CONNECTING TESTBEDS

THE ULTIMATE INTRODUCTION INTO DEWETRON'S WORLD OF TESTBED AND AUTOMATION INTERFACES

DEWETRON

THE MEASURABLE DIFFERENCE.

TABLE OF CONTENT

FURTHER INFORMATION?

Visit us on www.DEWETRON.com

DEWETRON provides various options for the integration of the measurement device into a testbed or any other third-party environment. This enables the usage of nearly any host as data sink and control instance.

		SCPI	XCP	DATA STREAM	ETHERNET RECEIVER / SENDER	ETHERCAT	CAN
	Physical Layer	Ethernet	Ethernet	Ethernet	Ethernet	TRION-EtherCAT	TRION-CAN
OUTPUT	Typical Speed	<= 100 S/s <= 10 kS/s (ELOG)	<= 10 kS/s	10 kS/s - 2 MS/s	<= 100 S/s	<= 500 S/s	<= 100 S/s
OUTPUT	Number of Channels	> 100	< 20	> 100	< 100	<= 100	> 20
OUTPUT	Timestamp Provision	Yes	Implicit	Yes	Yes	Yes	No
INPUT	Typical Speed	-	-	-	100 S/s - 1 kS/s	-	< 1 kS/s
INPUT	Number of Channels	-	-	-	> 100	-	> 100
INPUT	Sync to Timestamp	-	-	-	Yes	-	Receive timestamp
CONTROL	Start / Stop Recording	Yes	Yes	No	No	Yes	No
CONTROL	Save / Restore Setup	Yes	No	No	No	Partly	No
CONTROL	Modify Filename	Yes	No	No	No	No	No
CONTROL	Trigger	No	No	No	Yes	No	Yes
COMMENT		Generic interface for almost every application	Interfacing CANape or INCA	Performance interface for RAW streaming	Receiver and sender for UDP packets	Automation fieldbus compatible interface	Automotive fieldbus interface

Table: available testbed interfaces - overview

1 SCPI - THE GENERIC INTERFACE

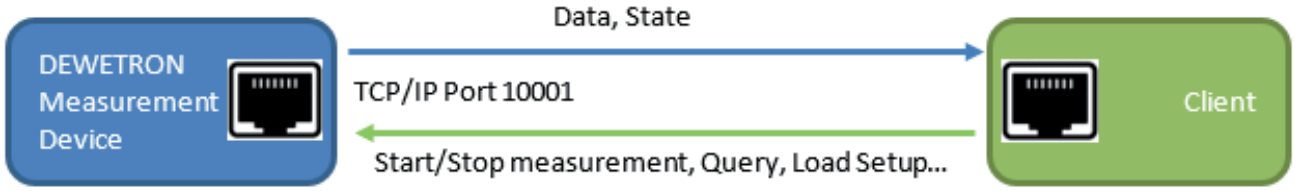

Figure: SCPI - exemplary depiction

SCPI is a plain text interface via Ethernet. It is used almost everywhere in the measurement automation area and is somehow standardized. The communication is performed with commands and queries. This interface is also used together with DATA STREAM as control layer.

Physical Interface	Ethernet
Communication Layer	TCP / IP, default port 10001
Language	Plain text
Data Output Capability	Single value fetch (scalar and array values) and data buffer fetch (ELOG)
Control Capability	Start / stop measurement, save / load setup, fetch single and buffered data
Implementation Complexity	Low
Timestamping / Sync	Optionally absolute or relative timestamp provided with the values

Table: SCPI - overview

1.1 MEASUREMENT DATA OUTPUT

The output via SCPI is performed via cyclic fetching of the data from the client application. Each query results in a single value (from 1 to n channels) or an array of values, if the ELOG buffered readout is used.

1.1.1 Single Value Fetching

Client -> OXYGEN	:NUM:ITEMS "ABS-TIME","U1_tRMS@POWER/0","I1_tRMS@POWER/0","P1_t@POWER/0"
Client -> OXYGEN	:NUM:VAL?
OXYGEN -> Client	:NUM:VAL "2019-03-18T08:41:55.359000+01:00",1.0099152E+2,1.0E+1,8.7461176E+2
Client -> OXYGEN	:NUM:VAL?
OXYGEN -> Client	:NUM:VAL "2019-03-18T08:41:59.108900+01:00",9.9607536E+1,1.0030489E+1,8.6756372E+2
Client -> OXYGEN	:NUM:VAL?
OXYGEN -> Client	:NUM:VAL "2019-03-18T08:42:09.058900+01:00",1.0053746E+2,1.0E+1,8.7068323E+2

Figure: single value fetching - example

1.1.2 Buffered Data Fetching

Since we have seen that it is potentially possible, that the same value can be fetched more than once and gaps can occur, we introduced the buffered readout (short ELOG, external logging)

```
:ELOG:ITEMS "U1_tRMS@POWER/0","I1_tRMS@POWER/0","P1_t@POWER/0"
:ELOG:PERIOD 0.1
:ELOG:TIMESTAMP ABS
:ELOG:START
:ELOG:FETCH?
:ELOG:FETC "2019-03-18T08:45:36.179000",1.007931E+2,1.0002865E+1,8.7337472E+2,"2019-03-18T08:45:36.279000",1.0067367E+2,
1.0000103E+1,8.7188635E+2,"2019-03-18T08:45:36.379000",1.0059721E+2,1.0000154E+1,8.7123651E+2,"2019-03-18T08:45:36.47900
0",1.004628E+2,9.9967834E+0,8.6952028E+2
```

Figure: buff ered data fetching - example

1.2 MEASUREMENT DATA INPUT

There is no measurement data input supported. See Ethernet receiver.

1.3 MEASUREMENT CONTROL

SCPI offers a wide command set for controlling the measurement device, like start / stop measurement and load / save setup. More information can be found in the OXYGEN user manual by following the link to DEWETRON's Customer Care Center.

2 XCP - THE ECU INTERFACE

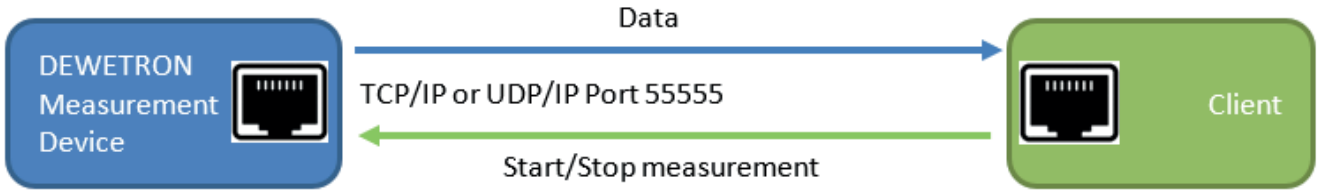

Figure: XCP - exemplary depiction

XCP is a commonly used interface in the automotive environment for ECU calibration. Due to its capability of polling and streaming data, it can also be used for performant measurement data transfer.

Physical Interface	Ethernet
Communication Layer	TCP / IP or UDP / IP, default port 55555
Language	Binary interface, ASAM standard
Data Output Capability	Data polling or DAQ-list with buffered data transfer
Control Capability	Start / stop measurement
Implementation Complexity	Medium
Compatible with	Vector CANape, ETAS INCA
Timestamping / Sync	Implicit due to DAQ-clock handshake

Table: XCP - overview

To establish the connection and configuration of the needed parameters, a device description file is needed to be loaded on the client side. Therefore, the measurement device generates an A2L-file, which represents the actual system configuration. Every time the setup gets changed on the measurement device, the A2L-file must be updated.

2.1 MEASUREMENT DATA OUTPUT

Due to the specifications of the XCP ASAM standard, a software-based DAQ-clock synchronization handshake is done at the beginning of the data transfer. The XCP implementation from DEWETRON supports data polling and DAQ-lists up to 10 kS/s data transfer rate.

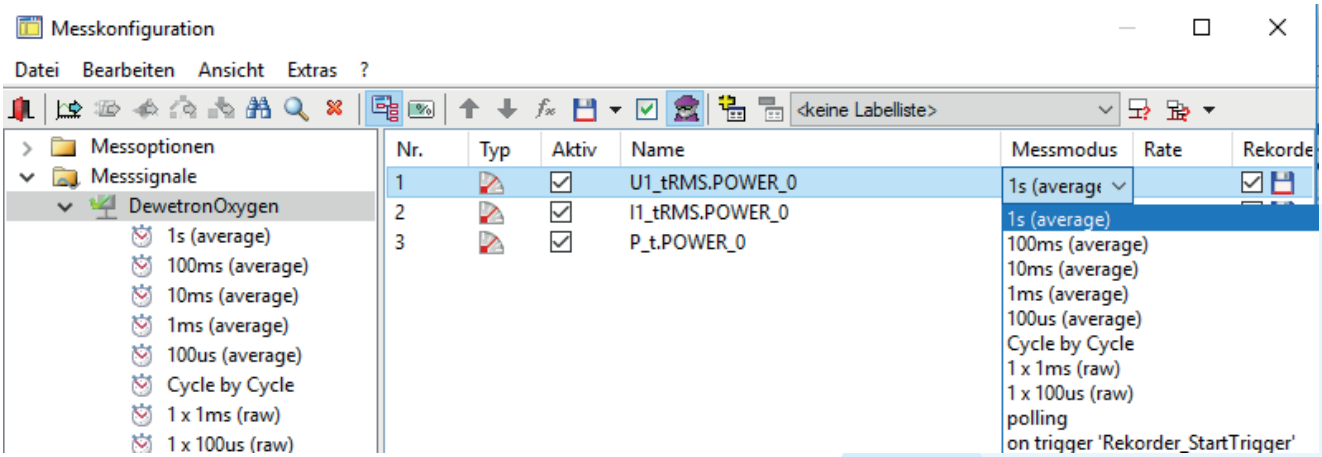

Figure: OXYGEN channel configuration in CANape for transfer over XCP - example

2.2 MEASUREMENT DATA INPUT

There is no measurement data input supported.

2.3 MEASUREMENT CONTROL

XCP offers a command set for start / stop of the measurement.

3 DATA STREAM - THE PERFORMANCE INTERFACE

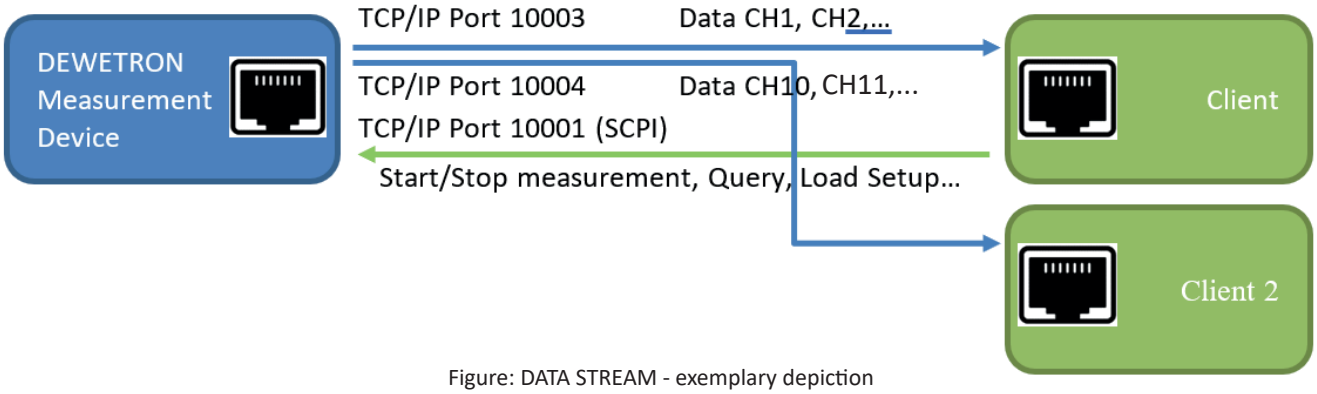

Figure: DATA STREAM - exemplary depiction

The DATA STREAM interface is the most performant interface for data output from the measurement device. It uses a pro-prietary streaming protocol (published on DEWETRON's Customer Care Center) optimized for speed and reliability.

Physical Interface	Ethernet
Communication Layer	Data: TCP / IP, default port 10003, multiple streams supported Control: TCP / IP, default port 10001 (SCPI)
Language	Binary interface, packet-based
Data Output Capability	Native data transfer („transferred as measured")
Control Capability	See SCPI
Implementation Complexity	Medium / high
Timestamping / Sync	Timestamp with each packet (equidistant data) or with each sample (non-equidistant data)

Table: DATA STREAM - overview

3.1 MEASUREMENT DATA OUTPUT

Selected measurement channels (selected via the SCPI interface) are provided on the defined TCP / IP socket. The used protocol is packet-based and needs a client with simple intelligence for decoding the data into single channels. The packets are optimized for highspeed data transfer with low overhead.

PACKET HEADER	PACKET INFO	CHANNEL INFO	SYNC CHANNEL VALUE+TIME	ASYNC CHANNEL VALUE+TIME	...	PACKET FOOTER
12 Byte	32 Byte	16 Byte ++	36 Byte ++	28 Byte ++	Val6	20 Byte

Table: general DATA STREAM definition

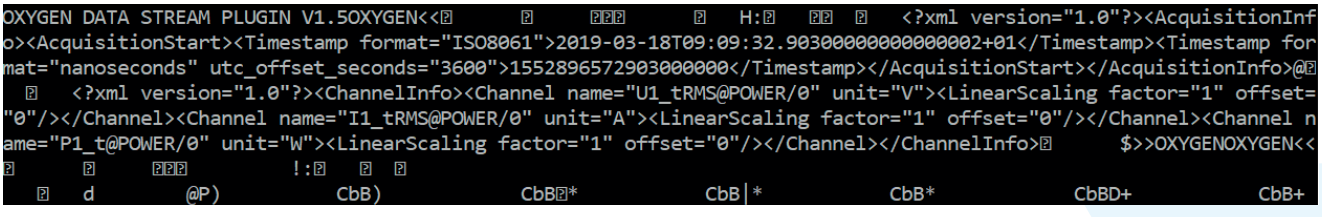

Figure: undecoded DATA STREAM - example

3.2 MEASUREMENT DATA INPUT

There is no measurement data input supported.

3.3 MEASUREMENT CONTROL

The configuration of one or more streams is done via SCPI commands. This means, the entire SCPI functionality is also available while using DATA STREAM.

4 ETHERNET RECEIVER / SENDER - THE UDP INTERFACE

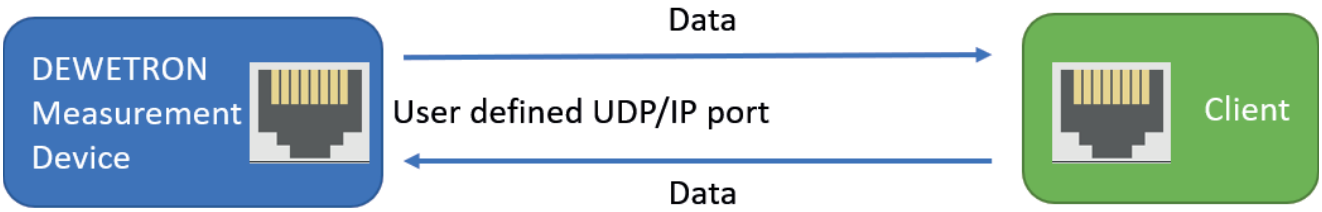

Figure: Ethernet receiver / sender - exemplary depiction

Complementary to the former listed interfaces, the Ethernet receiver is an input only interface. It is designed to receive and interpret UDP / IP packets to be used as sensor input. Vice versa, the Ethernet sender is designed to output data channels on an UDP socket.

Physical Interface	Ethernet
Communication Layer	UDP / IP
Language	Binary interface, packet-based, decoding information must be provided via xml-description
Data Output Capability	Interpret UDP packets as data channels, 1 timestamp each packet
Trigger Capability	Decoded input channel can be used as trigger
Implementation Complexity	Low
Timestamping / Sync	Timestamping on receive, optionally to included timestamp

Table: Ethernet interface - overview

4.1 MEASUREMENT DATA OUTPUT

Selected channels can be output on a defined UDP socket. The xml-file which describes the UDP stream is created automatically and can be used on the receiver side for data decoding.

4.2 MEASUREMENT DATA INPUT

The user can load an Ethernet receiver instance, which is described with an xml-file. The decoding follows a simple structure: each stream is basically bound to one UDP port and describes one “device”. The packets have to contain the data of one timestamp each and typically one or more data fields.

UDP-HEADER (INVISIBLE)	BYTE 0	BYTE 1	BYTE 2	BYTE 3	BYTE 4	BYTE 5	BYTE 6	BYTE 7	BYTE N
12 Byte	Channel 1 (signed)		Channel 2	Channel 3 (float)					

Table: general UDP stream definition

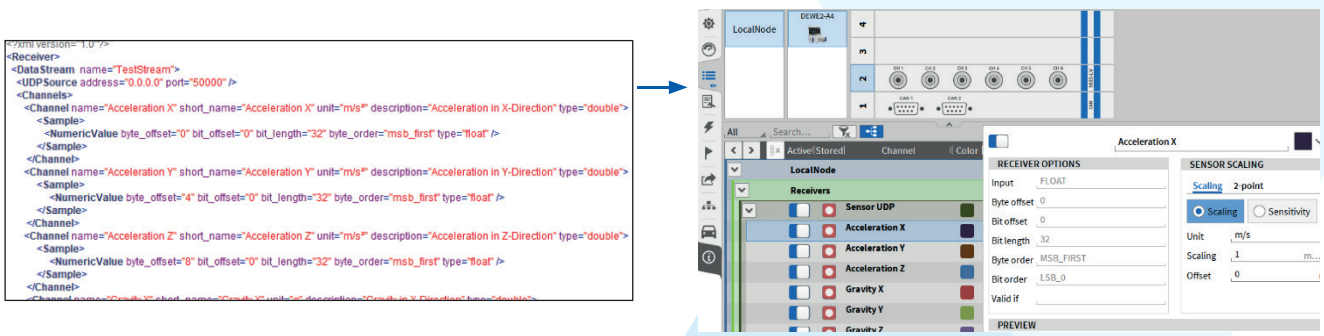

Figure: xml-file structure (left) and according OXYGEN channel list (right)

4.3 MEASUREMENT CONTROL

Use input data as trigger for recording.

5 ETHERCAT - THE AUTOMATION INTERFACE

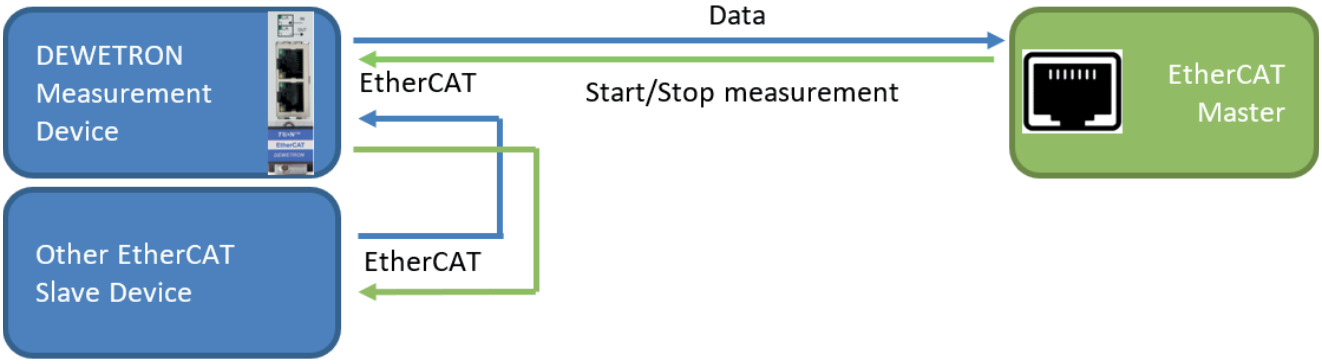

Figure: EtherCAT - exemplary depiction

Unlike the methods listed so far, a separate interface card is required (TRION-EtherCAT). This is necessary, because EtherCAT relies on a real time Ethernet stack within a line-topology. Each EtherCAT slave device (also the DEWETRON measurement device with TRION-EtherCAT) has one input and one output connector. The device description is generated according to output channel configuration in an ESI-file.

Physical Interface	RT-Ethernet, EtherCAT
Communication Layer	EtherCAT
Operation Type	Slave
Data Output Capability	Cyclical PDO update with measurement values (up to 100 channels)
Control Capability	Start / stop measurement, setup selection
Implementation Complexity	Medium
Timestamping / Sync	Absolute timestamp with each update
Compatible with	Beckhoff TwinCAT, KS Tornado

Table: EtherCAT - overview

5.1 MEASUREMENT DATA OUTPUT

The measurement device cyclically updates the data output memory (~ 500 times per second) with up to 100 channels with float (32-Bit) precision. Additionally, the first data slot is always the absolute timestamp in nanosecond resolution for post-synchronization.

5.2 MEASUREMENT DATA INPUT

There is no measurement data input supported.

5.3 MEASUREMENT CONTROL

Start / stop of measurement and selection of a pre-defined setup is possible.

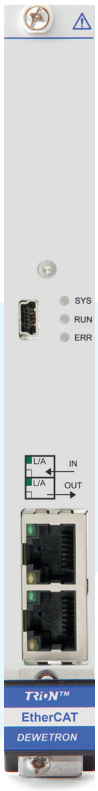

Figure: TRION-EtherCAT board

6 CAN - THE AUTOMOTIVE INTERFACE

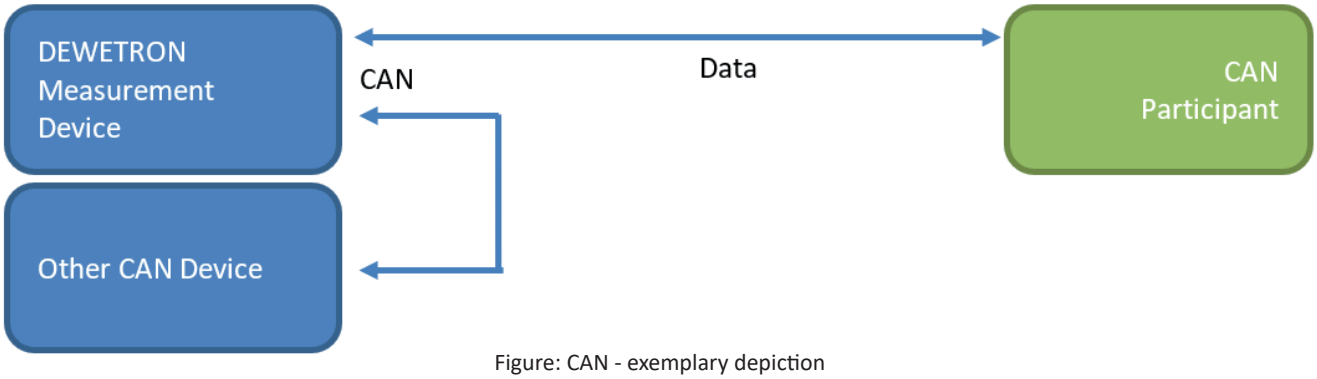

Figure: CAN - exemplary depiction

CAN is the second interface, where a separate interface card is needed (TRION-CAN, TRION-MULTI, TRION-1802-dLV, TRI-ON-1600-dLV) due to its different topology (2-wire differential interface). This is the most widely used interface bus in the automotive business. The CAN-interface provides in- and output from the measurement device.

Physical Interface	CAN (ISO 11898-2)
Communication Layer	Highspeed CAN
Data Input Capability	Decoding all types of CAN messages and signals
Data Output Capability	Cyclically send (up to 100 Hz) messages with measurement data
Control Capability	Use input signal as trigger
Implementation Complexity	Low
Timestamping / Sync	Receive timestamp

Table: CAN - overview

6.1 MEASUREMENT DATA OUTPUT

To output data, a DBC-file (CAN database format from Vector) has to be loaded and the desired measurement channel has to be assigned to the right signal. Due to protocol restrictions, the entire message is always sent, even if just one signal is transmitted.

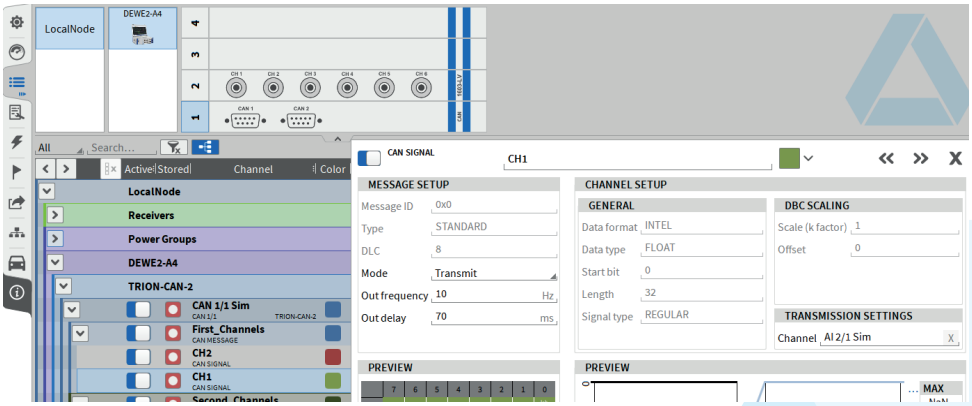

Figure: configuration of data output via CAN

6.2 MEASUREMENT DATA INPUT

Decoding, recording and visualization of the CAN signals is described within a DBC-file. For more details on CAN, please refer to the extensive CAN section in the OXYGEN user manual. Download it from DEWETRON's Customer Care Center.

6.3 MEASUREMENT CONTROL

An input signal can be used as trigger condition for actions.

7 MORE OPTIONS

In addition to the possibilities of OXYGEN measurement software, there are a few more ways for interacting with the measurement device and the data. Some of them are pre-installed with the operating system or can be installed by the user manually.

7.1 REMOTE DESKTOP CONNECTION (RDP)

The RDP protocol from Microsoft Windows is the commonly used interface in remote computing. It is lightweight and available within nearly every workstation.

7.2 VIRTUAL NETWORK COMPUTING (VNC)

An alternative to RDP is VNC, where the biggest advantage is the 1:1 mirroring of the measurement screen. We recommend the use of UltraVNC or TightVNC, which works best with the OpenGL rendering of the measurement application.

7.3 NETWORK DRIVES

The DEWETRON measurement device memory (HDD, SSD) can also be mapped as a network drive on a remote location or vice versa. This can be useful for backup purposes or cyclical data transfer to a separate file server.
Use this together with the powerful interfaces to automatically start and stop the recording and transfer the dmd-files to your host.

8 FAQ

Which interface is the right one for my application?
This cannot be answered in general. It depends on the quantity of data, the degree of automation and the implementation complexity. Please contact us, if you are not sure, we will guide you through a few steps to help with a decision.

Is the DEWETRON system real time compatible?
No, due to the fact, that all the measurement data is processed in a normal application with no real time capabilities, there is an I / O delay in the range of about 100 ms.

THE EXPERT

MICHAEL OBERHOFER

Michael Oberhofer is R&D manager and technical expert for power analysis at DEWETRON. He received his master's degree in electrical engineering with a focus on energy technology from the Graz University of Technology. Michael started his career at DEWETRON as an application engineer and sales consultant for electrical power analysis. In 2015, he became responsible for the technical product development, the definition of power analyzers and the software production backlog. Since 2019, he is DEWETRON's R&D manager and he therefore takes responsibility for the entire product development process including hardware as well as software.

FURTHER QUESTIONS? **CONTACT THE AUTHOR:**
michael.oberhofer@DEWETRON.com