendations

- Common rules for high-speed designs apply,
  see section Layout Recommendations for more information
- RX and TX data lanes should be routed impedance matched at 100 Ω differential.
- The single-ended reference clock should be routed impedance matched at 50 Ω .

Note
More detailed information about the HSCT/Zipwire/RHSIF interface and its implementation can be obtained from the microcontroller vendor.

WARNING!
The implementation of a HSCT/Zipwire/RHSIF interface might differ for different targets and use cases, thus it is necessary to clarify the details for a dedicated setup.
More information on the HSCT/Zipwire/RHSIF interface of a specific customer setup can be obtained from VXsupport@vector.com.

Note
Vector strongly recommends being involved in a partial HW design review when developing an interface to VX1000 components for an ECU. For more information, please contact VXsupport@vector.com.

8.3.5 HSSL

8.3.5.1 Disambiguation

In technical documents of the **Infineon** TriCore family, the term **HSSL** refers to a transport layer module that is used in combination with the HSCT module on the physical layer for a high-speed serial link (HSCT/HSSL).

In contrast, **Vector** uses the term **HSSL** for the Vector-proprietary physical connection and link protocol between a VX145x HSSL POD and a VX1000 Base Module.

To avoid ambiguity in this document,

- **HSSL** always describes the connection between a VX145x HSSL POD and a VX1000 Base Module (see section HSSL Cables),
- while the HSCT/HSSL interface of TriCore microcontrollers is always referred to as **HSCT** (see section HSCT/RHSIF/Zipwire).

8.3.6 JTAG

8.3.6.1 Description

Many microcontrollers offer a JTAG interface not only for boundary scan, but also for debugging purposes that can be utilized in a VX1000 Measurement Setup.

JTAG is implemented as a shift register interface with unidirectional serial communication lines supported by status and control lines.

Commonly it has the following features that may vary depending on the target microcontroller type:

- Target microcontroller: almost any device as JTAG is a common standard.
- TCK frequency: up to 40 MHz.
- Voltage level: standard I/O voltage, (e.g. 3V3, 5V0)

8.3.6.2 Physical Interface

The JTAG interface can be accessed via a test access port (TAP) described below. These signals need dedicated pullup or pulldown resistors to make the interface operational. The standard JTAG-TAP consists of following signals as seen from the target:

Signal	Dir	Need	Description	Design Recommendation
TDI	IN	m	Test Data Input	10 kΩ pullup
TDO	OUT	m	Test Data Output	10 kΩ pullup. Insert a serial 22 Ω resistor between the target and the JTAG-TAP connector and place it close to the microcontroller.
TCK	IN	m	Test Clock	Place a termination at the far end of the clock chain with a serial resistor/capacitor combination of 68 Ω and 100 pF to GND
TMS	IN	m	Test Mode Select	10 kΩ pullup
nTRST	IN	(m)	Test Reset Activates the JTAG interface. Mandatory on some targets.	10 kΩ pulldown
JCOMP	IN	(m)	Same functionality as nTRST. Only available on some PowerPC devices.	
nRDY	OUT	o	Ready handshake signal. Recommended for optimized performance. Only available on some PowerPC devices.	
RTCK	OUT	o	Return Test Clock Recommended for optimized performance. Only available on Texas Instruments TMSx70.	

8.3.6.3 VX1000 Specific

- JTAG voltage and clock rate can be set in the VX1000 settings and must match the target design
- VX1000 specific design recommendations may differ from the ones of the microcontroller vendor and are summarized in the table above. If in doubt contact VXsupport@vector.com for clarification.

WARNING!
For Infineon TriCore microcontrollers, instead of JTAG, the more powerful DAP interface is supported.

8.3.6.4 Design Recommendations

Due to the relatively low bus frequency, the JTAG interface is not treated as a high-speed interface. Regardless, general layout recommendations apply:

- See section Layout Recommendations for general notes
- Avoid long parallel traces for less crosstalk
- Follow the recommendations for pullup/pulldown resistors mentioned in the table above and compare to the recommendations given in the microcontroller datasheet
- Follow the layout recommendations of the microcontroller vendor

WARNING!
Although it is possible to link several JTAG devices, commonly known as “boundary scan” chain, this is not possible within a VX1000 Setup. The connection between VX1000 and the target must be implemented as an exclusive connection while the VX1000 is being used.

Note
More detailed information about the JTAG interface and its implementation can be obtained from the microcontroller vendor.

Note
Vector strongly recommends being involved in a partial HW design review when developing an interface to VX1000 components for an ECU. For more information, please contact VXsupport@vector.com.

8.3.7 RIF (LVDS Radar Interface)

8.3.7.1 Description

The Radar Interface (RIF) as described in this context is a high-speed LVDS interface used to transfer raw radar samples from a radar frontend chip (MMIC) to the processing controller. In one-chip radar solutions this interface can also be realized as a debug interface of the SOC.

- Target frontends: Infineon, Texas Instruments and other MMICs and SOCs.
- Number of lanes: configurable 2 or 4 data lanes.
- Data rate: up to 900 Mbit/s per lane.
- Bus frequency: up to 450 MHz.
- Voltage level: LVDS.

8.3.7.2 Physical Interface

The physical RIF interface is realized as an unidirectional high-speed LVDS interface consisting of two or four lanes for data, a bit clock and a frame signal.

The following table describes the RIF signals as seen from the target:

Signal	Dir	Need	Description
RIF DATA[0..3] p	OUT	m	Differential RIF DDR data output. Configurable bandwidth supporting 2 or 4 data lanes.
RIF DATA[0..3] n			
RIF CLK p	OUT	m	Differential RIF bit clock output
RIF CLK n			
RIF FRAME p	OUT	m	Differential RIF frame signal output
RIF FRAME n			
nLVDS_ACTIVE	OUT	(m)	LVDS active signal (low active). If the RIF CLK is undefined, like during MMIC sleep cycles, this signal is mandatory to disable RIF processing in the POD.

8.3.7.3 VX1000 Specific

- The number of lanes and maximum data rate may be restricted by the target radar frontend, microcontroller, and the used POD. This has to be checked within the dedicated VX1000 Setup.
- The VX1000 supports both, sniffing RIF data on the connection between MMIC and MCU, and a direct connection between a SOC and the POD. For ensuring signal integrity of the RIF interface in both cases, this connection should be implemented in close consultation with VXsupport@vector.com.
- In case the MMIC disables the LVDS interface during a sleep cycle resulting in undefined levels on the RIF CLK lane, it is mandatory to disable RIF processing in the POD using the nLVDS_ACTIVE signal.

8.3.7.4 Design Recommendations

- Common rules for high-speed designs apply.
  See section Layout Recommendations for more information.
- Route the LVDS lanes impedance matched at 100 Ω differential
- Route the traces within one lane length matched (<1 mm)
- Route the RIF lanes length matched between each other (<1 mm)
- Avoid stubs in the RIF lanes
- Avoid long parallel traces to minimize crosstalk
- Follow the layout recommendations of the microcontroller/MMIC vendor

Note
More detailed information about the RIF interface and its implementation can be obtained from the microcontroller/MMIC vendor.

Note
Vector strongly recommends being involved in a partial HW design review when developing an interface to VX1000 components for an ECU. For more information, please contact VXsupport@vector.com.

8.3.8 SWD

8.3.8.1 Description

The Serial Wire Debug (SWD) is a standard debug interface for ARM-based microcontrollers that implement a serial debug wire port (SW-DP).

SWD offers the test and debug functionality known from the JTAG interface and adds advanced functionality. It replaces the traditional physical JTAG interface with a 2-wire interface and is more robust and error tolerant due to a packet-based communication. In the target microcontroller, the SWD port can be combined with a JTAG port and is then referred to as the Serial Wire / JTAG Debug Port (SWJ-DP).

- Target microcontroller: ARM based devices with SWD/SWJ port, see section Supported Microcontrollers
- Bus frequency: up to 125 MHz, depending on target microcontroller and VX1000 settings.
- Voltage level: standard I/O voltage (e.g. 3V3, 5V0)

8.3.8.2 Physical Interface

The physical SWD interface is a 2-wire synchronous serial interface. The clock is always provided from the VX1000 to the target. The data is transferred via a bidirectional data signal.

The signals below are given as seen from the target:

Signal	Dir	Need	Description
SWDIO	IN/OUT	m	Bidirectional data, on SWJ ports shared with TMS
SWCLK	IN	m	Clock input, on SWJ ports shared with TCK

8.3.8.3 VX1000 Specific

- The VX1000 supports a point-to-point connection to one target microcontroller. Simultaneously connecting multiple devices (multi-drop) is not supported.
- Selected ARM based devices from ST provide a unidirectional 3-pin SWD mode. This mode is not supported by the VX1000.
- The SWD interface optionally provides a SWO debug/trace output signal. This signal is not supported by the VX1000.

8.3.8.4 Design Recommendations

- See section Layout Recommendations for general notes
- Route the SWD clock and data signals impedance matched at 50 Ω
- Keep the SWD traces as short as possible (overall length <100 mm)
- Keep the SWD traces equal in length (difference <3 mm)
- Route SWCLK with highest priority and avoid layer changes
- If possible, separate SWCLK using a GND-guard from other signals to avoid crosstalk. Make sure the GND-guard is well connected to the signal ground over the entire length.
- Apply a pullup resistor on the SWDIO data signal
- Use a pullup/pulldown resistor to pull SWCLK to a defined state
- Terminate the SWDIO data signal with a series resistor near the target micro-controller to 50 Ohm source impedance
- Minimize the number of vias in the SWD traces
- Avoid stubs in the SWD traces
- Avoid long parallel traces to minimize crosstalk
- Follow the layout recommendations of the microcontroller vendor

Note
More detailed information about the SWD interface and its implementation can be obtained from the microcontroller vendor.

Note
Vector strongly recommends being involved in a partial HW design review when developing an interface to VX1000 components for an ECU. For more information, please contact VXsupport@vector.com.

8.3.9 SYSTEM

The signals documented in this section are used by PODs for common status and control tasks. Dependent on the use case some of the signals are mandatory, whereas others are optional.

All the signals mentioned in the following are working on the same standard voltage level configured in the POD, with the exception of nRESET, which is always 5 V tolerant.

8.3.9.1 VREF

Target voltage reference

- Target voltage reference
- Analog input of the POD
- Mandatory signal

VREF as the target reference voltage is to be connected to the target I/O voltage used for POD communication.

Main purpose of this signal is to enable a target presence detection in the POD. This signal is mandatory for some target interfaces, and optional for others, although always recommended.

In PODs that support a flexible target voltage (e.g. 3V3/5V), this signal is commonly implemented as an analog input, enabling a target voltage supervision to avoid destructive target voltages.

8.3.9.2 nRESET

Target hardware reset

- Target hardware reset
- Low active input of the POD, optional open-drain output
- Mandatory signal

nRESET must be connected to the target's hardware reset signal (nRESET, nPORST, etc.). This pin is implemented as a low active signal to be compliant to the corresponding MCU pin.

Main function provided is to enable a basic target reset detection by the POD, thus this pin is mandatory in all designs. Optionally, this pin can be used as a POD output to apply a reset to the target. For this purpose, the output is implemented with an open drain output.

8.3.9.3 RSTOUT

Target software reset

- Target software reset
- Digital input of the POD
- Optional signal

RSTOUT can optionally be used to let the POD detect a target software reset and is hence an input of the POD.

It is recommended to connect this signal to the dedicated software reset output pin of the MCU (like ESR0/1 for Infineon Aurix) and enable the corresponding functionality in the microcontroller.

For features like measurement over reset, this pin becomes mandatory.

8.3.9.4 WDGDIS

Watchdog disable

- Watchdog disable
- Digital output of the POD
- Optional signal

WDGDIS is an optional output of the POD that can be used to disable the watchdog functionality inside a target ECU.

Using this signal requires a dedicated pin in the target design, usually provided by the PMIC, that is used to disable the watchdog functionality by pulling the pin high or low. The desired behavior of this signal can be set in VXconfig.

In certain VX1000 setups it is possible to provide a clock signal on this pin to emulate a watchdog input. This has to be checked within the dedicated setup.

8.4 Layout Recommendations

8.4.1 Introduction

Most of the interfaces used today for communication between the target ECU and the VX1000 system are high-speed interfaces that require special attention in the PCB layout design.

The following are general notes that can help achieve high signal integrity for implementing an electrically robust interface.

Note
Vector strongly recommends being involved in a partial HW design review when developing an interface to VX1000 components for an ECU. For more information, please contact VXsupport@vector.com.

8.4.2 PCB Layer Stack-Up

If a high-speed interface is to be integrated into an ECU design, it is advisable to select the PCB layer stack-up accordingly. This also applies if the interface is only used for measurement or debugging purposes.

A high-speed layer stack-up enables the planning of configurations for routing signals with matched impedance within defined tolerances, as well as the selection of a suitable via technology.

Reference
Contact your Electronic Manufacturing Services (EMS) for finding an appropriate PCB layer stack-up for a specific design.

8.4.3 Impedance Matching

Impedance matching is an important part of a high-speed design. When the impedance of the transmitter, receiver, and transmission path of a signal is matched, signal reflections are reduced, resulting in high signal integrity.

The impedance to be met depends on the I/O standard used by the specific interface. An appropriate PCB layer stack-up defines possible routing layers together with the corresponding reference planes as well as the geometry of an impedance matched trace.

- Make sure that the desired impedance can be achieved with the selected PCB layer stack-up.
- Ensure that both, the transmitter and receiver of a signal have the required impedance. It may be necessary to place external components (for example resistors) in the design to reach the correct impedance.
- If possible, avoid impedance discontinuity along the high-speed signal trace caused by deviating geometry (for example test pads, solder pads, or vias).
- If a pad in a signal trace is causing an impedance discontinuity, consider sparing out the reference layer underneath the pad for reducing the impedance disruption. This should be done with care and requires further analysis in the design.

Note

There may be differences in the relevant literature, but a general recommendation for designs related to the VX1000 is:

- For single ended signals, a line impedance of 50 Ohm (±10%) shall be fulfilled.
- For differential signals, a coupled (differential) impedance of 100 Ohm (±10%) shall be fulfilled.

8.4.4 Routing of Traces

8.4.4.1 General Notes

- Do not auto-route high-speed signals.
- Route high-speed signal traces with high priority before other traces are routed.
- High-speed signal traces should not be routed with sharp bends or 90° bends. Use 45° bends or arcs instead.
- Try to minimize the number of vias in high-speed signal traces and use appropriate via technology to avoid stubs.
- Avoid test pads in high-speed signal traces, as they may cause impedance discontinuity and thus signal reflections.
- The return current path must be close to the signal path to keep the current loop as small as possible.
- When changing the routing layer of a high-speed signal, ensure that also the return path has one or more vias to change the reference plane (see section Reference Planes).
- Ensure that the ground reference plane is a solid low impedance ground reference.

8.4.4.2 Microstrip/Stripline

It is highly recommended to implement high-speed signal traces impedance-matched in a stripline or microstrip configuration. In a stripline configuration the signal trace is routed on an inner layer surrounded by two reference planes. In a microstrip configuration the signal can also be routed on an outer layer and has one reference plane.

- Whenever possible, use stripline as it has better EMI performance than microstrip.
- Use ground traces or ground areas running alongside the signal traces, if possible.
- Ground traces or areas should have as many stitching vias as possible to guarantee a tight coupling to the system ground.

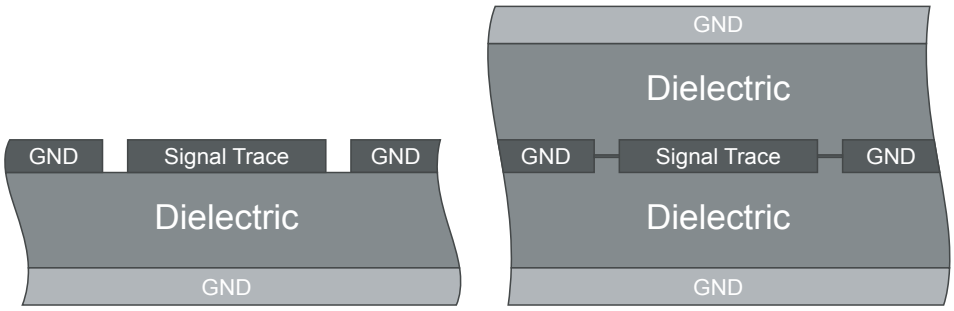


Figure 275: Microstrip (left) versus stripline (right) with sideways GND (sectional view)

8.4.4.3 Trace Distance

When traces are routed in parallel over a longer distance, this facilitates crosstalk, which should be avoided for high signal integrity.

- Avoid parallel routed traces on adjacent layers because they have higher coupling than parallel traces on the same layer.
- For clock signals and impedance controlled signals in general, keep a distance to the neighboring signals of at least 3 times the trace width.

8.4.4.4 Stubs

Stubs in high-speed signal traces must be avoided as they may cause reflections or act as antennas at high frequencies or high slew rates.

- Select an appropriate via technology when changing layers of a high-speed signal. Unused parts of a via may be a stub as well.
- Stubs can also occur on an automatically filled ground planes as well as on loosely coupled ground traces. Make sure that these ground areas closely connected to the system ground using vias.
- In case of connecting a test pad, measuring point or additional pins with a specific signal, prefer a loop line over a stub:

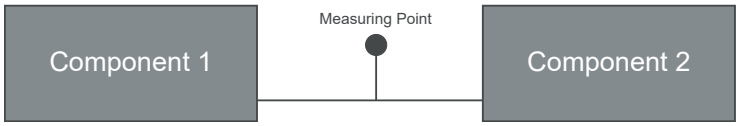


Figure 276: Critical connection of a measuring point via a stub

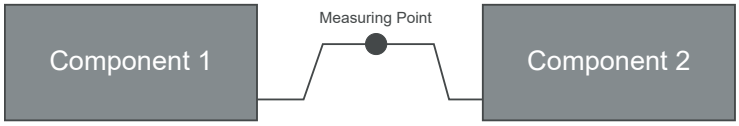


Figure 277: Good connection of a measuring point via a loop line

8.4.5 Reference Planes

High-speed signals need a signal return path close to the signal trace that is usually implemented as a coupled reference (ground) plane. A gap in the reference plane leads to a discontinuity in the impedance of the trace that causes reflections and reduces the signal integrity.

Therefore:

- Avoid routing traces over a split reference plane.
- Ensure that the reference plane is a solid ground reference. A loosely coupled reference may float at higher frequencies.
- If the reference plane cannot be implemented with a ground plane, a supply voltage plane can also serve as a reference. In this case, the supply voltage plane requires good capacitive coupling to ground to achieve low impedance in the desired frequency range.
- When a high-speed signal changes the routing layer and uses a different reference plane, make sure that these reference planes are as well connected with vias near the changing signal trace.
- If two reference planes are contiguous but on adjacent planes, they should be constructed overlapping and not with a gap.
- If the splitting of reference planes cannot be avoided, use a capacitor near the signal path as an RF return path between the split planes as shown below.

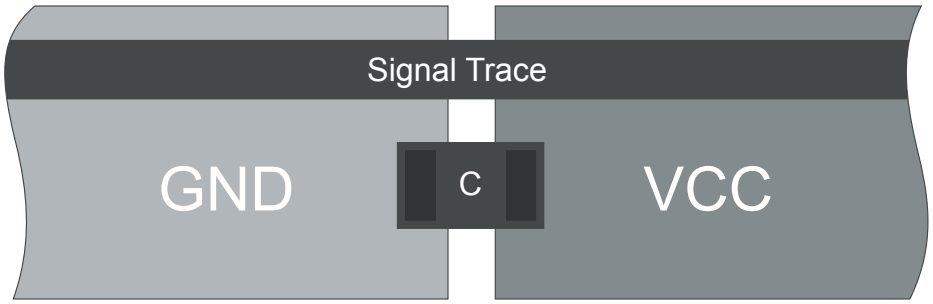


Figure 278: Coupling capacitor used to bridge a plane split (top view)

WARNING!
In case of separate GND or VCC planes for analog and digital parts of your design, use the above recommendation with extreme care as it can undermine the intended separation of analog and digital ground or supply layers.

8.4.6 Differential Signals

In general, the recommendations given above for single-ended signals also apply to differential signals. In addition, further recommendations should be observed:

- Route the traces of a differential pair symmetrically.
- Traces within a differential pair should be length matched. A general recommendation is to keep the length difference of the traces less than 1 mm for a Gbit/s interface.
- A length mismatch between the traces of a differential pair should be compensated as close as possible to the location where it occurred.
- High-speed signal traces should not be routed with sharp bends or 90° bends. Instead, use 45° bends or arcs and compensate the length difference of the traces directly.
- Keep a distance between neighboring signal pairs of at least 3 times the differential trace distance.
- Dependent on the used I/O standard, it can be necessary to match the length of several differential lanes of the same interface.

8.5 Galvanic Isolation and Ground Concept

8.5.1 Understanding the VX1000 Ground Concept

In a VX1000 measurement setup, different electrical grounds exist. To avoid electrical interference and unwanted DC paths in the setup, it is important to understand the ground concept and to analyze the potentials of the different grounds in the entire system.

Principles

The VX1000 ground concept is based on following principles:

- The POD signal ground is identical to the ECU microcontroller signal ground
- There is no galvanic isolation done inside VX1000 PODs
- If galvanic isolation exists in a setup, this is implemented inside the VX1000 Base Module
- Some interfaces of the Base Modules do not provide a galvanic isolation between the POD signal ground and the cable shield

Individual characteristics

Apart from these principles, some characteristics of an ECU with integrated VX1000 components have significant influence on the grounds of a specific setup, for example:

- Are ECU and POD mounted in a conductive housing?
- Is the ECU housing in contact with vehicle ground / chassis?
- Is the POD cable shield connected to the ECU housing?
- Is there any galvanic isolation implemented inside the ECU design?

WARNING!
When the POD is integrated in a metal housing as requested, the signal ground of the target microcontroller can be connected to the ECU housing via the VX1000 components. This may result in different GND potentials than in a setup without VX1000 components.

Since such characteristics are individual for an ECU and also the VX1000 components may include customer specific changes, it is necessary to analyze the ground concept for each setup.

WARNING!
It is strongly recommended to consult VXsupport@vector.com for analyzing a dedicated VX1000 Setup before the implementation is started.

8.5.2 Grounds of a VX1000 Setup

To ensure a stable VX1000 setup, it is essential to consider two main aspects in the context of the ground concept:

DC paths

Direct contact as well as a low resistive connection between different ground potentials can cause undesirable DC current flow in a setup. While the system may function in a normal situation, for example a simple break in the connection of two components can cause severe damage due to an unwanted current flow. The analysis of the possible DC paths can help overcome these undesirable current flows. Typical error cases are discussed in section Disconnecting an ECU from its Power Supply.

RF paths

In addition, the system performance may be degraded by interference at higher frequencies. In contrast to the DC current, these line-conducted interferences do not necessarily require a direct connection between components, but can also use capacitively coupled connections, for example. These connections will be referred to as RF path (radio frequency path) in the following.

In a VX1000 system, high frequencies are used to enable high bandwidth data transmission. Depending on the setup, this may require to consider RF paths in a frequency range up to 10 GHz.

In order to understand the system behavior, the type of connection between the individual components must be analyzed. Components may not be connected at all, or directly connected. They may also have a mainly resistive, capacitive, or inductive coupling, which plays a supporting role especially in RF considerations.

Note
The connection type of the various ground connections has a major influence on the system behavior in the event of an error.

8.5.2.1 Analyzing the Grounds of a VX1000 Setup

To analyze the ground potentials of a particular setup, it is helpful to create a drawing of the major components with their respective grounds. Possible connections between the grounds should be included and further described with a connection type.

Connection type

The abbreviations from the table below can be used to describe the connection types between the individual grounds. A combination of R/L/C can help characterize the main properties of a non-direct connection and find existing RF paths over grounds in the system. To increase the readability of the drawing, direct connections can also be drawn as continuous lines and negligible connections can be omitted.

Sign	Connection Type	Description
N	No connection	No DC-path; parasitic capacitance is negligible up to 10 GHz.
D	Direct connection	Low impedance connection with DC path.
R	Resistive coupling	Via a defined resistor, eventually combined with a capacitive or inductive coupling.
C	Capacitive coupling	Via a defined capacitance or non-negligible parasitic capacitance.
L	Inductive coupling	Via inductors, chokes, ferrite beads, long wires, etc.

Reference
Support for analyzing the ground potentials of a specific VX1000 Setup can be obtained from VXsupport@vector.com.

8.5.2.2 Example VX1000 Setup Grounds

Following figure shows the different grounds of an example setup. Possible connections are marked with numbered circles.

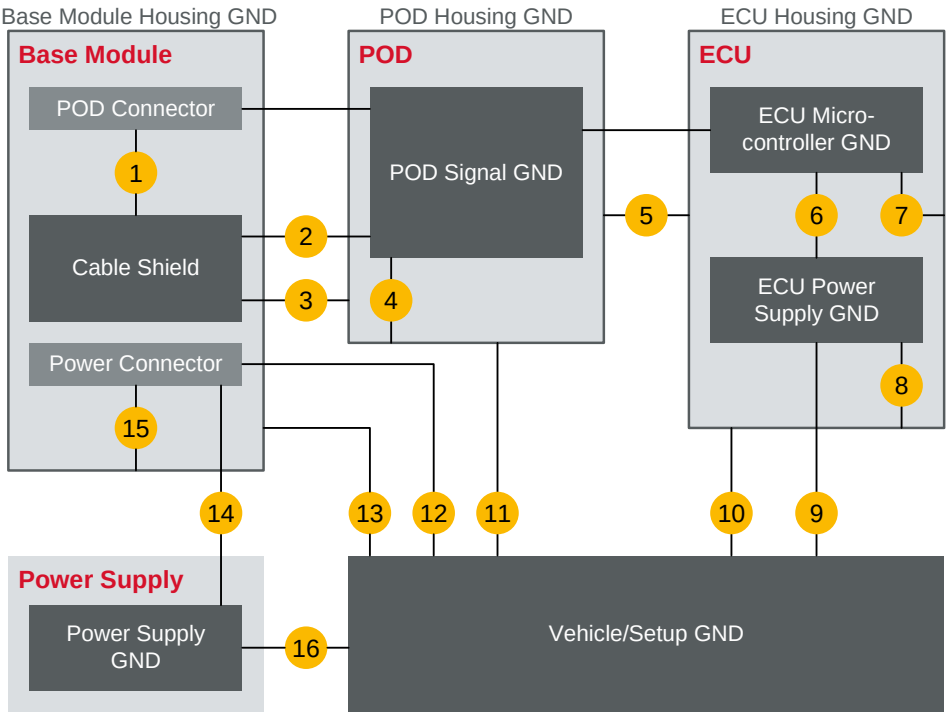


Figure 279: Grounds inside a VX1000 Setup

Signal ground

Connecting a VX1000 POD with an ECU implies that the ECU microcontroller ground is connected to the POD signal ground, which is then routed through the POD cable into the VX1000 Base Module. This means that the ECU may have unwanted ground connections or ground loops which may be hidden by the complexity of the setup.

Housings

In the example above the POD has its own metal housing, such as the VX1544A with sealed enclosure, which is screwed onto the housing of the ECU. If the ECU housing is also conductive, there is a direct connection between both of them (5).

It is important to check whether a housing is in contact with vehicle ground via (10) or (11). This may be intentional, or the result of a fault that should be considered.

It should also be known how the POD signal ground and ECU grounds are connected to the housings via (4) and via (7) or (8). Depending on the ECU design, a ground separation, for example between the ECU microcontroller and the ECU power supply (6), must be considered.

In setups, in which the POD is integrated into the ECU housing, a separate POD housing as well as the connections (5) and (11) do not exist. The cable shield connection (3) would be drawn from the VX1000 Base Module to the ECU housing. In this case it is still important to verify how the POD itself is connected to the ECU housing (4).

Cable shield

When the POD is integrated into the housing, the cable shield shall be connected to the POD signal ground via (2) or a combination of (3) and (4). A connection to the POD housing (3) may exist depending on the used cable gland or panel mount.

In this context it is important to know whether the used VX1000 Base Module internally connects the POD signal ground with the cable shield (1). Section Grounds inside a VX Base Module describes the different options.

Power supply

The VX1000 Base Module can be powered by the vehicle battery and therefore have a direct connection to the vehicle ground (12). Alternatively, a separate power supply or an additional battery may be used to supply the VX1000 Setup and the ground is connected via (14). In this case it is also relevant how the power supply or additional battery is connected to the vehicle ground (16).

VX1000 Base Module housing

If the VX1000 Base Module is screwed to the chassis or its housing has a direct connection to the vehicle/setup ground (13), it should be considered that there are base modules that have no separation between housing ground and power connector ground (15).

Section Grounds inside a VX Base Module describes which VX1000 Base Modules internally connect the power supply ground with the housing ground.

Unwanted DC paths

When the ground connections in the setup are known, wanted and unwanted DC paths can be analyzed.

For example, if a connection between the ECU microcontroller ground and the ECU housing (7) should be avoided, this connection may also be established though (1), (3) and (5).

8.5.3 Grounds inside a VX Base Module

In the context of a VX Base Module, there are several types of electrical grounds, mainly:

Ground Type	Description
Power supply ground	Negative power supply lead of the VX Base Module. Equal to the car battery minus respectively the chassis.
Housing ground	The housing of a VX1000 Base Module. ► VX1060/VX113x: Electrically equal to power supply ground. ► VX1161: Capacitively coupled to power supply ground.
Signal ground	Internal ground of the VX Base Module electronic components. Electrically separated from power supply ground.
Cable shield	Electrical shielding of the cable that connects the VX Base Module and the POD. This includes the jacket of the connector that plugs into the VX Base Module.

Inside the VX1000 Base Module there is an electrical separation between the housing and the internal signal ground and POD cable shield. If necessary, the VX1000 Base Module can be mounted directly to the chassis. In this case note that the housing can be connected internally to the power supply ground (see above).

The internal signal ground of a VX1000 Base Module and the POD cable shield are either connected, or capacitively coupled. **Connected** describes a low resistance DC-connection (<10 Ω), whereas **capacitively coupled** implies a resistance of >100 kΩ parallel to the coupling capacitance.

To make this difference visible at a glance, the POD sockets of a VX1000 Base Module are color-coded:

Color Code	Connection Type	Coupling between Signal Ground and POD Cable Shield
Red	Serial, HSSL	Connected
Blue	Serial	Capacitively coupled
Green	HSSL	Capacitively coupled
Yellow	HSSL2	Capacitively coupled
-	FAKRA/Coax	Capacitively coupled

Reference
More information on grounds of a specific VX1000 Setup can be obtained from VXsupport@vector.com.

8.5.4 Grounds Inside a POD

A POD has the following types of electrical grounds:

Ground Type	Description
Signal ground	Ground of the internal electronics of a POD. Identical to the signal ground of the ECU.
Cable shield	Electrical shielding of the cable that connects the VX Base Module and the POD. This includes the jacket of the connector that plugs into the VX Base Module.

Depending on the type of POD, signal ground and cable shield are either connected by design, or shall be connected at the point where the POD is mounted.

POD Type	Cable Shield/ Signal Ground	Mounting Points
VX1543A	Not connected	Signal ground[1)]
VX1543B/C	Capacitively coupled	Signal ground[1, 2)]
VX1544x	Capacitively coupled	Signal ground[1, 2)]
VX145x	Not connected	Signal ground[1)]
VX146x	Not connected	Signal ground[1)]
VX152x	Capacitively coupled	Signal ground
VX1621A	Not connected	Signal ground
VX1622x	Capacitively coupled	Signal ground[3)]

1) When the POD is installed in a conductive housing, the POD signal ground is connected directly to the housing via the mounting points. There are also different cable shield connection options available for the pigtails (shield connected or not connected to the cable gland). When the shield within the pigtail is connected to the cable gland, the POD signal ground is connected directly to the cable shield via the housing.

2) The capacitive coupling on the POD is shortened.

3) Only the mounting point close to the connectors is capacitively coupled; the other two are unconnected.

Note
The POD should be mounted into a metal housing for mechanical and EMI reasons. The cable shield shall be connected to the POD's signal ground at the point where the POD is mounted.

WARNING!
The electrical connection of the signal ground and cable shield of a POD has to be taken into account in the overall system (see section Grounds of a VX1000 Setup).

8.5.5 Disconnecting an ECU from its Power Supply

If an ECU has to be disconnected from the power supply, it is recommended to disconnect the +12V supply instead of GND to power down the ECU.

In case it cannot be avoided to power down the ECU by disconnecting ground, make sure:

- that the shield of a POD cable does not get in contact with the chassis
- that the metal housing of the POD does not get in contact with the chassis (directly or via the metal housing of the POD)

When disconnecting an ECU under test from power by unplugging ground, care must be taken to avoid the ECU current accidentally flowing through the POD cable or housing.

Whether such an error condition occurs, depends on several factors:

- Are signal ground and cable shield electrically separated inside the VX Base Module?
- Are signal ground and cable shield electrically separated inside the POD?
- Is any blank part of the POD cable in contact with the chassis?
- Is the POD housing in contact with the chassis?

The table below shows the error current matrix. The (error) cases #1 to #6 are depicted in detail on the following pages.

#	Base Module GND & Shield	Cable Shield on VX Side	POD Housing	POD GND & Shield	Result
1	-	-	-	-	Normal operation. No error current.
2	-	-	-	-	ECU disconnected. No error current.
3	Not connected	In contact with chassis	-	Not connected	No error current.
4	Connected	In contact with chassis	-	Not connected	Error current via VX Base Module.
5	-	In contact with chassis	Metal, isolated from chassis	Connected via POD housing	Error current via cable shield.
6	-	-	Metal, in contact with chassis	Connected via POD housing	Error current via POD housing.

8.5.5.1 Normal Operation (#1)

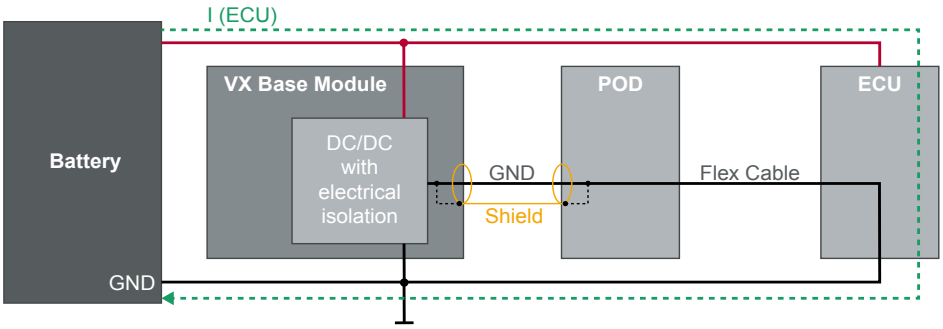


Figure 280: ECU in normal operation

8.5.5.2 Disconnection (#2)

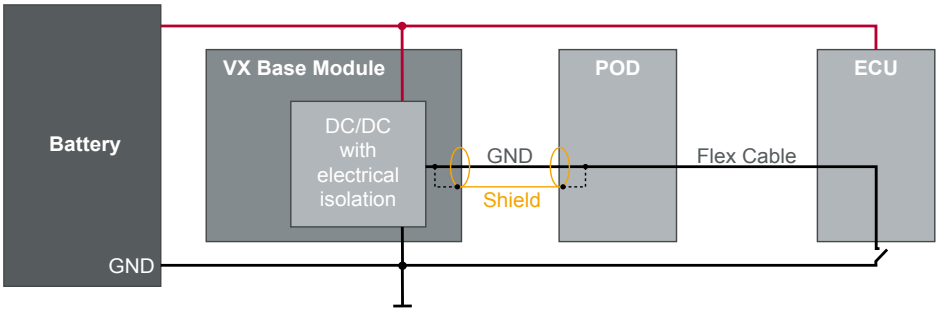


Figure 281: ECU disconnected from GND

8.5.5.3 Short (#3)

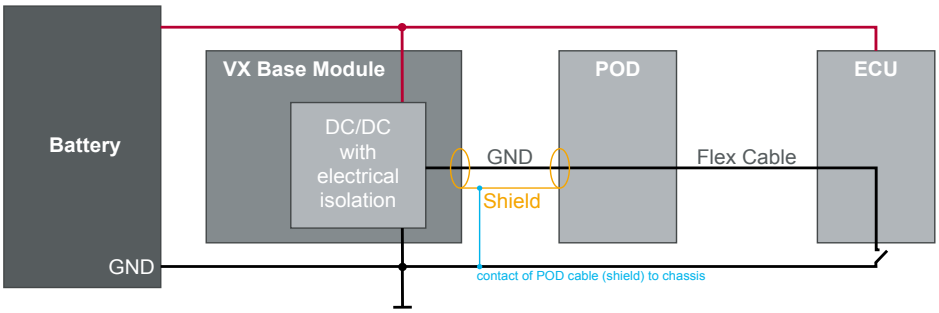


Figure 282: ECU with a short from cable shield to chassis

8.5.5.4 Current through VX Base Module (#4)

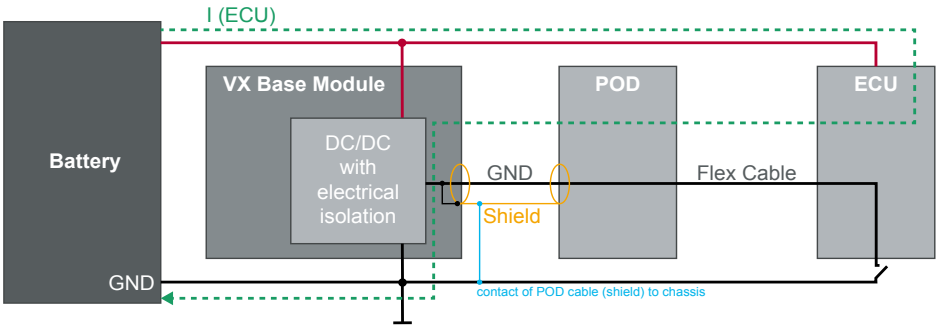


Figure 283: Error current through ground-shield connection in VX Base Module

Error case #4 happens if:

- The VX Base Module has an electrical ground-shield connection
- **and** the POD cable shield accidentally touches the chassis (while being plugged into the VX Base Module).

8.5.5.5 Current through POD Cable (#5)

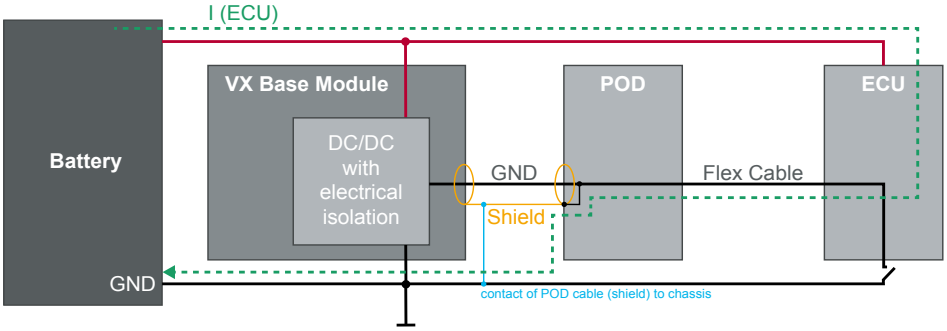


Figure 284: Error current through ground-shield connection in POD housing

Error case #5 happens if:

- The POD is mounted in a metal housing (leads to a ground-shield connection)
- **and** the POD cable shield accidentally touches the chassis.

8.5.5.6 Current through POD Housing (#6)

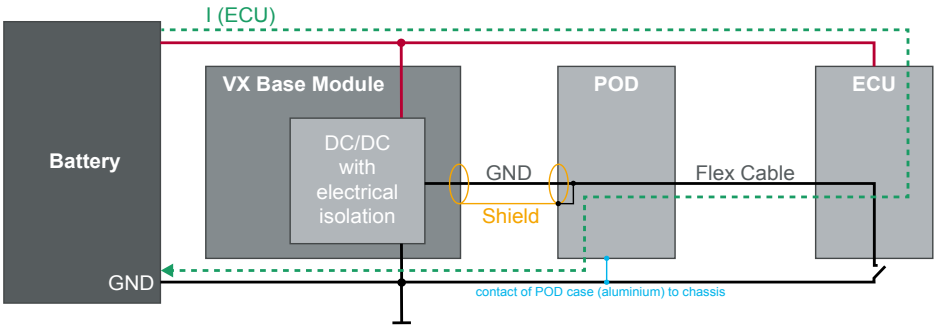


Figure 285: Error current through ground contact of POD housing

Error case #6 happens if:

- The POD is mounted in a metal housing (leads to a ground-shield connection)
- **and** the POD accidentally touches the chassis
- **or** the POD is mounted to the chassis (directly or via ECU housing).

8.6 Mechanical Integration

8.6.1 Introduction

The intended usage of a POD is to plug it directly onto the ECU or to connect it via a flex cable or an adapter board. In either case, the POD must be integrated into the housing of the ECU or mounted close to the ECU in an own housing.

Especially in a vehicle setup it is important to provide a robust housing for the POD that meets the mechanical requirements applicable for the ECU. In addition, there are other points to consider, for example:

- The POD should be integrated in a metal housing for EMI reasons and heat dissipation
- The POD and the ECU must be accessible for debugging or repair purposes, if necessary
- If applicable, the POD housing must be able to withstand the ingress of water and dust
- Active components of the POD may have to be thermally connected
- The POD housing needs to provide space for the internal and external wiring

Note
Integrating a POD is an individual task for each ECU. This task can be performed by the customer with Vector support or provided by Vector as a service. For more information contact VXsupport@vector.com .

WARNING!
VX1000 components must not be used in condensing environments, or in environments with conductive particles such as metal chips or swarf.

8.6.2 Basic Thoughts about Integrating a POD

There are several things to consider when integrating VX1000 components into an ECU. In this context, VX1000 components comprise a POD with internal wiring or adapter PCB, as well as the external wiring, such as a cable pigtail or a panel mount.

Following questions cover the main aspects:

Housing design and assembly

- Is it possible to assemble the POD inside the ECU housing, or is it necessary to extend the ECU housing?
- Is there a dedicated housing for ECUs with VX1000 components, or is a standard housing basically used?
- How can the housing with VX1000 components be assembled?
  - Can the VX1000 components be retrofitted?
  - Can the VX1000 components be integrated in series production?
  - Can preparatory steps simplify the integration afterwards?
- What material is the housing made of, can it be easily reworked?
- Are sealing planes easy to find, or is it difficult due to cooling fins or a complex surface structure?

Environment

- In which temperature range should ECU and VX1000 components operate?
- Can size and design of the housing dissipate the generated heat of ECU and POD?
  - How is the generated heat dissipated?
  - Is passive or active convection possible?
- What are the requirements for water and dust ingress (IP protection class)?
- On which side can external connectors be placed with regard to water ingress?

Installation location

- How much space is available around the ECU for VX1000 components?
- Is it possible to access the ECU in case of service?
- Is the VX1000 system connected via a cable pigtail or a panel mount?
- Can all plugs be connected and disconnected as desired?
- Is the constructed space surrounding the ECU sufficient for the external wiring?
  - Is the size of the connectors considered?
  - Is the minimal bending radius considered?

Inside the housing

- If POD and ECU share the same housing, is it possible to decouple the POD from the ECU by adding a metal shield between them?
- Which temperatures exist inside the ECU housing?
- Can the heat generated by the POD be dissipated inside the ECU housing or to the outside?
- Where is the POD attached to?
  - Are there mounting points on the ECU or in the housing?
  - Are all mechanical tolerances considered?
- Is there sufficient space for the internal wiring of the VX1000 components?
  - Is the minimal bending radius considered?

8.6.3 Heat Dissipation of a POD

Active components

On a POD, an FPGA is performing real time tasks as close as possible to the target microcontroller. Besides this, there may be other electronic components such as memory or power supplies, that generate mentionable heat on a POD. In the following these components are referred to as active components.

To guarantee stable operation of the setup over the complete temperature range, it may be necessary to dissipate the generated heat.

> **Note**
> Details on the need for heat dissipation and the location of the active components of a particular POD can be obtained via VXsupport@vector.com.

Heatsink dome

The heat generated by the POD should be dissipated through a metal housing if possible. For this purpose it is common practice to use heatsink domes in the design as part of the housing. These domes must be located above the active components.

A heatsink dome should thermally connect the surface of the component to the housing, but not create electrical contact. The distance between the component and the dome should be kept as small as possible, but as large as necessary to accommodate the mechanical tolerances in the design.

Filler material

The remaining gap between the component to be cooled and the heat sink must be filled with a suitable filler material. This is necessary to establish an adequate thermal contact and at the same time to compensate for the mechanical tolerances. To avoid condensation of released silicone oil, it is recommended to use a silicone free filling material.

Two types of filler material are usually considered:

Gap pad

As long as the combined tolerances of the gap are small, a gap pad can be used. Since gap pads cannot be compressed at will, the nominal height must be taken into account in the design. In addition, special attention should be paid to the tolerance calculation, as otherwise damage due to undesired pressure is possible. Depending on the type of gap pad, it may be allowed to reuse it if the housing has been opened for repair.

Liquid gap filler

Another option for closing the thermal gap is a liquid gap filler. It can be used for both, small and large gaps. While it is easy to dispense, it may require extra effort in case of a repair.

> **Note**
> Selected PODs are delivered with a gap pad. These included pads can only be used if the mechanical design is precisely matched to the given dimensions.

Following figure shows an example design that can be implemented with a heat pad between the heatsink dome and the FPGA surface:

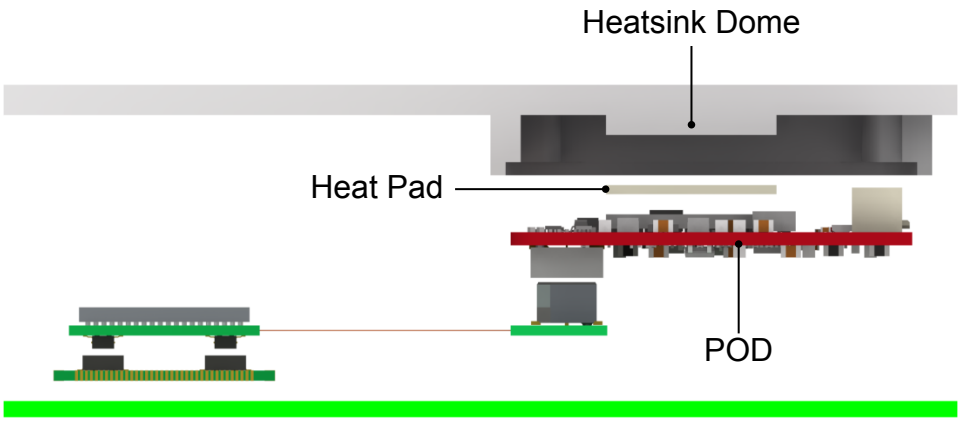


Figure 286: Example heat dissipation of a POD

In the design above the POD is mounted to the lid of the ECU housing at the top, which at the same time provides a heatsink dome to dissipate the heat. There is no rigid connection between the POD and the ECU. The accumulated tolerances of the gap between the dome and the active component of the POD are relatively small. In this case a gap pad can be used to close the thermal gap.

8.7 Wiring

8.7.1 Cable Pigtail

The cable pigtail consists of the following parts:

- **Mechanical Fixture including Strain Relief**
  The mechanical fixture and strain relief allows passing the pigtail through the wall of the ECU housing, at the same time sealing the opening and protecting the pigtail from getting pulled out of the housing by tension on the cable.
- **Sealing**
  The sealing has the function to keep moisture away from the ECU electronics.
- **EMC Protection**
  To fulfill EMC requirements the use of shielded cables is essential.
- **Shield Rest**
  The shield rest electrically connects the shield of a cable with the electrical ground of an assembly. This electrical ground is not necessarily equal to the metal housing of the assembly.

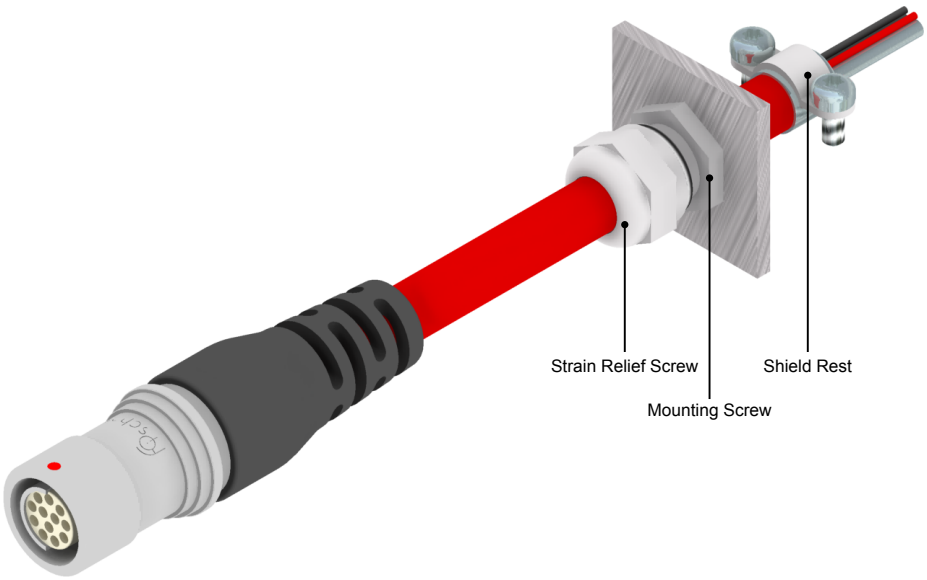


Figure 287: Cable Pigtail with mounting screw and shield rest

8.7.2 VX1381 HSSL2 Pigtail Installation

This section describes how a VX1381 HSSL2 Pigtail should be installed in an ECU. This step-by-step instruction is also enclosed with every VX1381 HSSL2 Pigtail.

8.7.2.1 Technical Specifications

Please consider the following cable characteristics when installing the pigtail:

- Temperature range: -40 °C ... +105 °C
  (plugging and unplugging only permitted above 0 °C)
- IP protection class when plugged: IP67

8.7.2.2 Installation Instructions

On POD side, the pigtail has crimp contacts. To allow insertion into the ECU housing, the contacts are not inserted into the connector body when delivered.

Each HSSL2 pigtail is accompanied by a 14-pin connector body. There are also PODs with 11-pin connector, like VX1463.02 or VX1463.20, which are delivered with the appropriate connector body!

Note
Please always ensure that connector body and POD match, before inserting the crimp contacts.

Step by Step Procedure

1. **Insert the pigtail into the designated opening in the ECU housing**
   - Avoid damaging the contacts
2. **Tighten the screw connection**
   - Spanner wrench with size 12
   - Torque: max. 6.5 Nm
3. **Twist the wires pairwise**
   - Each 2 cm one full 360° turn around the cable axis
   - The twisting direction of the wires within the cable must be continued
   - Twist evenly over wire length
4. **Insert crimp contacts into the connector body**
   - Mind the connector orientation (see figure)
   - Assign pins according to the table below
5. **Cable Routing**
   - Minimum bending radius of the cable: 70 mm,
     but can be reduced to 15 mm if permanently installed in a fixed position
   - Minimum bending radius of internal wires: 8 mm if paired,
     3 mm for separated wires
   - Do not damage or squeeze the cable sleeve

8.7.2.3 Pin Alignment of the 14 Pin Connector

Following table and figure show the pin-alignment suiting the 14 pin connector 87439-1400 (Molex):

Pin	Wire
1	-
2	white
3	brown
4	white/blue
5	white/yellow
6	-
7	white/green
8	white/orange
9	-
10	white/brown
11	white/black
12	-
13	-
14	-

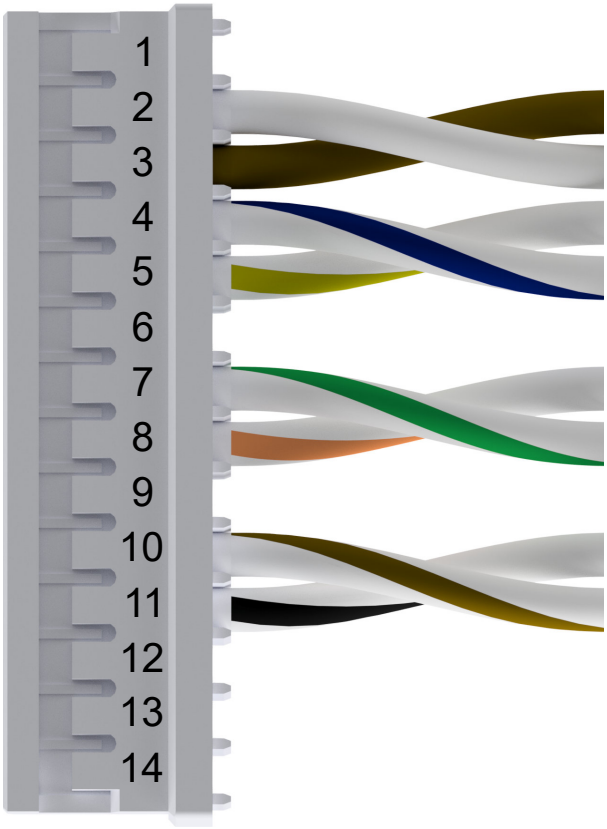


Figure 288: 14-Pin Connector 87439-1400 (Molex)

8.7.2.4 Pin Alignment of the 11 Pin Connector

Following table and figure show the pin-alignment suiting the 11 pin connector 87439-1100 (Molex):

Pin	Wire
1	white
2	brown
3	white/blue
4	white/yellow
5	-
6	white/green
7	white/orange
8	-
9	white/brown
10	white/black
11	-

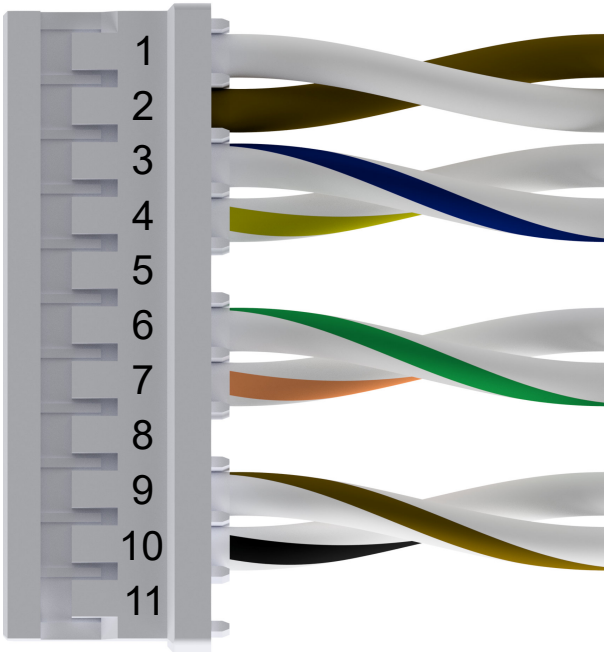


Figure 289: 11-Pin Connector 87439-1100 (Molex)

8.8 Advanced Topics

8.8.1 Aurora Reference Clock

8.8.1.1 Introduction

To implement data trace via Aurora, both participants must be equipped with a sufficiently stable clock base. This can be achieved either with independent oscillators (OSC) for the sender (TX) and the receiver (RX), or by providing an external clock to the sender.

If the target microcontroller has an integrated Aurora clock PLL, an external reference clock (for example 100 MHz) can be provided by the POD.

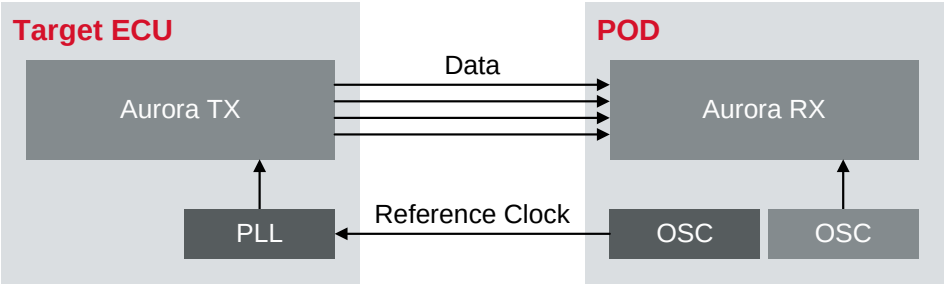


Figure 290: Aurora setup with external reference clock

In case the target microcontroller does not have an integrated Aurora clock PLL, the full Aurora clock (for example 1.25 GHz) has to be provided by the POD.

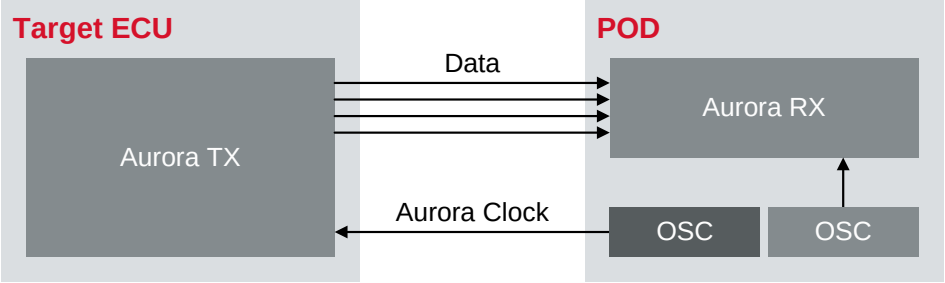


Figure 291: Aurora setup with full Aurora clock

If the required clock rate cannot be delivered by the POD, it may be necessary to implement a separate Aurora clock oscillator in the target design.

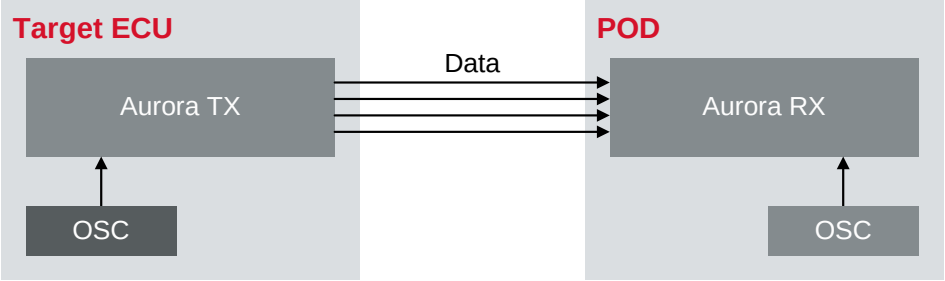


Figure 292: Aurora setup with independent clock oscillators

This section describes in which cases a separate clock oscillator is required in the

target ECU design and how it can be implemented.

8.8.1.2 Is a Separate Clock Oscillator Required?

With regard to the target microcontroller, a distinction must be made between three cases:

- The target provides an Aurora interface with an integrated oscillator that fulfills the clock quality requirements.
  Neither an external clock nor a separate oscillator is needed.
- The target provides an Aurora interface with PLL.
  A reference clock is required on the clock lanes (for example 100 MHz).
- The target provides an Aurora interface without PLL.
  The full Aurora clock is required (for example 1.25 GHz).

Various PODs can output an Aurora clock, whereas others have this option only in combination with an EEK head. In a bench setup the EEK head is usually an option, but it is not intended for use inside the vehicle.

Following table gives an overview which VX1000 components can output an Aurora clock:

VX1000 Component	Aurora Clock Output	Details
VX1452 POD	-	No Aurora clock output
VX1453 POD	-	No Aurora clock output
VX1454.01 POD	125 MHz	Fixed reference clock (70 ppm)
VX1461 POD	-	No Aurora clock output
VX1463.02 POD	100 MHz	Fixed reference clock (70 ppm)
VX1463.20 POD	100 MHz	Fixed reference clock (70 ppm)
VX1902.02 EEK Head	100 MHz	Fixed reference clock (70 ppm)
VX1902.03 EEK Head	Configurable	Configurable clock up to 1.5 GHz (50 ppm)
VX1902.04 EEK Head	Configurable	Configurable clock up to 1.5 GHz (50 ppm)

Note
The Aurora clock provided by the VX1000 components is an AC-coupled differential signal. To meet the requirements of the target microcontroller, this signal may need to be biased in the ECU design.

If the target microcontroller requires an external Aurora clock with a clock rate the intended VX1000 setup can not deliver, a separate Aurora clock oscillator needs to be implemented. This can either be done in a customized VX1000 solution, or implemented in the target ECU design as described below.

Note
For a customized VX1000 solution that provides a specific Aurora clock, please contact VXsupport@vector.com.

8.8.1.3 Choosing an Oscillator

The Aurora protocol is able to compensate clock differences between TX and RX in the range of 100 ppm. Hence, there is no need for a synchronized or shared clock. Instead, a free running clock oscillator eases the hardware integration on ECU side.

The required clock frequency is given by the target design. Microcontrollers such as the MPC57xx do not have an integrated Aurora clock PLL. In this case, the full Aurora clock (for example 1.25 GHz) must be provided. Devices like the NXP S32R294 (RaceRunner Ultra2) have an on-chip PLL for the Aurora clock that can be supplied with a reference clock at a lower frequency (for example 100 MHz).

The needed quality of the oscillator clock is mainly determined by the jitter requirements of the target microcontroller. A low-jitter oscillator is usually needed to meet these requirements. Refer to the microcontroller documentation or contact the semiconductor vendor for details.

In addition to the targeted frequency and quality of the clock signal, the outputs of the oscillator must comply with the I/O standard of the inputs of the target microcontroller. Usually a differential clock is needed that can for example be derived from oscillators with LVDS or LVPECL outputs. Notes on level conversion between different I/O standards are generally available.

8.8.1.4 Interfacing the Oscillator

If the setup is implemented with a VX145x HSSL POD, it is recommended to supply and enable the external Aurora clock oscillator by the POD. In case the setup is used for measurement over ECU reset, enabling the oscillator by the POD is mandatory.

Following pins of the target ECU connector of the HSSL POD (Samtec QSH-030-01-L-D-A) can be used to supply and control the oscillator:

Pin	Signal	Group	Need	Dir	I/O	Details
1	UPOD_EXT	-	o	PWR	5V	5 V voltage output delivering a maximum current of 50 mA. The capacitive load must not exceed 10 uF.
31	OE_SWITCH	-	(m)	OUT	3V3	Enable signal for an external oscillator (active high). Mandatory for measurement over ECU reset.

WARNING!
The OE_SWITCH pin of a HSSL POD is a 3.3 V output. Components controlled by the POD must either be supplied with 3.3 V, have 3.3 V tolerant inputs, or voltage limiting devices such as level shifters have to be used.

Note
If an external oscillator is to be controlled by the POD, please contact VXsupport@vector.com to ensure proper operation in the context of a dedicated VX1000 setup.

8.8.1.5 Example Implementation

The UPOD_EXT pin provides a 5 V voltage output that needs to be converted to the appropriate oscillator supply voltage using an low-noise LDO regulator.

Below is an example circuit diagram for a 2.5 V low noise power supply:

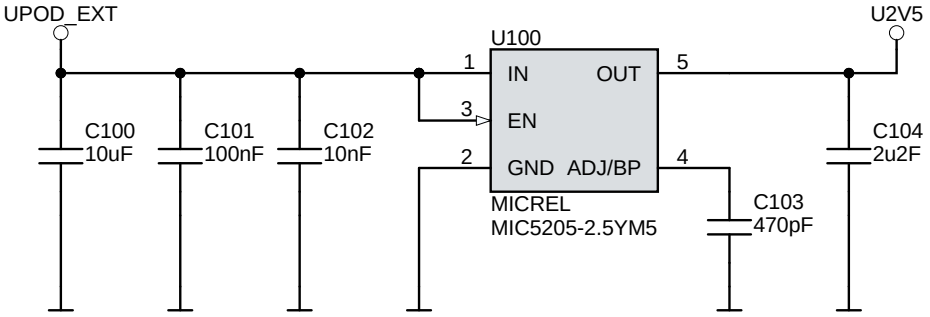


Figure 293: Aurora clock power Supply example circuit

It is recommended to add an additional filter in the power supply of the oscillator for improving the phase noise of the generated clock. This filter should be designed to suppress spurious signals on the supply voltage in a broad spectrum around the clock frequency.

The following example circuit can be used to generate a 100 MHz reference clock for a target Aurora interface. For other frequencies there are similar components available.

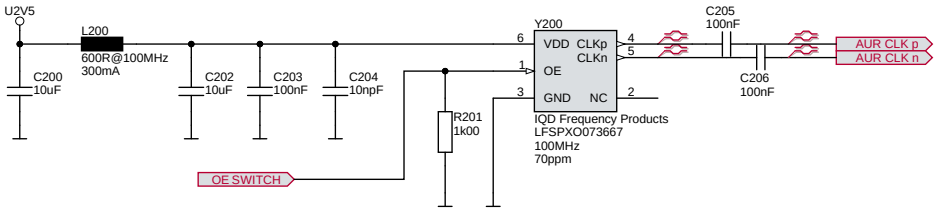


Figure 294: Aurora Reference Clock Oscillator Example Circuit

To further improve the quality of the clock signal, it is necessary to avoid crosstalk and minimize reflections on the clock traces by following the design recommendations given in section AUR (Aurora). Additionally, general notes on high-speed designs can be found in section Layout Recommendations.

9 Important Notes - Details

In this chapter you find the following information:

9.1 Safety Instructions and Hazard Warnings **737**
9.1.1 Proper Use and Intended Purpose 737
9.1.2 Hazards 738
9.1.3 Warning of Hot Surface 738
9.1.4 Electrostatically Sensitive Components (ESD) 738
9.2 Disclaimer **740**
9.3 Disposal of Vector Hardware **741**

9.1 Safety Instructions and Hazard Warnings

WARNING!
In order to avoid personal injuries and damage to property, you have to read and understand the following safety instructions and hazard warnings prior to installation and use of this hardware. Keep this documentation (manual) always near the hardware.

WARNING!
The use of the VX1000 hardware can cause unpredictable behaviour of the ECU if used incorrectly! It is strongly recommended that the safety concept of the ECU is considered before use, and that any use of calibration tools and the VX1000 hardware is in accordance with this safety concept! See also section Impact on ECU Safety.

9.1.1 Proper Use and Intended Purpose

WARNING!
The hardware is designed for analyzing, controlling and otherwise influencing control systems and electronic control units. This includes bus systems like CAN, FlexRay, Ethernet and/or Automotive Ethernet.

The hardware may only be operated (i) according to the instructions and descriptions of this manual; (ii) with the electric power supply designed for the hardware, e.g. USB-powered power supply; and (iii) with accessories manufactured or approved by Vector.

The hardware is exclusively designed for use by skilled personnel as its operation may result in serious personal injuries and damage to property. Therefore, only those persons may operate the hardware who (i) have understood the possible effects of the actions which may be caused by the hardware; (ii) are specifically trained in the handling with the hardware, bus systems and the system intended to be influenced; and (iii) have sufficient experience in using the hardware safely.

The knowledge necessary for the operation of the hardware can be acquired in work-shops and internal or external seminars offered by Vector. Additional and hardware specific information, such as „Known Issues“, are available in the „Vector KnowledgeBase“ on Vector´s website at www.vector.com. Please consult the „Vector KnowledgeBase“ for updated information prior to the operation of the hardware.

9.1.2 Hazards

WARNING!
The hardware may control and/or otherwise influence the behavior of control systems and electronic control units. Serious hazards for life, body and property may arise, in particular, without limitation, by interventions in safety relevant systems (e.g. by deactivating or otherwise manipulating the engine management, steering, airbag and/or braking system) and/or if the hardware is operated in public areas (e.g. public traffic, airspace). Therefore, you must always ensure that the hardware is used in a safe manner. This includes, inter alia, the ability to put the system in which the hardware is used into a safe state at any time (e.g. by „emergency shutdown“), in particular, without limitation, in the event of errors or hazards.

Comply with all safety standards and public regulations which are relevant for the operation of the system. Before you operate the system in public areas, it should be tested on a site which is not accessible to the public and specifically prepared for performing test drives in order to reduce hazards.

9.1.3 Warning of Hot Surface

WARNING!
There is danger of burning! The device can become very hot during operation and should not be touched without appropriate precautions.

9.1.4 Electrostatically Sensitive Components (ESD)

WARNING!
A sudden electrostatic discharge could destroy sensitive components. Therefore, proper packaging and grounding prescriptions must be observed. Please always observe the following safety advices:

1. Plug-in cards always have to be transported in electrostatically safe containers or bags.
2. Please leave electrostatically sensitive components in their containers until they have reached the electrostatically safe assembly site.
3. Take care that you are correctly grounded if you touch electrostatically sensitive components.
4. Make sure that electrostatically sensitive components are stored in protective packages or on antistatic mats.

Example

By observing the following measures, electrostatic damage to the device can be avoided:

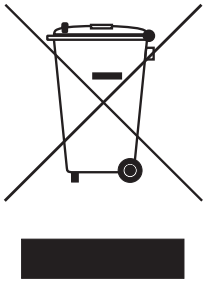
1. Lay out antistatic mats at the work place. Wear a grounding strap which is connected to the work place and the working tools.
2. Use antistatic foot mats, foot grounding equipment or air ionizers to provide additional safety.
3. Only touch sensitive components, plug-in cards and units on the housing or at the outer edges of the plug-in cards.
4. Avoid contact with pins, wires and conductor tracks.
5. Make sure that all voltage and signal sources are switched off before establishing or disconnecting electric connections or connecting testing devices.
6. Avoid non-conductive materials such as usual mounting accessories made of plastics or polystyrene at the work place.
7. Use conductive tools (e.g. ESD screwdrivers) when working on electrostatically sensitive units and components.
8. Always put plug-in cards and drives on the antistatic mat with the component side downwards.

9.2 Disclaimer

WARNING!
Claims based on defects and liability claims against Vector are excluded to the extent damages or errors are caused by improper use of the hardware or use not according to its intended purpose. The same applies to damages or errors arising from insufficient training or lack of experience of personnel using the hardware.

9.3 Disposal of Vector Hardware

Please handle old devices responsibly and observe the environmental laws applicable in your country. Please dispose of the Vector hardware only at the designated places and not with the household waste.



Within the European Community, the Directive on Waste Electrical and Electronic Equipment (WEEE Directive) and the Directive on the Restriction of the Use of Certain Hazardous Substances in Electrical and Electronic Equipment (RoHS Directive) apply.

For Germany and other EU countries, we offer free take-back of old Vector hardware.

Please carefully check the Vector hardware to be disposed of before shipping. Please remove all items that are not part of the original scope of delivery, e.g. storage media. The Vector hardware must also be free of licenses and must no longer contain any personal data. Vector does not perform any checks in this regard.
Once the hardware has been shipped, it cannot be returned to you. By shipping the hardware to us, you have relinquished your rights to the hardware.
Before shipping, please register your old device via:

https://www.vector.com/int/en/support-downloads/return-registration-for-the-disposal-of-vector-hardware/

10 Wichtige Hinweise

In diesem Kapitel finden Sie die folgenden Informationen:

10.1 Sicherheits- und Gefahrenhinweise **743**
10.1.1 Sach- und bestimmungsgemäßer Gebrauch 743
10.1.2 Gefahren 744
10.1.3 Warnung vor heißen Oberflächen 744
10.1.4 Elektrostatisch gefährdete Bauelemente (ESD) 744
10.2 Haftungsausschluss **746**
10.3 Entsorgung von Vector Hardware **747**

10.1 Sicherheits- und Gefahrenhinweise

WARNUNG!
Um Personen- und Sachschäden zu vermeiden, müssen Sie vor der Installation und dem Einsatz dieser Hardware die nachfolgenden Sicherheits- und Gefahrenhinweise lesen und verstehen. Bewahren Sie diese Dokumentation (Handbuch) stets in der Nähe dieser Hardware auf.

WARNUNG!
Die Verwendung der VX1000 Hardware kann bei unsachgemäßer Verwendung zu unvorhersehbarem Verhalten des Steuergeräts führen! Es wird dringend empfohlen, das Steuergeräte-Sicherheitskonzept vor dem Einsatz zu berücksichtigen und die Kalibrierwerkzeuge und VX1000 Hardware nur in Übereinstimmung mit diesem Sicherheitskonzept zu betreiben! Siehe auch Abschnitt Impact on ECU Safety.

10.1.1 Sach- und bestimmungsgemäßer Gebrauch

WARNUNG!
Die Hardware ist für die Analyse, die Steuerung sowie für die anderweitige Beeinflussung von Regelsystemen und Steuergeräten bestimmt. Das umfasst unter anderem die Bussysteme CAN, FlexRay, Ethernet und/oder Automotive Ethernet.

Diese Hardware ist entsprechend den Anweisungen und Beschreibungen dieses Handbuchs einzusetzen. Dabei darf nur die dafür vorgesehene Stromversorgung, wie z. B. USB-powered, Netzteil, und das Originalzubehör von Vector bzw. das von Vector freigegebene Zubehör verwendet werden.

Diese Hardware ist ausschließlich für den Einsatz durch geeignetes Personal bestimmt, da der Gebrauch dieser Hardware zu erheblichen Personen- und Sachschäden führen kann. Deshalb dürfen nur solche Personen diese Hardware einsetzen, welche die möglichen Konsequenzen der Aktionen mit dieser Hardware verstanden haben, speziell für den Umgang mit dieser Hardware, den Bussystemen und dem zu beeinflussenden System geschult worden sind und ausreichende Erfahrung im sicheren Umgang mit dieser Hardware erlangt haben.

Die notwendigen Kenntnisse zum Einsatz dieser Hardware können bei Vector über interne oder externe Seminare und Workshops erworben werden. Darüber hinausgehende und Hardware-spezifische Informationen wie z. B. „Known Issues“ sind auf der Vector Webseite unter www.vector.com in der „Vector KnowledgeBase“ verfügbar. Bitte informieren Sie sich dort vor dem Betrieb der Hardware über aktualisierte Hinweise.

10.1.2 Gefahren

WARNUNG!
Die Hardware kann das Verhalten von Regelsystemen und Steuergeräten steuern und in anderweitiger Weise beeinflussen. Insbesondere durch Eingriffe in sicherheitsrelevante Bereiche (z. B. durch Deaktivierung oder sonstige Manipulation der Motorsteuerung, des Lenk-, Airbag-, oder Bremssystems) und/oder der Einsatz der Hardware in öffentlichen Räumen (z. B. Straßenverkehr, Luftraum) können erhebliche Gefahren für Leib, Leben und Eigentum entstehen. Stellen Sie daher in jedem Fall eine gefahrfreie Verwendung sicher. Hierzu gehört unter anderem auch, dass das System, in dem die Hardware eingesetzt wird, jederzeit, insbesondere bei Auftreten von Fehlern oder Gefahren, in einen sicheren Zustand geführt werden kann (z. B. durch Not-Abschaltung).

Beachten Sie alle sicherheitstechnischen Richtlinien und öffentlich-rechtlichen Vorschriften, die für den Einsatz des Systems relevant sind. Zur Verminderung von Gefahren sollte das System vor dem Einsatz in öffentlichen Räumen auf einem nicht-öffentlich zugänglichen und für Testfahrten bestimmten Gelände erprobt werden.

10.1.3 Warnung vor heißen Oberflächen

WARNUNG!
Es besteht Verbrennungsgefahr! Das Gerät kann während des Betriebs sehr heiß werden und sollte ohne entsprechende Vorsichtsmaßnahmen nicht berührt werden.

10.1.4 Elektrostatisch gefährdete Bauelemente (ESD)

WARNUNG!
Eine plötzliche elektrostatische Entladung kann empfindliche Bauteile zerstören. Daher sind die Vorschriften für Verpackung und Erdung zu beachten. Bitte beachten Sie immer die folgenden Sicherheitshinweise:

1. Steckkarten müssen immer in elektrostatisch sicheren Behältern oder Taschen transportiert werden.
2. Bitte lassen Sie elektrostatisch gefährdete Bauelemente in ihren Behältern, bis sie die elektrostatisch sichere Montagestelle erreicht haben.
3. Achten Sie auf die richtige Erdung, wenn Sie elektrostatisch gefährdete Bauelemente berühren.
4. Achten Sie darauf, dass elektrostatisch gefährdete Bauelemente in Schutzverpackungen oder auf antistatischen Matten gelagert werden.

Beispiel
Durch die Beachtung der folgenden Maßnahmen können elektrostatische Schäden am Gerät vermieden werden:

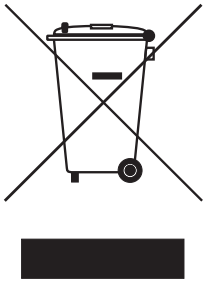
1. Antistatikmatten am Arbeitsplatz auslegen. Tragen Sie ein Erdungsband, das mit dem Arbeitsplatz und den Arbeitsgeräten verbunden ist.
2. Verwenden Sie antistatische Fußmatten, Fußerdungsgeräte oder Luftionisatoren, um zusätzliche Sicherheit zu gewährleisten.
3. Berühren Sie empfindliche Komponenten, Steckkarten und Geräte nur am Gehäuse oder an den Außenkanten der Steckkarten.
4. Vermeiden Sie den Kontakt mit den Pins, Drähten und Leiterbahnen.
5. Stellen Sie sicher, dass alle Spannungs- und Signalquellen ausgeschaltet sind, bevor Sie elektrische Verbindungen herstellen oder trennen oder Prüfgeräte anschließen.
6. Vermeiden Sie nicht leitfähige Materialien wie z. B. übliches Montagezubehör aus Kunststoff oder Polystyrol am Arbeitsplatz.
7. Verwenden Sie leitfähige Werkzeuge (z. B. ESD-Schraubendreher) bei Arbeiten an elektrostatisch empfindlichen Geräten und Komponenten.
8. Stecken Sie Steckkarten und Laufwerke immer mit der Bauteilseite nach unten auf die Antistatikmatte.

10.2 Haftungsausschluss

WARNUNG!
Soweit die Hardware nicht sach- oder bestimmungsgemäß eingesetzt wird, übernimmt Vector keine Gewährleistung oder Haftung für dadurch verursachte Schäden oder Fehler. Das Gleiche gilt für Schäden oder Fehler, die auf einer mangelnden Schulung oder Erfahrung derjenigen Personen beruhen, die die Hardware einsetzen.

10.3 Entsorgung von Vector Hardware

Bitte gehen Sie verantwortungsvoll mit Altgeräten um und beachten Sie die in Ihrem Land geltenden Umweltgesetze. Entsorgen Sie die Vector Hardware bitte nur bei den dafür vorgesehenen Stellen und nicht über den Hausmüll.



Innerhalb der Europäischen Gemeinschaft gelten die Richtlinie über Elektro- und Elektronik-Altgeräte (WEEE-Richtlinie) und die Richtlinie zur Beschränkung der Verwendung bestimmter gefährlicher Stoffe in Elektro- und Elektronikgeräten (RoHS-Richtlinie).

Für Deutschland und andere EU-Länder bieten wir Ihnen eine kostenlose Rücknahme der alten Vector Hardware an.

Bitte überprüfen Sie die zu entsorgende Vector Hardware vor dem Versand sorgfältig. Bitte entfernen Sie alle Gegenstände, die nicht zum ursprünglichen Lieferumfang gehören, z. B. Speichermedien. Die Vector Hardware muss außerdem frei von Lizenzen sein und darf keine personenbezogenen Daten mehr enthalten. Vector führt keine Kontrollen diesbezüglich durch. Sobald die Hardware versandt wurde, kann sie nicht mehr an Sie zurück geliefert werden. Mit dem Versand der Hardware an uns haben Sie Ihre Rechte an der Hardware abgetreten. Bitte melden Sie vor dem Versand Ihr Altgerät an über:

https://www.vector.com/de/de/support-downloads/return-registration-for-the-disposal-of-vector-hardware/

VECTOR

Visit our website for:

- News
- Products
- Demo software
- Support
- Training classes
- Addresses