



DEWETRON



SOFTWARE REFERENCE MANUAL

QUICK GUIDE DEWETRON LABVIEW DRIVERS 1.0.0.3

Version: 1.1

Date: 21st August 2026

Author: Maximilian Kaiß



DEWETRON

The information contained in this document is subject to change without notice.

DEWETRON GmbH (DEWETRON) shall not be liable for any errors contained in this document. DEWETRON MAKES NO WARRANTIES OF ANY KIND WITH REGARD TO THIS DOCUMENT, WHETHER EXPRESS OR IMPLIED. DEWETRON SPECIFICALLY DISCLAIMS THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

DEWETRON shall not be liable for any direct, indirect, special, incidental, or consequential damages, whether based on contract, tort, or any other legal theory, in connection with the furnishing of this document or the use of the information in this document.

Technical Support

Please contact your local authorized DEWETRON representative first for any support and service questions.

For Asia and Europe, please contact:

DEWETRON GmbH

Parking 4
8074 Grambach
AUSTRIA

Tel.: +43 316 3070
Fax: +43 316 307090
Email: support@dewetron.com
Web: <http://www.dewetron.com>

For America, please contact:

DEWETRON, Inc.

PO Box 1460
Charlestown, RI 02813
U.S.A.

Tel.: +1 401 364 9464
Toll-free: +1 877 431 5166
Fax: +1 401 364 8565
Email: support@dewamerica.com
Web: <http://www.dewamerica.com>

The telephone hotline is available Monday to Friday between 08:00 and 17:00 GST (GMT -5:00)

Restricted Rights Legend:

Use Austrian law for duplication or disclosure.

DEWETRON GmbH

Parking 4
8074 Grambach
AUSTRIA

Printing History:

Please refer to the page bottom for printing version. Copyright © DEWETRON GmbH



DEWETRON

This document is intended to be a start and quick guide for using CAN on NEX[DAQ] and on TRION-CAN and TRION3-CAN-FD boards in LabVIEW

INFORMATION:

In the example VIs a **NEX[DAQ]** is considered as a **TRIONet** with IP Address and Subnet Mask. Multiple NEX[DAQ]s are only assigned to one IP Address of the Adapter (USB or LAN), not the NEX[DAQ]s themselves.

A Library for Remote Control of DEWETRON Hardware via LabView and SCPI is also available (visit ccc.dewetron.com for downloads and the latest updates)

This document contains information which is protected by copyright. All rights are reserved. Reproduction, adaptation, or translation without prior written permission is prohibited, except as allowed under the copyright laws.

All trademarks and registered trademarks are acknowledged to be the property of their owners.

Before updating your software please contact DEWETRON. Use only original software from DEWETRON.

Please find further information at www.dewetron.com.

Table of Contents

1	GENERAL PROCEDURE.....	5
2	DRIVER INSTALLATION	6
3	EXAMPLE OVERVIEW.....	7
4	EXAMPLE SETUPS	8
1.	CAN ON NEX[DAQ]	8
2.	CAN ON TRION-CAN / TRION3-CAN-FD IN A CHASSIS.....	11
3.	CAN ON TRIONET	13
4.	CAN FUNCTIONALITY CHECK WITH CAN – LOOPBACK DEMO	14
5	TROUBLESHOOTING	15



DEWETRON

1 GENERAL PROCEDURE

1. Install “dewetron_gmbh_lib_dewetron_drivers” VI Package
2. Select desired example VI
3. In case of a Frontend (NEX[DAQ] or TRIONet):
 - a. Connect the Hardware (NEX[DAQ] or TRIONet)
 - b. Make sure IP Addresses assigned via DHCP
 - c. Find out the IP Address
 - d. In VI Front Panel enter correct IP Address
4. Run the VI



DEWETRON

2 DRIVER INSTALLATION

To install the “dewetron_gmbh_lib_dewetron_drivers-x.x.x.x.VI” you need the LabView VI Package Manager (VIPM). The easiest way to install the drivers is to just double click on the “dewetron_gmbh_lib_dewetron_drivers-x.x.x.x.VI” file in the file explorer. Follow the installation instructions.

dewetron_gmbh_lib_dewetron_drivers-1.0.0.1.vip 7/21/2026 3:11 PM VI Package 4,621 KB

1 Required

2 Install

JKI VIPM - DEWETRON Drivers

Package Information

Select an action to perform on the package.

2022.0.0 DEWETRON Drivers

DEWETRON Drivers v1.0.0.1 by DEWETRON GmbH
Author: DEWETRON GmbH
Copyright: Copyright (c) 2026, DEWETRON GmbH
License: MIT
Compatible LabVIEW Versions: >= 2021.
Compatible OS Versions: ALL.

Description:
Device drivers for DEWETRON's Trion hardware.

Release Notes:
CAN FD Support

This Package depends on these other packages:
OpenG Array Library >= 6.0.1.20
OpenG Error Library >= 6.0.0.26
OpenG File Library >= 6.0.2.28
OpenG Variant Library >= 6.0.0.45
OpenG String Library >= 6.0.2.35
OpenG Time Library >= 6.0.1.6

*Browse All Versions

3 Wait for Installation

JKI VIPM - DEWETRON Drivers

Package Information

Select an action to perform on the package.

2022.0.0

Install

Show in Palettes

Show Examples

*Browse All Versions

Extracting File: LabVIEW 2022\menus\Categories\Measurement\functions_Dewetron_GmbH_lib_Dewetron_Drivers.mnu ...

4 Finish

JKI VIPM - DEWETRON Drivers

Package Information

Select an action to perform on the package.

2022.0.0

Install

Show in Palettes

Show Examples

*Browse All Versions

Here are the results of the last action.

Product	Action	Status	V
✓ DEWETRON Drivers v1.0.0.1	installed	No Errors	

Include Dependencies
Select / Deselect All

Help

Finish

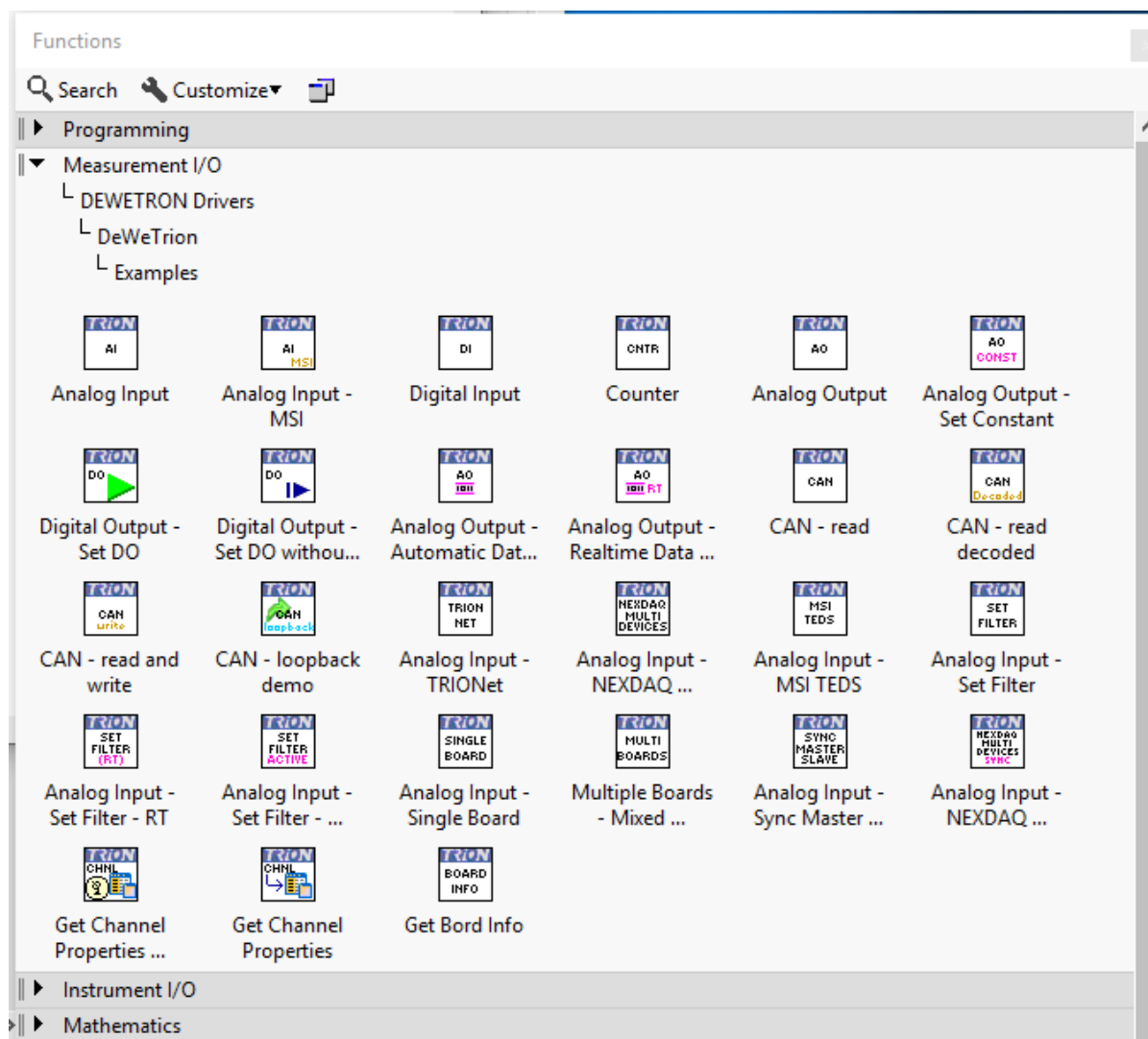


DEWETRON

3 EXAMPLE OVERVIEW

To access the DEWETRON example VIs in LabView open a Block Diagram (from a **Blank VI**). The Dewetron Examples can be found under

Functions->Measurement I/O-> Dewetron Drivers -> DeWeTrion -> Examples



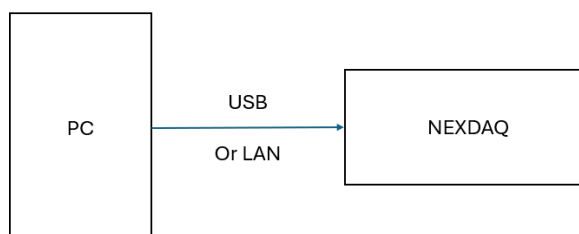


4 EXAMPLE SETUPS

1. CAN ON NEX[DAQ]

Connect the NEX[DAQ] via USB or LAN to the LabView PC. Wait about 60s for the start-up procedure of the hardware. Look up the IP Address of the Network Adapter (usually starting with 169.xxx.xxx.xxx and subnet mask 255.255.0.0). This can be done via the **ipconfig** command in the Windows Command Prompt. Use the **CAN – read decoded.vi** example.

Enter the IP Address and subnet mask in the VI Front Panel.



```
C:\WINDOWS\system32\cmd. X + v
Microsoft Windows [Version 10.0.22631.5039]
(c) Microsoft Corporation. All rights reserved.
C:\Users\Productmanagement>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Ethernet adapter Ethernet 2:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Ethernet adapter Ethernet 3:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Ethernet adapter Ethernet 13:

    Connection-specific DNS Suffix  . :
    Link-local IPv6 Address . . . . . : fe80::8b01:f35d:9819:86be%10
    IPv4 Address. . . . . : 192.168.56.1
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . :

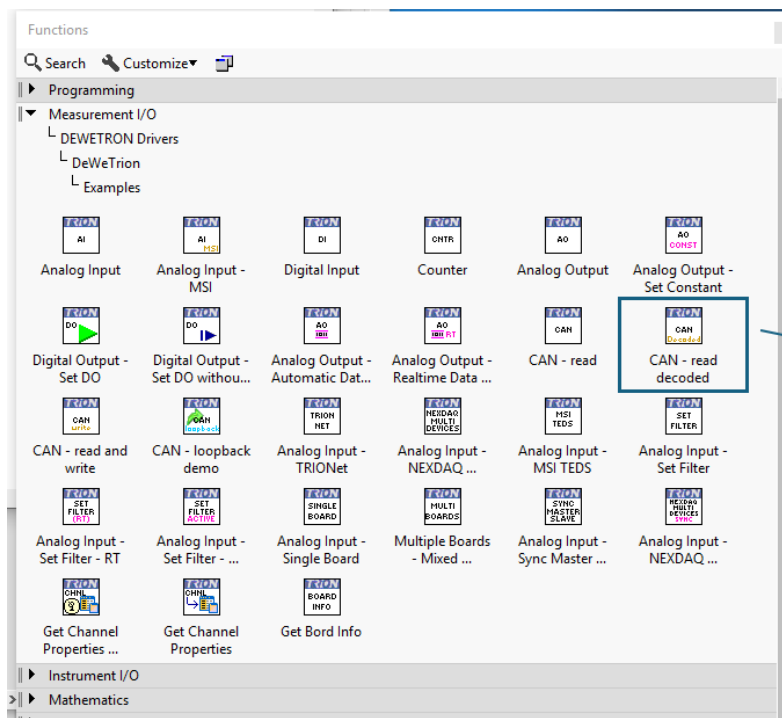
Ethernet adapter Ethernet 15:

    Connection-specific DNS Suffix  . :
    Link-local IPv6 Address . . . . . : fe80::1969:a3eb:fcec:2262%37
    Autoconfiguration IPv4 Address. . : 169.254.229.104
    Subnet Mask . . . . . : 255.255.0.0
    Default Gateway . . . . . :

C:\Users\Productmanagement>
```

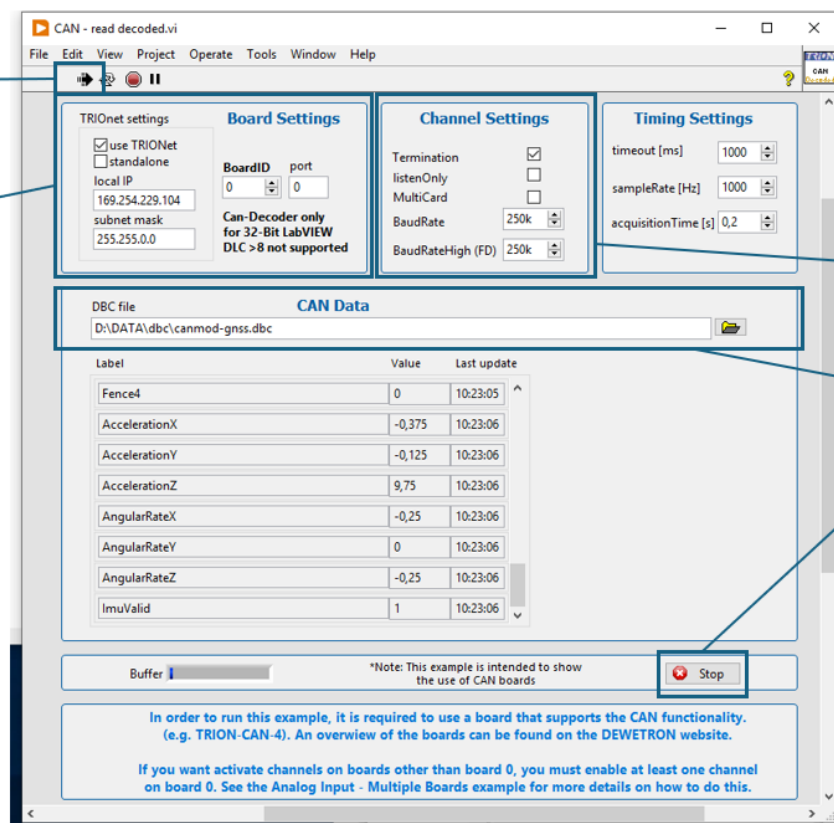


DEWETRON



Run VI

Network and Board Settings



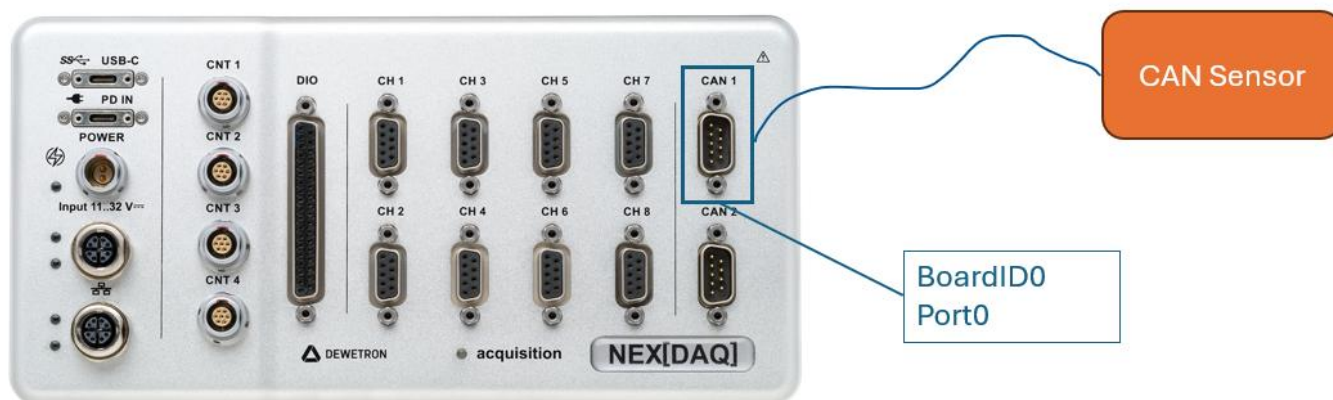
CAN Settings

Load DBC

Stop VI



DEWETRON



The CAN sensor in this setup communicates using Classical CAN at 250 kBd and is connected to the first CAN connector of the NEX[DAQ], which corresponds to BoardID 0, Port 0. The second CAN connector on the device is assigned to BoardID 0, Port 1.

To decode the incoming CAN data, load the appropriate DBC file in the CAN configuration of the VI. If no DBC file is available, the CAN-read.vi example can be used to view raw CAN frames.

Ensure that CAN bus termination is active; if the connected cable does not include a 120 Ω termination resistor, enable the termination option in the CAN settings to activate the internal termination of the NEX[DAQ]. For CAN-FD applications, the BaudRateHigh (FD) parameter must also be configured, but this sensor uses only Classical CAN, so the standard 250 kBd baud rate is sufficient.

After selecting the correct network, board, port, configuring the CAN parameters, and loading the DBC file, run the VI. The decoded signals will be displayed in a timestamped table on the front panel.

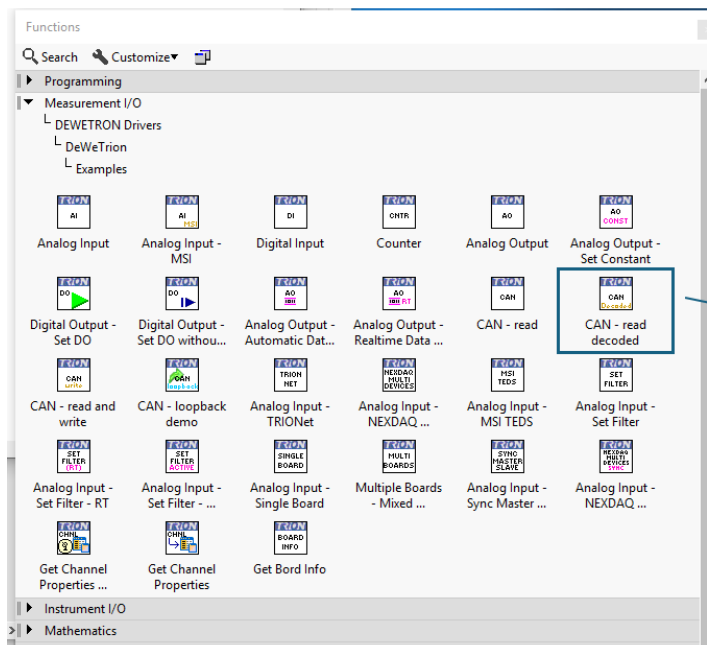


DEWETRON

2. CAN ON TRION-CAN / TRION3-CAN-FD IN A CHASSIS

For using a TRION-CAN or TRION3-CAN-FD board in a chassis, a DEWE3-A4 for example, the Network (TRIONet settings are not necessary).

Use the **CAN – read decoded.vi** example.



correct example

Run VI

Board Settings

CAN Settings

Load DBC

Stop VI

CAN - read decoded.vi

File Edit View Project Operate Tools Window Help

TRIONet settings

☐ use TRIONet
☐ standalone
local IP
subnet mask

Board Settings

BoardID port
3 0
Can-Decoder only for 32-Bit LabVIEW
DLC > 8 not supported

Channel Settings

Termination ☒
listenOnly ☐
MultiCard ☐
BaudRate 250k
BaudRateHigh (FD) 250k

Timing Settings

timeout [ms] 1000
sampleRate [Hz] 1000
acquisitionTime [s] 0.2

DBC file
D:\DATA\dbc\canmod-gnss.dbc

CAN Data

Label	Value	Last update
Fence4	0	09:28:04
AccelerationX	-0,375	09:28:04
AccelerationY	-0,125	09:28:04
AccelerationZ	9,75	09:28:04
AngularRateX	-0,5	09:28:04
AngularRateY	0	09:28:04
AngularRateZ	-0,25	09:28:04
ImuValid	1	09:28:04

Buffer

*Note: This example is intended to show the use of CAN boards

Stop

In order to run this example, it is required to use a board that supports the CAN functionality. (e.g. TRION-CAN-4). An overview of the boards can be found on the DEWETRON website.

If you want activate channels on boards other than board 0, you must enable at least one channel on board 0. See the Analog Input - Multiple Boards example for more details on how to do this.



DEWETRON

The CAN sensor in this setup communicates using Classical CAN at 250 kBd and is connected to the first CAN connector of the TRION3-CAN-FD board, which is BoardID 3 in this case, because on the used DEWE3-A4 there were 3 other boards before the TRION3-CAN-FD board. which corresponds to BoardID 0, Port 0. The second CAN connector on the device is assigned to BoardID 0, Port 1.

To decode the incoming CAN data, load the appropriate DBC file in the CAN configuration of the VI. If no DBC file is available, the CAN-read.vi example can be used to view raw CAN frames.

Ensure that CAN bus termination is active; if the connected cable does not include a 120 Ω termination resistor, enable the termination option in the CAN settings to activate the internal termination of the TRION(3)-CAN board. For CAN-FD applications, the BaudRateHigh (FD) parameter must also be configured, but this sensor uses only Classical CAN, so the standard 250 kBd baud rate is sufficient.

After selecting the correct board, port, configuring the CAN parameters, and loading the DBC file, run the VI. The decoded signals will be displayed in a timestamped table on the front panel.

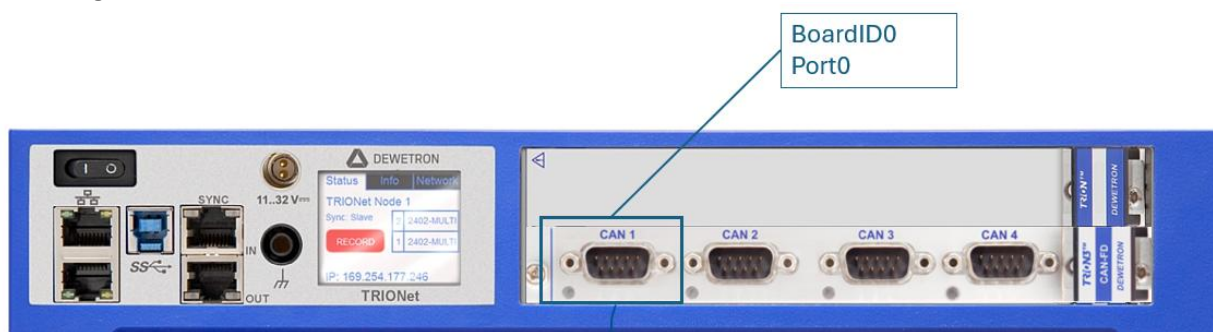


DEWETRON

3. CAN ON TRIONET

Using a TRIONet or TRIONet3 with a CAN board works identical to using CAN on NEX[DAQ]. Only difference is that TRIONets can have BoardID0 and BoardID1 for the 2 available slots, NEX[DAQ] is limited to BoardID0
Use the **CAN – read decoded.vi** example and see example description of **Error! Reference source not found..**

Tested configuration:



CAN Sensor

Run VI

Network and Board Settings

CAN - read decoded.vi

File Edit View Project Operate Tools Window Help

TRIONet settings

☒ use TRIONet
☐ standalone

local IP: 169.254.229.104
subnet mask: 255.255.0.0

Board Settings

BoardID: 0 port: 0

Can-Decoder only for 32-Bit LabVIEW
DLC > 8 not supported

Channel Settings

Termination: ☒
listenOnly: ☐
MultiCard: ☐
BaudRate: 250k
BaudRateHigh (FD): 250k

Timing Settings

timeout [ms]: 1000
sampleRate [Hz]: 1000
acquisitionTime [s]: 0.2

DBC file

Load DBC: D:\DATA\dbc\canmod-gnss.dbc

Label	Value	Last update
Fence4	0	10:23:05
AccelerationX	-0,375	10:23:06
AccelerationY	-0,125	10:23:06
AccelerationZ	9,75	10:23:06
AngularRateX	-0,25	10:23:06
AngularRateY	0	10:23:06
AngularRateZ	-0,25	10:23:06
ImuValid	1	10:23:06

Buffer: [Progress Bar]

*Note: This example is intended to show the use of CAN boards

Stop

In order to run this example, it is required to use a board that supports the CAN functionality. (e.g. TRION-CAN-4). An overview of the boards can be found on the DEWETRON website.

If you want activate channels on boards other than board 0, you must enable at least one channel on board 0. See the Analog Input - Multiple Boards example for more details on how to do this.

CAN Settings

Load DBC

Stop VI



DEWETRON

4. CAN FUNCTIONALITY CHECK WITH CAN – LOOPBACK DEMO

To check the CAN functionality on NEX[DAQ], TRION(3)-CAN in TRIONet and TRION(3)-CAN in chassis it is possible to use the **CAN – loopback demo.vi** example with just connecting CAN Port 1 to CAN Port 0 with a CAN cable.

This does not require a CAN sensor and just requires a 1:1 CAN cable connection from one CAN Port to the next on a TRION(3)-CAN board or on the NEX[DAQ].



Run VI

Network and Board Settings

CAN Settings

message to send

received message

Send message to other port

Stop VI

The used system for the example was a TRION3-CAN-FD on a DEWE3-A4 with board ID2. The first two CAN ports of the TRION3-CAN-FD board were used, sent from port 1 to port 0.

When using a TRIONet or NEX[DAQ] enter the correct IP address and subnet mask and tick the “**use TRIONet**” option. Set the BoardID and port to the CAN port to receive the manually sent message. Set the Data to send from Port 1 (for example) and hit the button “**Send**”, this message should appear on the received message on the left on Port 0.



DEWETRON

5 TROUBLESHOOTING

In case running an Example VI is throwing an error during execution try one of the following:

1. Make sure no other software is accessing the TRION API (OXYGEN and Dewetron Explorer are closed)
2. Make sure the TRION Applications and Firmware of the connected Hardware is up to date (visit ccc.dewetron.com for the latest updates)
3. Make sure the number of channels and the selected sample rate in the front panel is correct.
4. Make sure the IP Addresses are acquired via DHCP
5. Make sure the IP Address of the network adapter is entered, not the IP of TRIONet / NEX[DAQ] itself
6. Try turning off the Firewall
7. Make sure "Get Board Info.VI" example is functioning. That ensures the hardware can be accessed and read. You can also get the correct board Id information, if needed.
8. Restart LabView
9. Restart the connected hardware