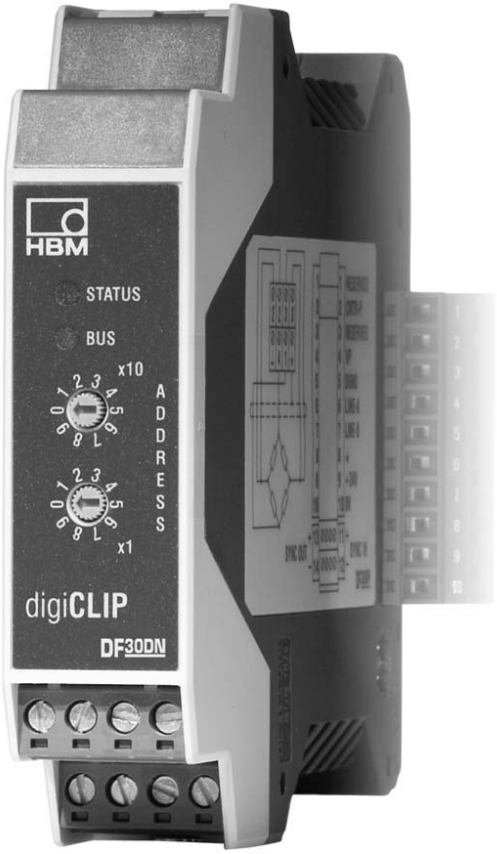
Operating Manual

DF30DN, DF31DN

digiCLIP



A3159-2.0 en

Contents **Page**

1 Safety instructions

Appropriate use

The digiCLIP with the connected transducers may be used for measurement and directly related control and regulation tasks, only. Any other use is not appropriate.

To ensure safe operation, the transducer may only be used according to the specifications given in this manual. It is also essential to comply with the legal and safety requirements for the application concerned during use. The same applies to the use of accessories.

Each time, before starting up the equipment, you must first run a project planning and risk analysis that takes into account all the safety aspects of automation technology. This particularly concerns personal and machine protection.

Additional safety precautions must be taken in plants where malfunctions could cause major damage, loss of data or even personal injury. In the event of a fault, these precautions establish safe operating conditions.

This can be done, for example, by mechanical interlocking, error signaling, limit value switches, etc.

AVERTISSEMENT

The device must not be connected directly to the mains supply. The supply voltage must be 10 V … 30 V (DC). It is essential to ensure that the device can be quickly disconnected from the mains supply at any time.

Before connecting the device, make sure that the mains voltage and current type specified on the name plate correspond to the mains voltage and current type at the site of installation and that the current circuit used is sufficiently safe.

The device complies with the safety requirements of DIN EN 61010-part1 (VDE 0411-part1).

General dangers in the case of non-observance of the safety instructions

The digiCLIP complies with the state of the art and is operationally reliable. If the device is used and operated inappropriately by untrained personnel, residual dangers might develop.

Any person charged with device installation, operation, maintenance or repair must in any case have read and understood the operating manual and the safety instructions, in particular.

Conditions on site

- Protect the device from direct contact with water.
- Protect the device from moisture and humidity or weather conditions such as rain, snow, etc. The degree of protection per EN 60529 standard is IP 20.
- Do not expose the device to direct sunlight.
- Please observe the permissible maximum ambient temperatures stated in the specifications.
- The permissible relative humidity at 31 °C is 95 % (non condensing); linear reduction to 50 % at 40 °C.
- Install the device so that it can be disconnected from the supply voltage at any time without difficulty.
- It is safe to operate the device up to a height of 2000 m.

Maintenance and cleaning

digiCLIP devices are maintenance-free.

- Withdraw the mains plug from the socket before carrying out any cleaning.
- Clean the housing with a soft, slightly damp (not wet!) cloth. You should **on no account** use solvent, since it may damage the labelling on the front panel and the indicator box.
- When cleaning, ensure that no liquid gets into the device or connections.

Residual dangers

The digiCLIP's scope of performance and supply covers part of the measuring-technology, only. The plant designer/constructor/operator must in addition design, realise and take responsibility for the measuring-system's safety such that potential residual dangers are minimized. The respective regulations must in any case be observed. Residual dangers regarding the measuringsystem must be specified explicitly.

Product liability

In the following cases, the protection provided for the device may be adversely affected. Liability for device functionality then passes to the operator:

- The device is not used in accordance with the operating manual.
- The device is used outside the field of application described in this Chapter.
- The operator makes unauthorized changes to the device.

Warning signs and danger symbols

Important instructions for your safety are specifically identified. It is essential to follow these instructions in order to prevent accidents and damage to property.

Safety instructions are structured as follows:

⚠AVERTISSEMENT

Type of danger

Consequences of non-compliance

Averting the danger

- **Warning sign:** draws your attention to the danger
- **Signal word:** indicates the severity of the danger (see table below)
- **Type of danger:** mentions the type or source of the danger
- **Consequences:** describes the consequences of non-compliance
- **Defense:** indicates how the danger can be avoided/bypassed

Danger class according to ANSI

Warning sign, signal word	Significance
WARNING	This marking warns of a *potentially* dangerous situation in which failure to comply with safety requirements *can* **result in death or serious physical injury.**
CAUTION	This marking warns of a *potentially* dangerous situation in which failure to comply with safety requirements *can* **result in slight or moderate physical injury.**
NOTE	This marking draws your attention to a situation in which failure to comply with safety requirements *could* **lead to damage to property**.

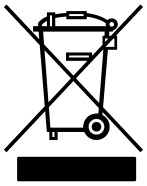
On the module

Meaning: **Take details in the operating manual into account**

On the module

Meaning: **CE mark**

The CE mark is used by the manufacturer to declare that the product complies with the requirements of the relevant EC directives (the Declaration of Conformity can be found at http://www.hbm.com/HBMdoc).



On the module

Meaning : **Statutory waste disposal mark**

The electrical and electronic devices that bear this symbol are subject to the European Waste Electrical and Electronic Equipment Directive 2002/96/EC.

The symbol indicates that the device must not be disposed of as household garbage.

In accordance with national and local environmental protection and material recovery and recycling regulations, old modules that can no longer be used must be disposed of separately and not with normal household garbage.

If you need more information about waste disposal, please contact your local authorities or the dealer from whom you purchased the product.

As waste disposal regulations within the EU may differ from country to country, we ask that you contact your supplier as necessary.

Environmental protection

The product will comply with general hazardous substances limits for at least 20 years, and will be ecologically safe to use during this period, as well as recyclable. This is documented by the following symbol.

On the modul

Meaning: **Statutory mark of compliance with emission limits in electronic equipment supplied to China.**

Working safely

Note

The device must not be connected directly to the mains supply. The supply voltage must be 10 V … 30 V (DC).

The supply connection, as well as the signal and sense leads, must be installed in such a way that electromagnetic interference does not adversely affect device functionality (HBM recommendation: ”Greenline shielding design”, downloadable from the Internet at http://www.hbm.com/Greenline).

Automation equipment and devices must be covered over in such a way that adequate protection or locking against unintentional actuation is provided (such as access checks, password protection, etc.).

When devices are working in a network, these networks must be designed in such a way that malfunctions in individual nodes can be detected and shut down.

Safety precautions must be taken both in terms of hardware and software, so that a line break or other interruptions to signal transmission, such as via the bus interfaces, do not cause undefined states or loss of data in the automation device.

The digiCLIP module must be operated with a safety extra low voltage (18 to 30 V supply voltage (DC)).The supply voltage lead must be no more than 3 m long. **Connecting to a direct voltage network in accordance with EN 61326 is not permitted.** Instead you must use a power pack mounted, for example, in the control cabinet, together with the digiCLIP modules.

Note

This is a Class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

Reconstruction and modifications

HBM’s express consent is required for modifications regarding the digiCLIP’s construction and safety. HBM does not take responsibility for damage resulting from unauthorized modifications.

In particular, repair and soldering works on the boards are prohibited. If complete componentry is replaced use original HBM components, only.

The product is delivered from the factory with a fixed hardware and software configuration. Changes can only be made within the possibilities documented in the manuals.

Qualified personnel

Qualified personnel means persons entrusted with siting, mounting, starting up and operating the product, who possess the appropriate qualifications for their function (qualified electrician, or by someone with electrical training under the supervision of a qualified electrician).

This device is only to be installed and used by qualified personnel strictly in accordance with the specifications and with the safety rules and regulations which follow.

This includes people who meet at least one of the three following requirements:

- Knowledge of the safety concepts of automation technology is a requirement and as project personnel, you must be familiar with these concepts.
- As automation plant operating personnel, you have been instructed how to handle the machinery and are familiar with the operation of the equipment and technologies described in this documentation.
- As commissioning engineers or service engineers, you have successfully completed the training to qualify you to repair the automation systems. You are also authorized to activate, to ground and label circuits and equipment in accordance with safety engineering standards.

It is also essential to comply with the legal and safety requirements for the application concerned during use. The same applies to the use of accessories.

Maintenance and repair work on an open device with the power on must only be carried out by trained personnel who are aware of the dangers involved. During installation and operation, operating personnel must act in accordance with the electrostatic discharge safety measures.

Important

The safety instructions are also included in paper format with the product.

2 Introduction

2.1 Scope of supply and accessories

Scope of supply:

- 1 digiCLIP module — Order no.: 1−DF30DN / Order no.: 1−DF31DN
- Coded plug connector for the sensor connection — Order no.: 3−3312.0404
- Connector terminal for DeviceNet and supply voltage — Combicon order no.: CR−MSTB
- digiCLIP Operating Manual

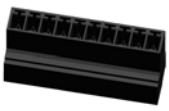
CD−ROM including free setup software (digiCLIP Assistant), (a free updated version of the Assistant can be downloaded from http://www.hbm.com/support).

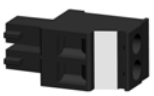
For DF31DN:

- Coded plug connector for digital IN/OUT (2 pieces)
  - 24 V / 0 V Order No.: 3−3312.0418
  - IN / OUT Order No.: 3−3312.0444

Accessories:

- 1 connector set: Order no.:1−digiCLIP−ST

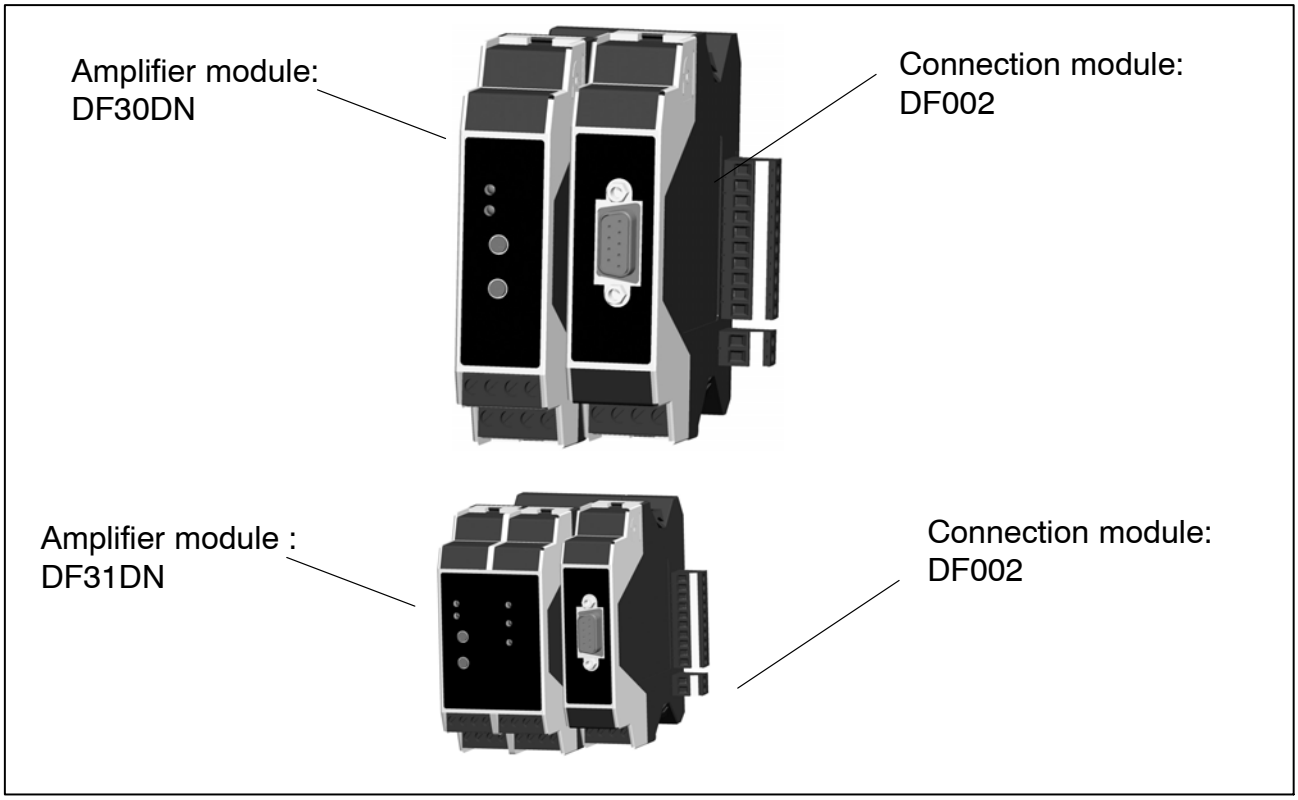
  containing 1 ”DeviceNet” connector terminal  and

  1 male and 1 female connector for ”synchronization”  

  (needed for two−tier installation in the control cabinet)

- Setup Toolkit for digiCLIP with a CAN to USB adapter, a connection cable and free setup software (digiCLIP Assistant) Order no.:1−digiCLIP−Setup

- Connection module for frontal assignment of the rear terminal strip (bus and power supply) Order no.:1−DF002



2.2 General

The DF30DN/DF31DN module from the digiCLIP product line is a carrier−frequency amplifier suitable for connecting force transducers, pressure transducers, torque transducers and load cells.

The DF30DN/DF31DN module is set up and parameterized by means of the digiCLIP Setup Assistant and a simple interface under MS−Windows.

The Setup Assistant also provides extensive Online Help, with descriptions of all the functions and many tips for the DF30DN/DF31DN.

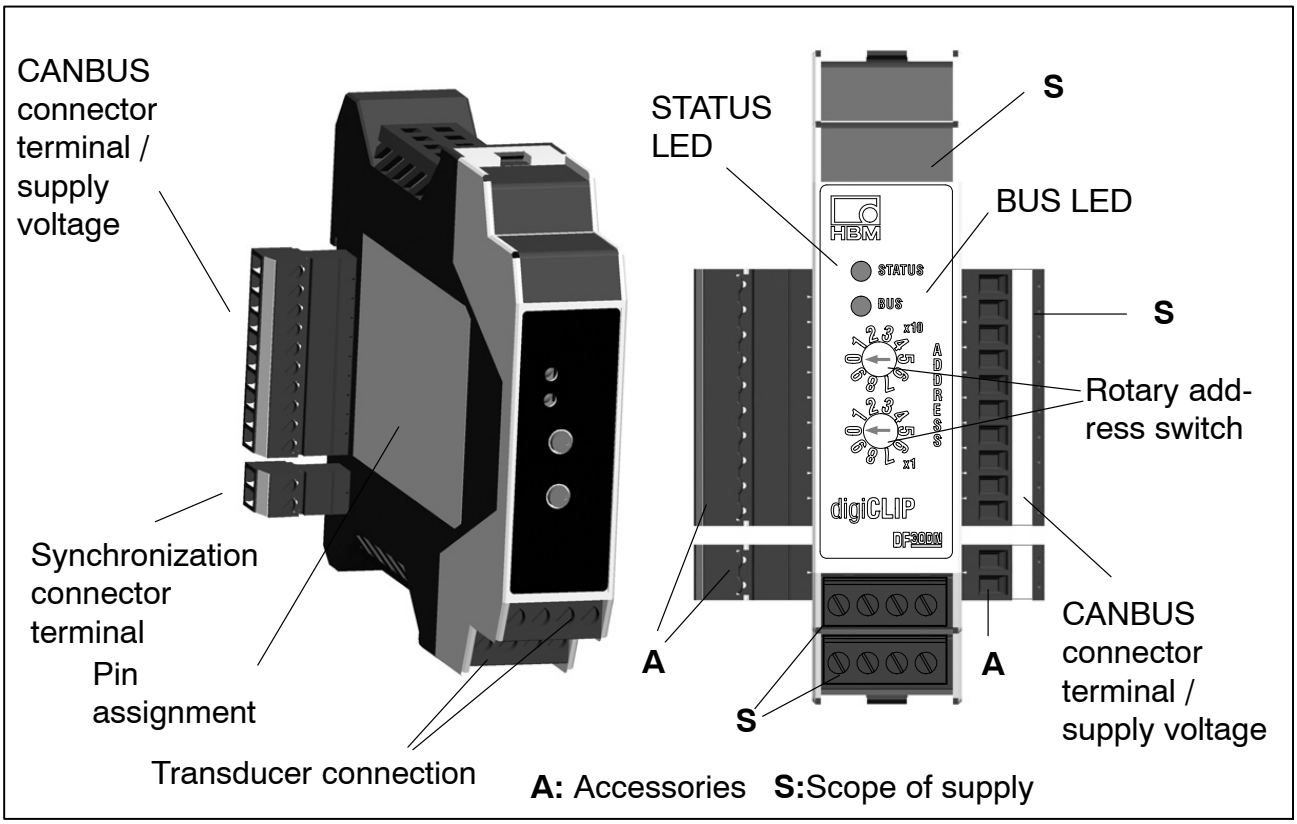


Fig. 2.1: digiCLIP module DF30DN

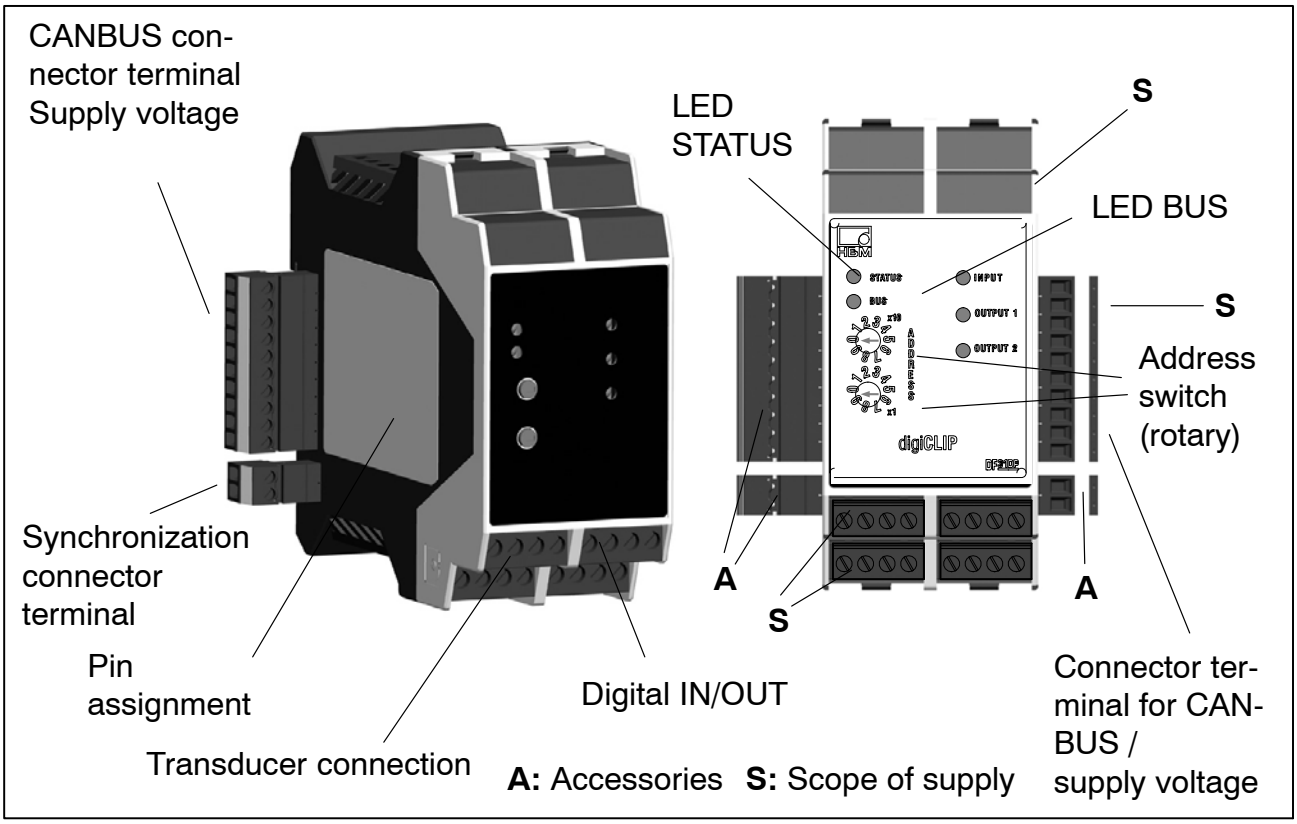


Fig. 2.2: digiCLIP-Modul DF31DN

3 Installation

The modules are mounted on support rails in accordance with DIN EN 60715 (IEC 60715) by hooking on the top edge and engaging the spring plate at the bottom edge.

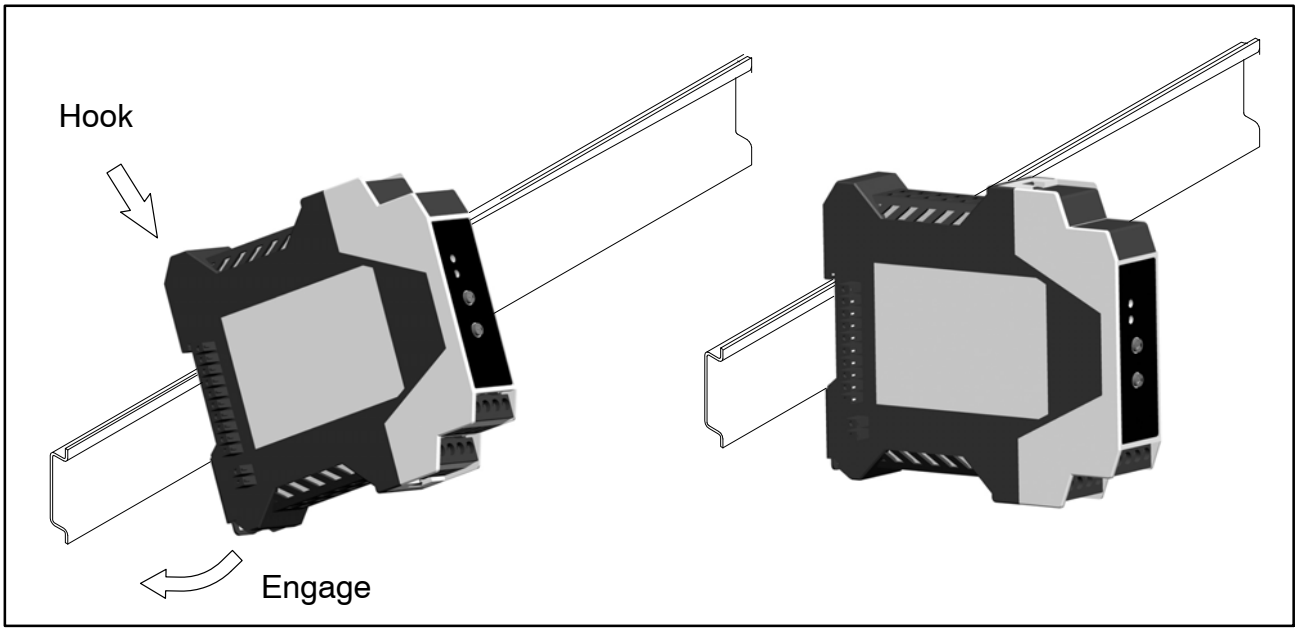


Fig. 3.1 Mounting on a support rail (here: DF30DN)

To remove, press down on the spring plate with a screwdriver and detach the housing.

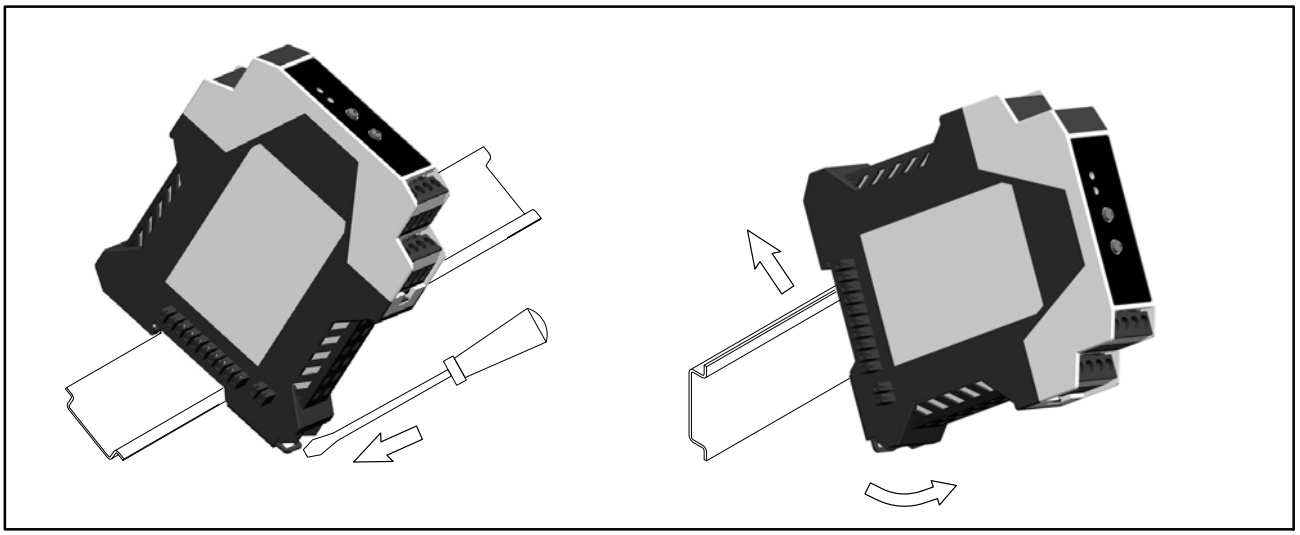


Fig. 3.2: Removal

AVERTISSEMENT

The support rail should be connected to grounded conductor potential ⏚ .

Several type DF30DN and DF31DN can be connected, also in mixed operation. The rear multipoint connector with internal wiring makes the local connection for supply voltage, DeviceNet and synchronization.

Interconnecting several modules:

- Intermate modules 1, 2 and 3
- When mounting at several levels: mate the SYNC–OUT connector to module 3 (see Fig. 3.4 and Fig. 4.5) and connect to SYNC–IN of the first module of the next level

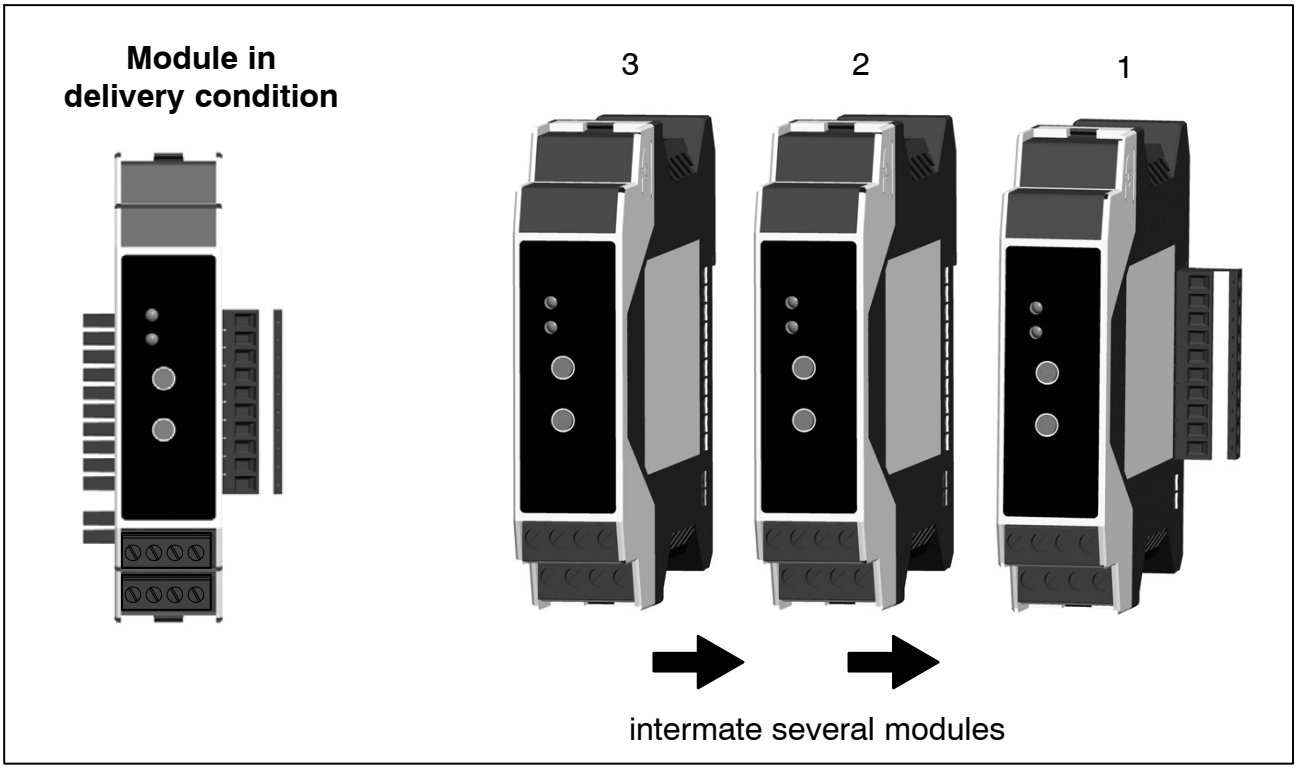


Fig. 3.3 Module installation

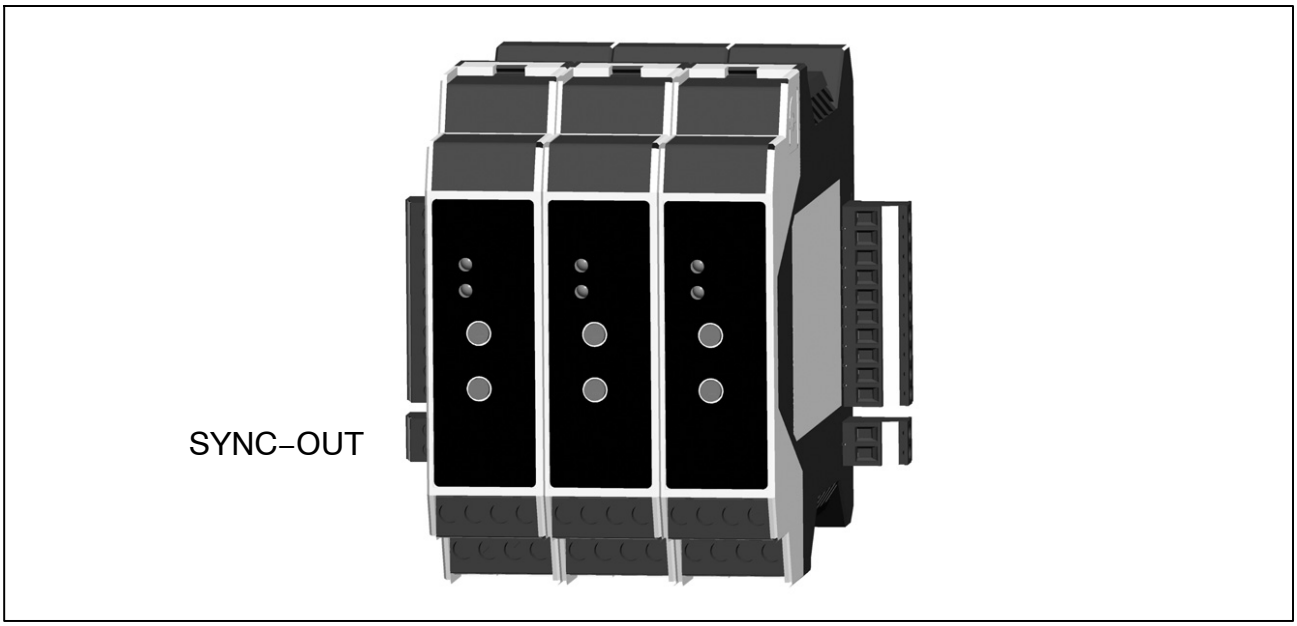


Fig. 3.4: Modules mounted side–by–side

4 Electrical connection

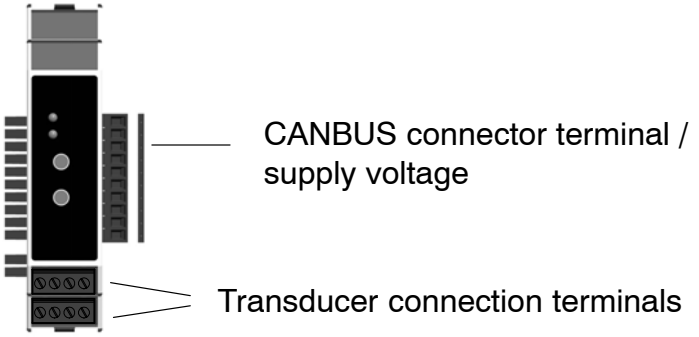
Transducers can be connected to the module in a strain gage full–bridge circuit.

The transducers are connected via 2 screw terminals on the front. Use the strain relief provided. The shield of the transducer cable must make contact over a large area. The clamping area is 0.2 mm^2 to 3.3 mm^2.

If several conductors are to be connected to a terminal, the line cross–section must be adapted accordingly.

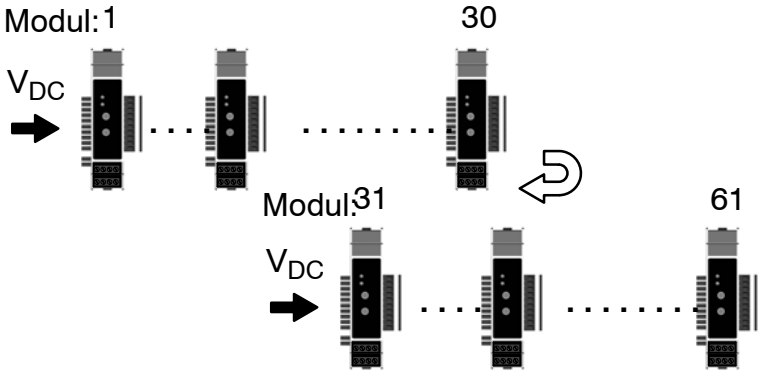
The DeviceNet and the power supply can be connected via the 10–pin terminal strip at the side, or via an adapter module. The clamping area is 0.05 mm^2 to 2 mm^2. Connection module DF002 can be used as an alternative.

End sleeves (without plastic collars, length 10 mm) should be used on the strands to connect the wires to the terminals.



Note

*To ensure that the electric load-carrying capacity of the plug terminals is not exceeded, a maximum of 30 modules can be intermated when **one** voltage source is connected.*
*If there are more than 30 modules, the series must be split and an **additional** voltage source connected.*



Power is supplied via the DF002 adapter module's *front panel* or the *multipoint connector at its rear* (pin 9 and 10).

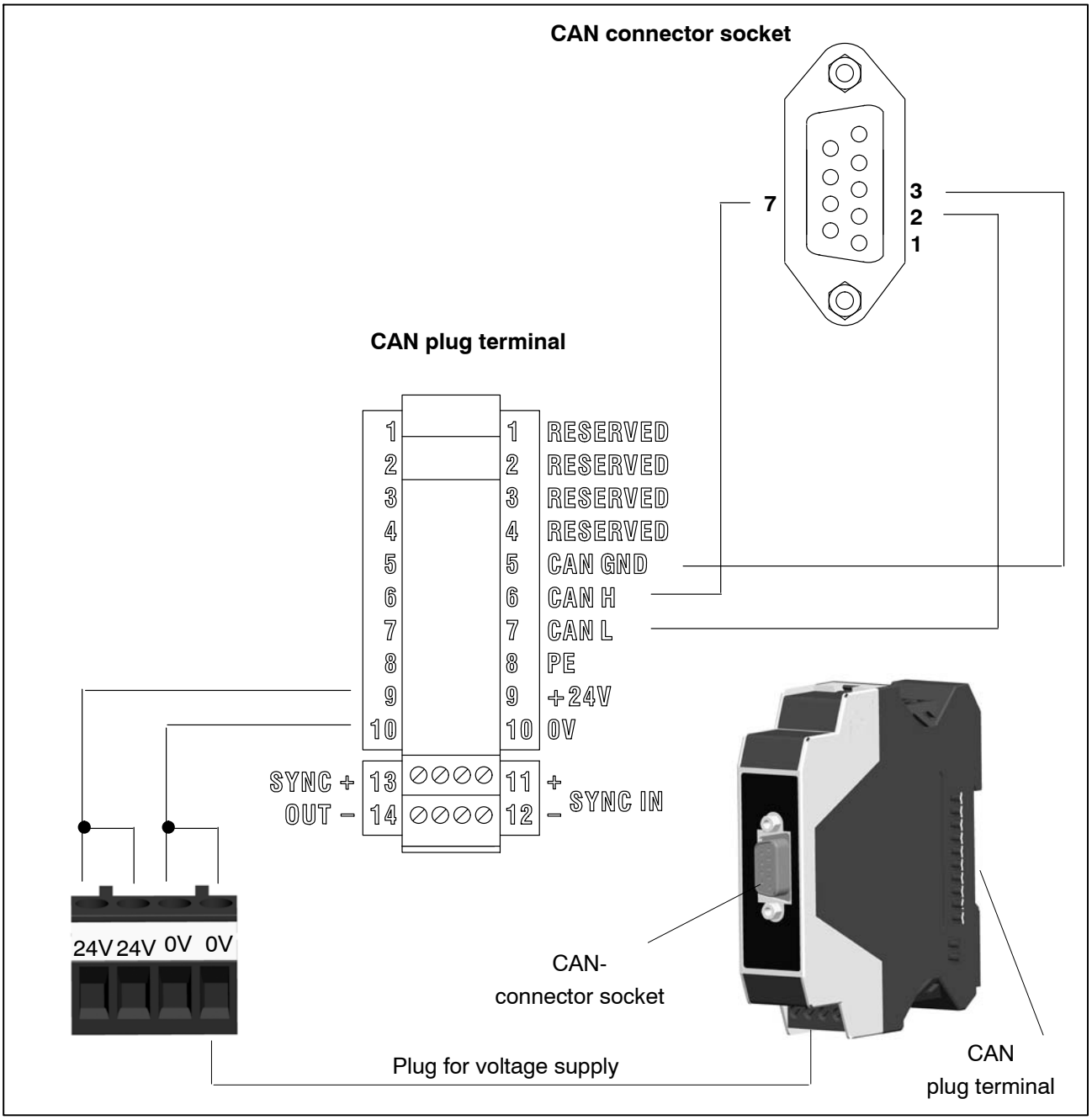


Fig. 4.1: DF30DN/DF31DN :pin assignment of the DF002 adapter module

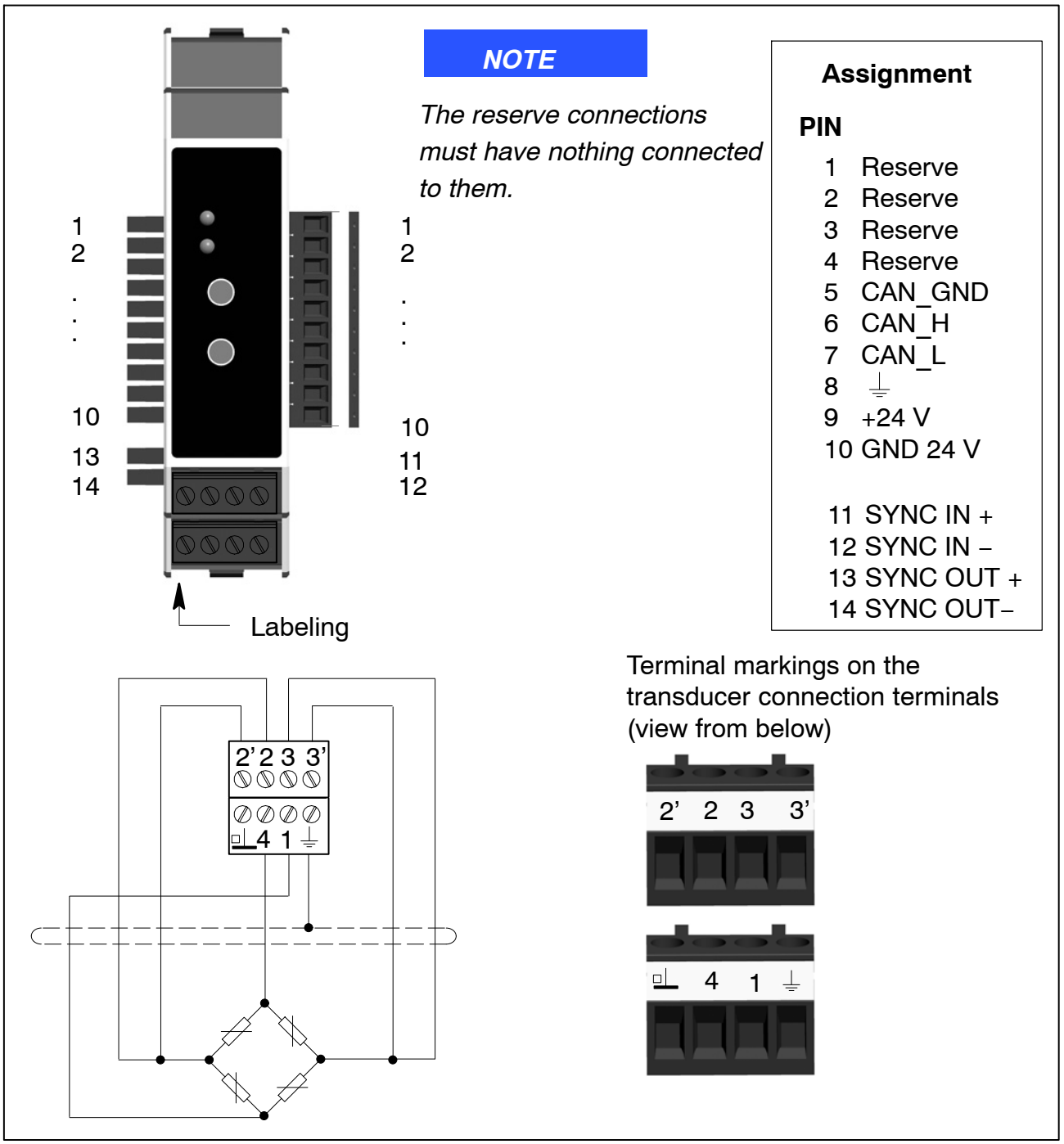


Fig. 4.2: Plug–in terminal assignment (single–shielded cable)

Terminal	function	Color (HBM cable)
1	measurement signal (+)	WH (white)
2	excitation voltage (–)	BK (black)
2'	sense lead (–)	GY (gray)
3	excitation voltage (+)	BU (blue)
3'	sense lead (+)	GN (green)
4	Measurement signal (–)	RD (red)
⏚	cable shield / grounding	

Transducer connection in six-wire configuration

The transducer connection terminals are coded with coding tabs, to prevent confusion when attaching them to the female connectors.

Six–wire circuitry is used for connection (with two sense leads).

Note

With double-shielded cables, the inner shield is connected to ground, the outer shield to the housing connection.

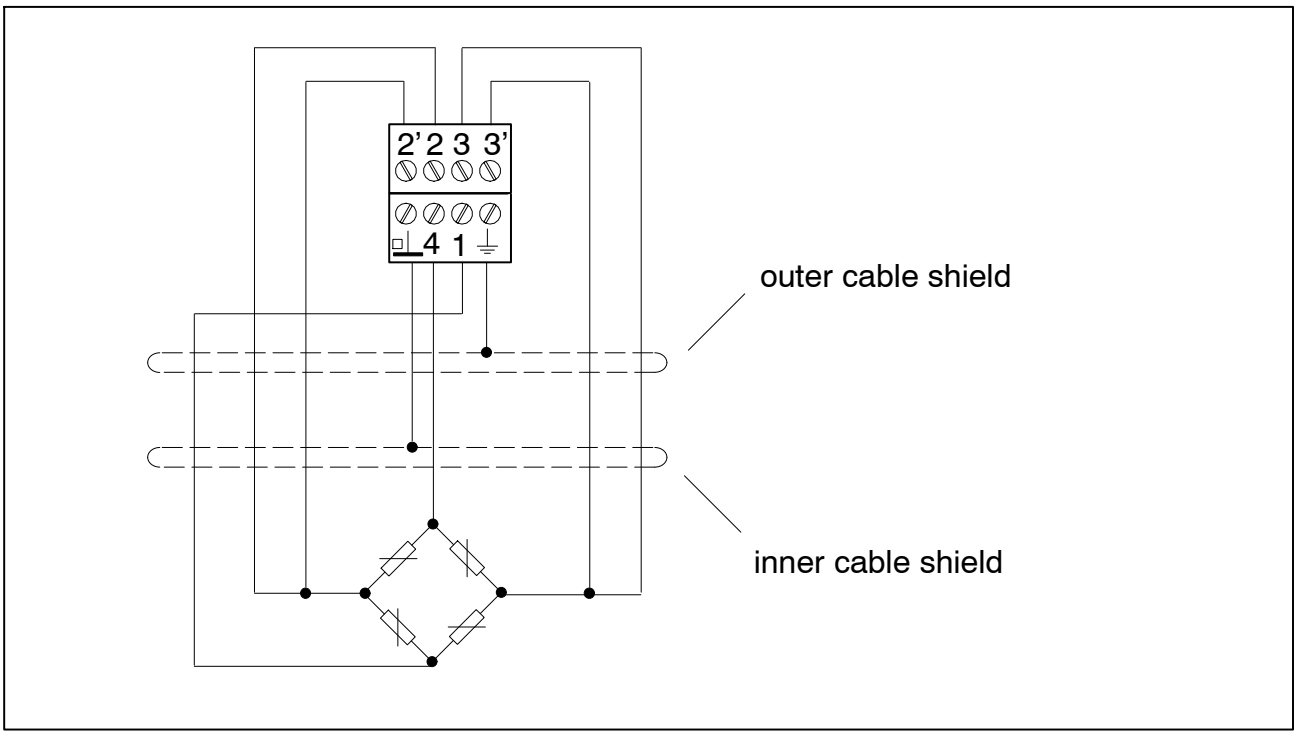


Fig. 4.3: Transducer connection in six–wire configuration

Transducer connection in four-wire configuration

When connecting a transducer in four–wire circuitry, the sense leads must be connected to the relevant bridge excitation line (PIN 2'–2 and Pin 3'–3) by jumpers, as otherwise a sensor error will be detected.

When connecting in four–wire circuitry, TEDS functionality is not available.

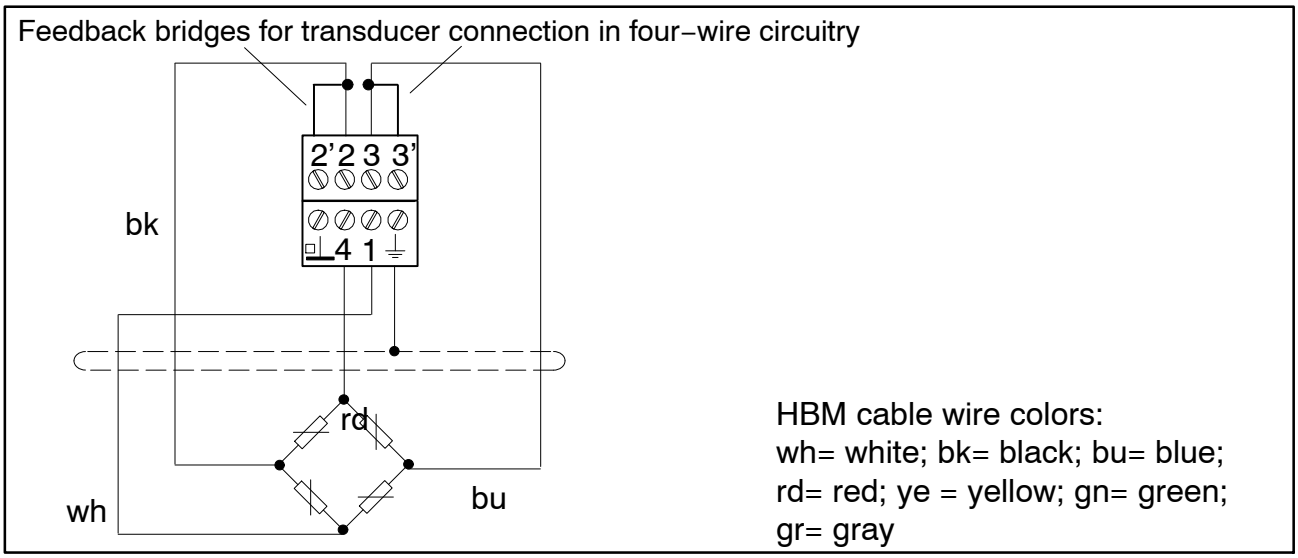


Fig. 4.4: Four−wire connection with feedback bridges

When connecting in four−wire circuitry, the connections for long lead compensation are missing. So line influences have to be calibrated in. This can be done by the digiCLIP Assistant in the ”2−point scaling” range.

Note

Use standard HBM cables for connecting the transducers. When using other shielded, low-capacitance measurement cables, attach the shield of the transducer cable to the cable shield connection in accordance with HBM Greenline information. The power supply for the digiCLIP modules must not come from a direct voltage network. We recommend a local voltage supply, in the control cabinet.

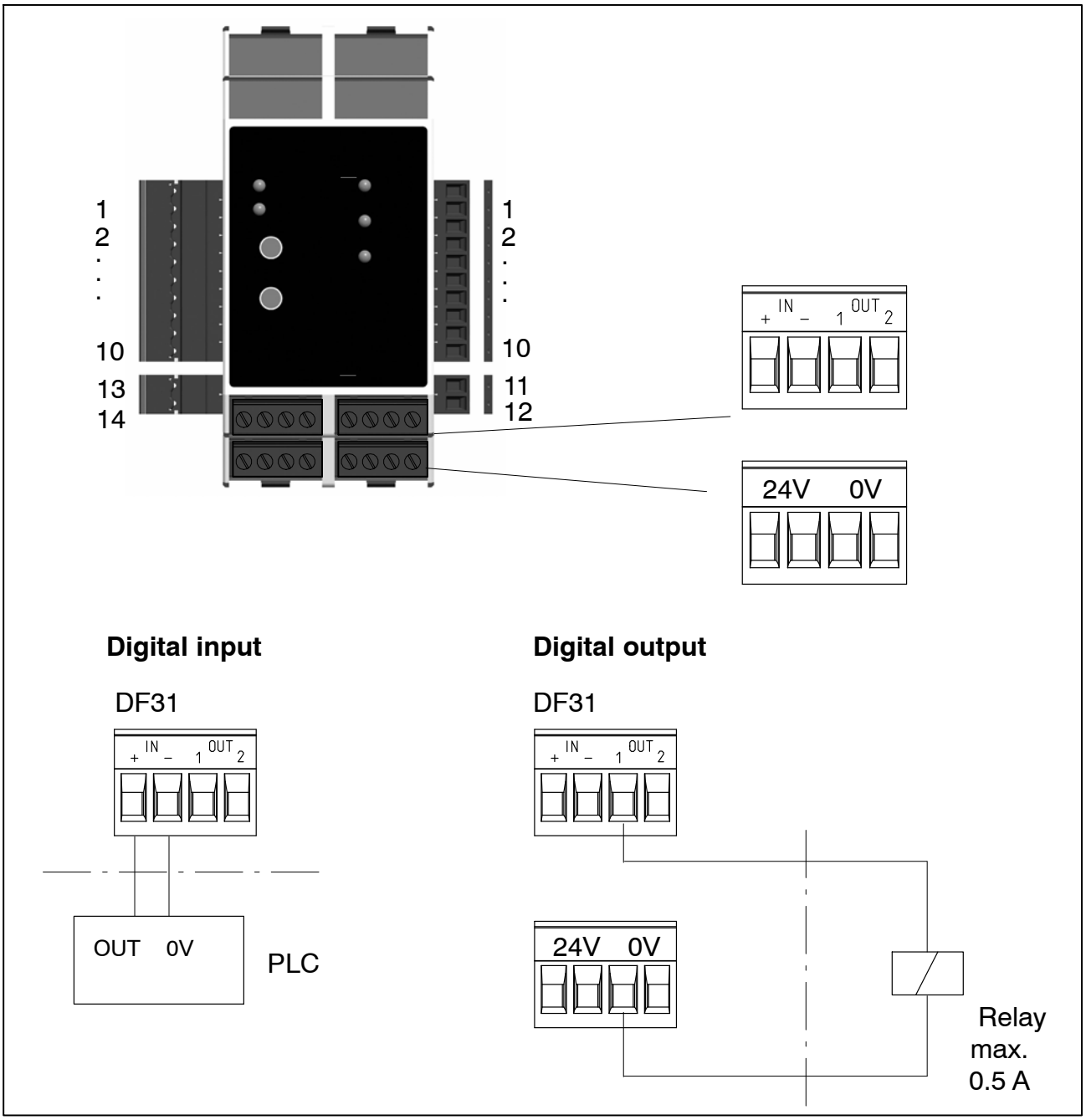
Connect the shield of the particular transducer cable via as short a lead as possible (< 5 cm). Alternatively, you can use the supplied cable holder, that also acts as strain relief. This ensures EMC protection.

Please also note:

- When connecting the leads, measures need to be taken to prevent electrostatic discharge.
- The relevant connection diagram is printed on the side of the housing.
- digiCLIP modules are designed for installation in enclosed, metal housings (such as a control cabinet); however, they can be operated without any additional housing.

4.1 Connecting the digital I/O

This function is only available in DF31DN.



The frontal terminals ”24V” are connected to the side bus terminals ”+24V” (Pin 9). The frontal terminals ”0V” are connected to the side bus terminals ”0V” (Pin 10).

4.2 Use with Zener barriers

When transducers are used in potentially explosive atmospheres, intrinsically safe measurement circuits (Ex II (1) GD, [EEX ia]IIC) have to be set up by connecting SD01A safety barriers (Zener barriers) to the digiCLIP. Similar to the digiCLIP modules, the safety barriers are also mounted on DIN rails. An ATEX test certificate must be available for the transducers that are used.

For use with Zener barriers, the excitation voltage has to be set to 1 V on the digiCLIP. For this purpose, use the ”Transducer – Excitation voltage” menu in the digiCLIP Assistant.

For more information on the design, mounting, and use of the safety barriers, please see the SD01A manual.

Note

TEDS transducer identification is not available for use with Zener barriers. Use with line lengths > 100 m and transducer resistances < 80 ohms is not permitted.

4.3 Synchronization of the carrier frequency

The first device (starting from the right) is used as the master when synchronizing. All the subsequent modules are automatically set as slaves and work at the carrier frequency of the first module. Should the connection between the modules be interrupted, the first module after the interruption is automatically set as the master and synchronizes the subsequent modules.

If the modules are divided among several rails, use the 2−pin synchronization connectors, 1−digiCLIP−ST; (see Fig. 4.5).

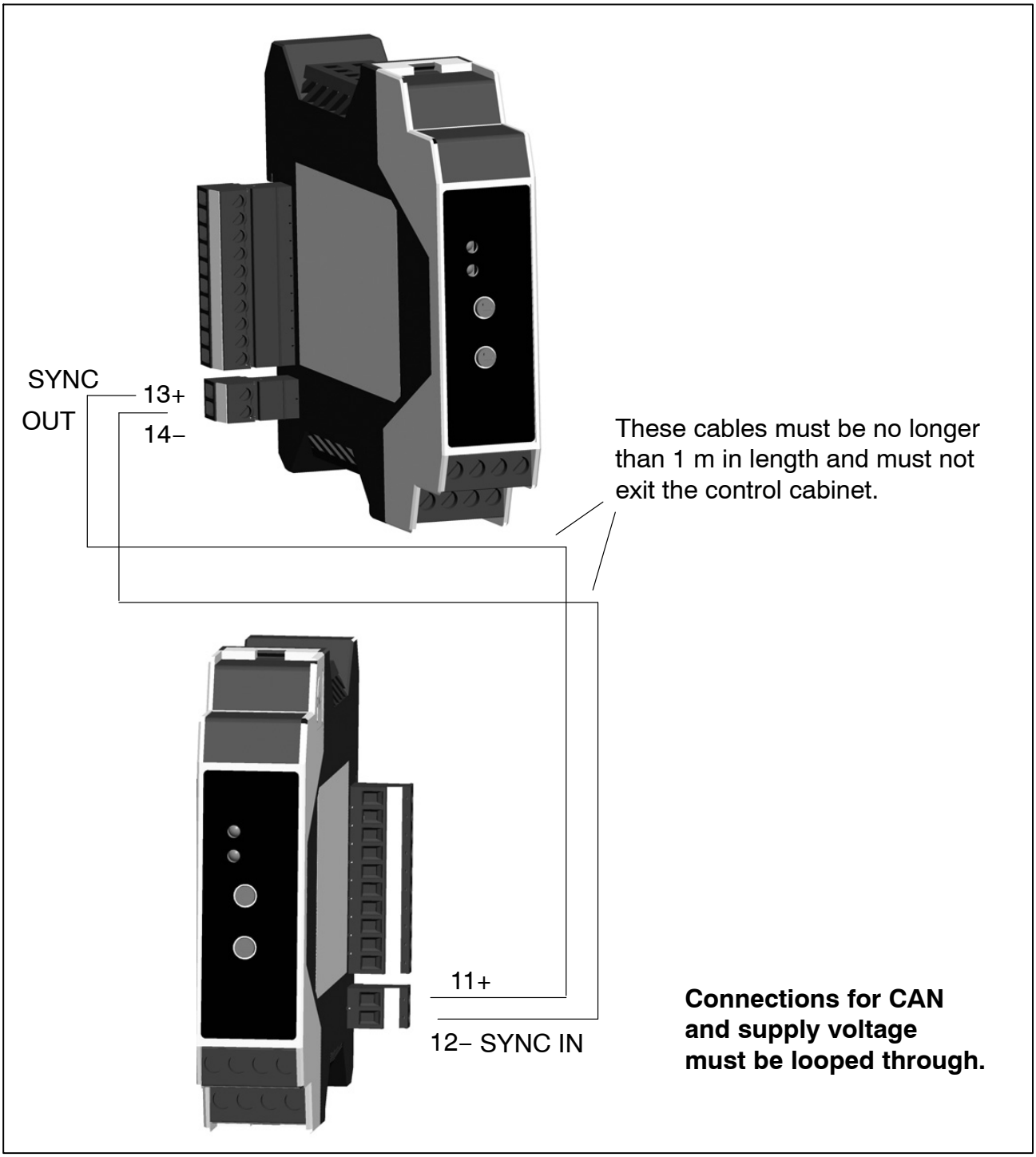


Fig. 4.5: Mounting at several levels (here :DF30DN)

Synchronizing:

Synchronizing is advisable for transducers with carrier frequency excitation when

- the transducer cables of several devices run side by side
- the measuring points are unshielded and are close together

Synchronization prevents differences in the carrier frequency causing disturbing superpositions. A maximum of 99 modules can be interconnected.

4.4 DeviceNet network installation

Located on the front of the DF002 connection module is a 9−pin D−Sub connection socket for the PROFIBUS connection. As an alternative, the PROFIBUS connection can also be made via the 10−pin connector terminal at the side.

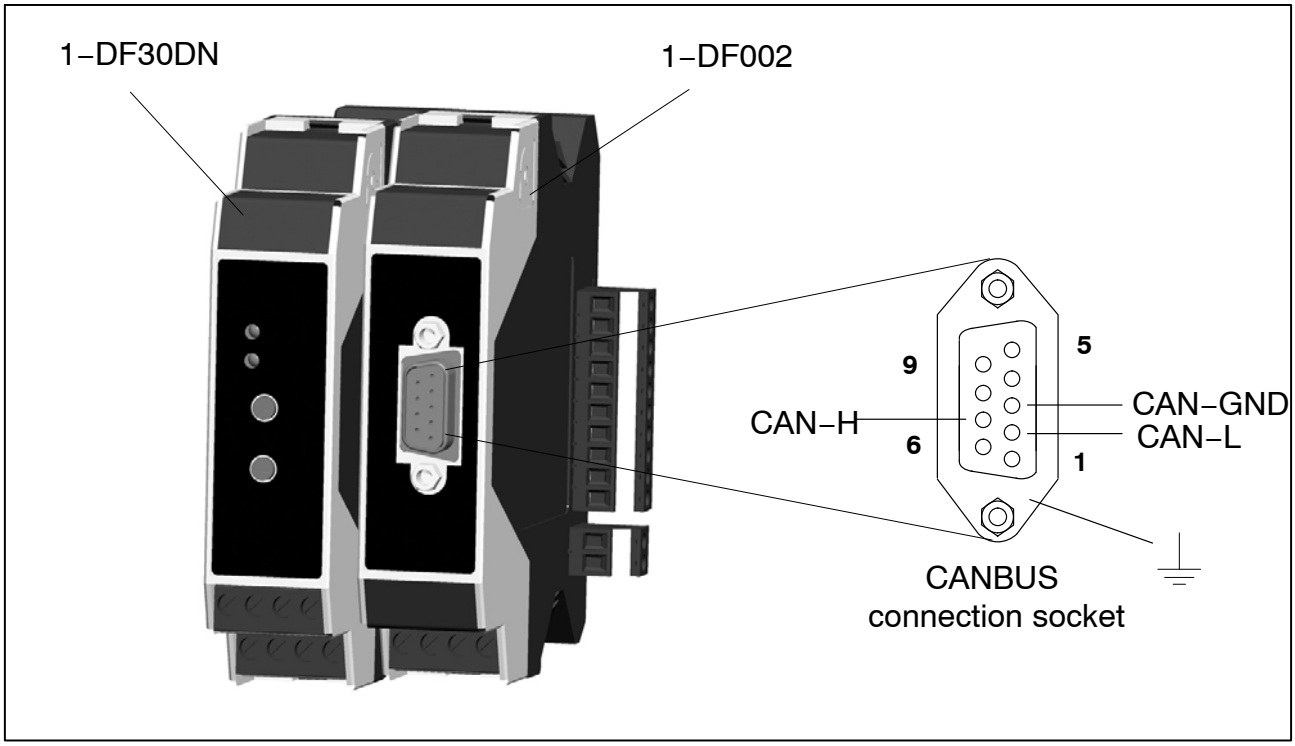


Fig. 4.6: CANBUS connector (9−pin D−Sub connection socket)

As many as 64 bus nodes can communicate with one another in a DeviceNet network at bit rates of 125, 250 or 500 kBit/s. In addition to the two signals for CAN-L and CAN-H data transmission, the DeviceNet cable also provides two lines for supplying DeviceNet bus nodes with a 24-volt operating voltage. Bus nodes can be designed to be bus-powered or externally supplied. A bus

topology is used for installation – with or without branches – and there are termination resistors at both ends. The value of the termination resistors (terminators) is 120 ohms.

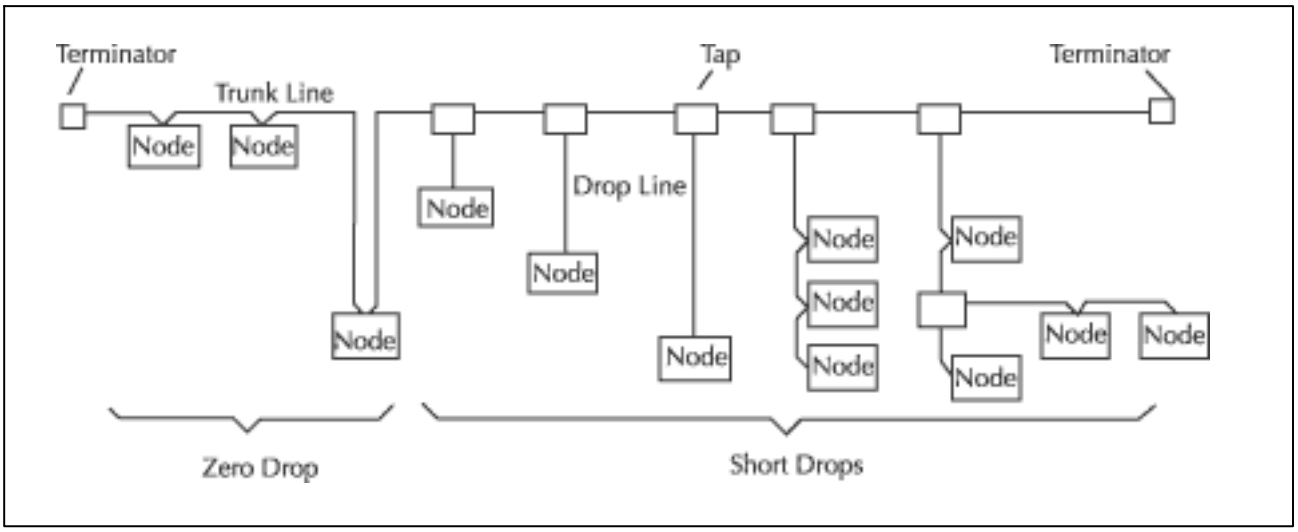


Fig. 4.7: DeviceNet bus topology

The maximum length of a DeviceNet cable depends on the selected cable type and the bit rate.

Line length	125 kbit/s	250 kbit/s	500 kbit/s
Total length with thick cable	500 m	250 m	100 m
Total length with thin cable	100 m	100 m	100 m
Total length with ribbon cable	380 m	200 m	75 m
Maximum stub line length	6 m	6 m	6 m
Maximum length of all stub lines	156 m	78 m	39 m

4.5 CAN bus line termination

The DeviceNet is connected via the 10–pin plug–in terminal. A maximum of 64 DeviceNet nodes can be connected in one bus segment – each with different DeviceNet addresses – (in accordance with the CANopen specification).

In the **first** and **last** bus nodes, the DeviceNet needs a termination resistor of 120 Ω (min. 1/4 W). The bus line must have no more than two termination resistors. The digiCLIP has an integrated termination resistor which is activated by a slide switch. The alternative is for the termination resistor to be connected to the connection terminals. When the digiCLIP is delivered, this slide switch is in the ”OFF” position.

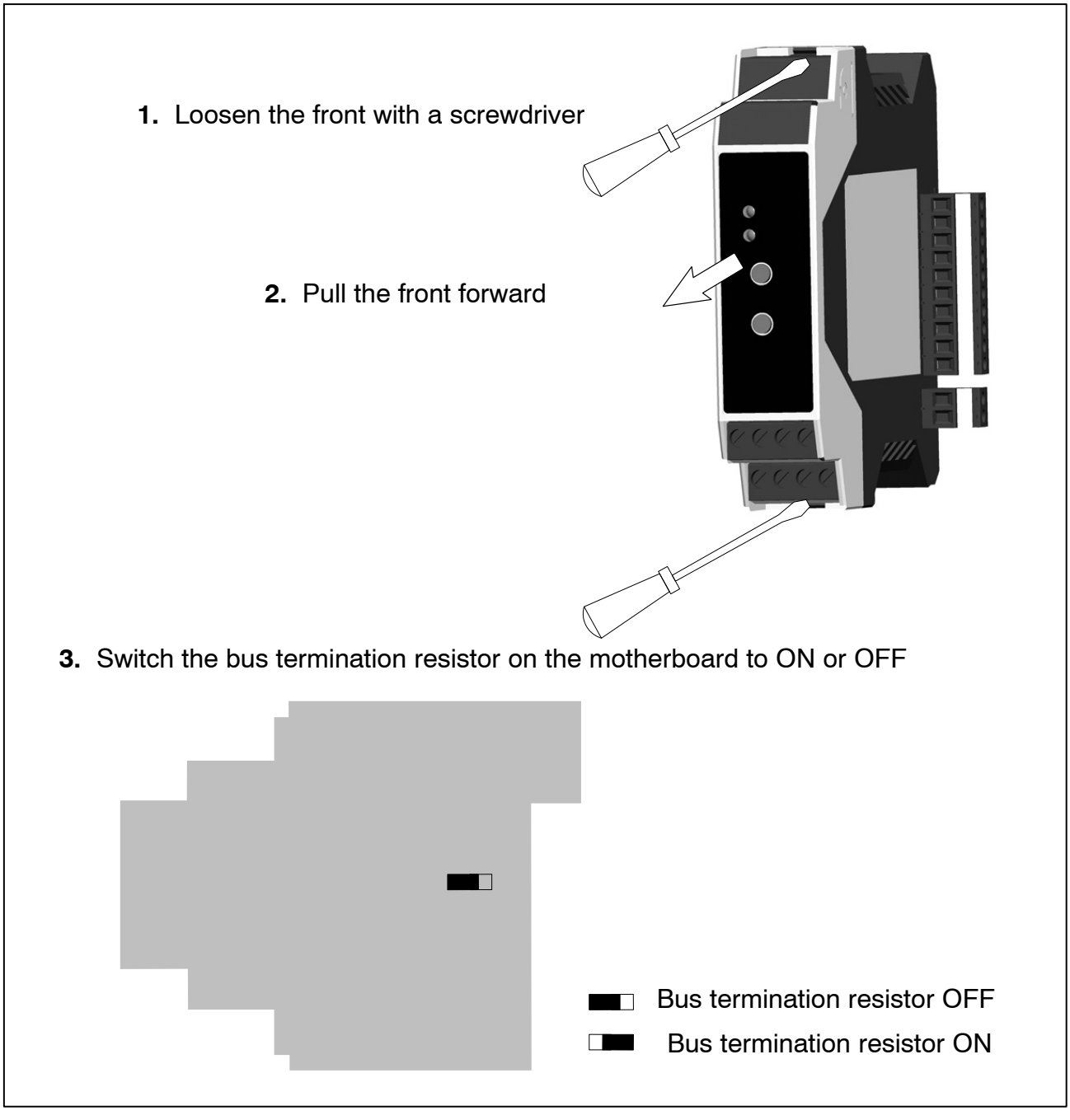
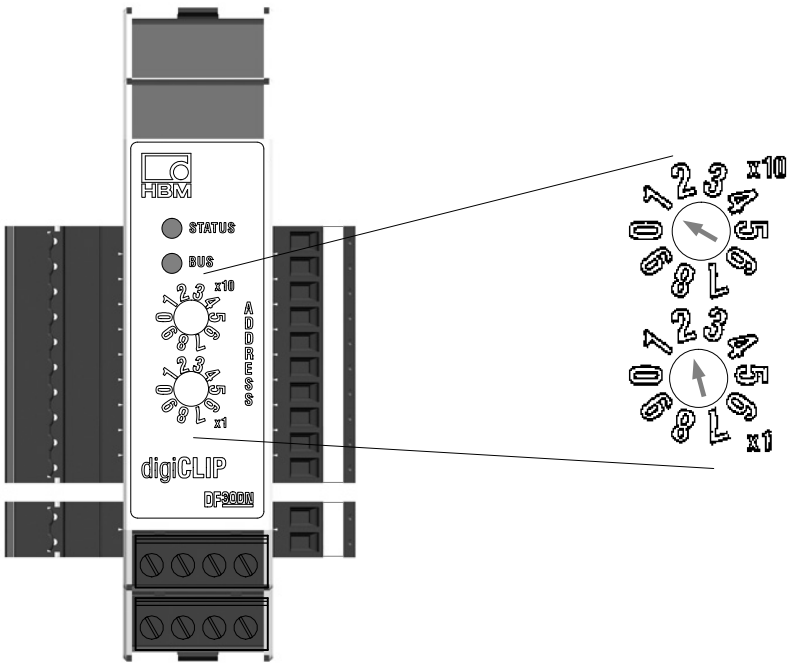


Fig. 4.8: Slide switch for the DeviceNet termination resistor

4.6 Selecting the module address

Address 0 to address 63 can be set as the module address.



Example:

Address 12 is set

4.7 Bit rate

The Assistant must be used when setting the bit rate. Unlike all the other entries, this variation is immediately stored in the device and activated the next time the device is started up. Should the bit rate have been changed in the CAN network, proceed as follows:

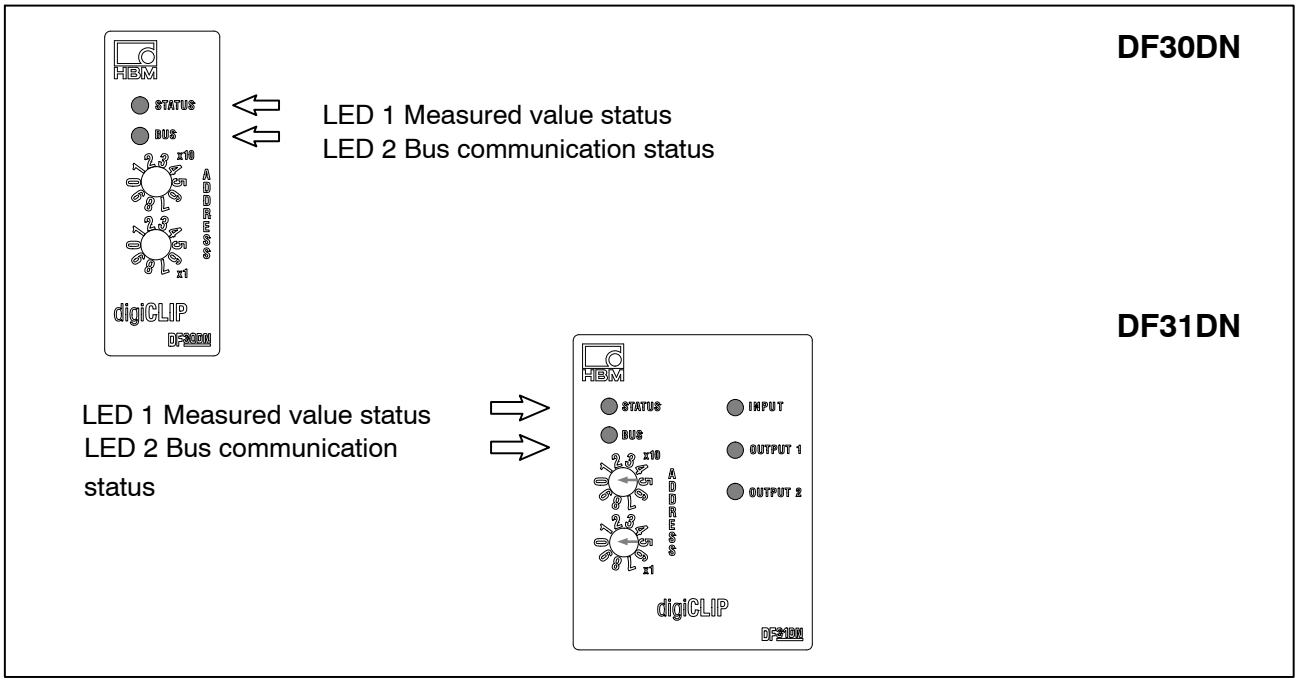
- Connect the digiCLIP
- Start up the digiCLIP
- In the top part of the window, under DeviceNet, select the network with the bit rate set on the digiCLIP (at the time of delivery: 125 kbit/s).

The bus LED (lower LED) on the digiCLIP indicates the set bit rate after the device is started up for approx. 1 second:

red = 500 kbit/s
yellow = 240 kbit/s
green = 125 kbit/s

4.8 Display LED status, error messages

Two LEDs show the operating states. The upper LED relates to measurement acquisition, the lower to communication.



STATUS LED (top): Measured value status	
Green	No error, normal operation, measured values valid.
Flashing green	No error, normal operation. However, the amplifier does not acquire the transducer signal, it acquires the internal reference signals
Orange	No error, normal operation, measured values valid, but out-of-range or limit value switch active.
Red	Error, measured values invalid. As there could be a number of different causes, you should use Device → Show device status, in the digiCLIP Assistant to call the status window and evaluate the detailed displays that are shown there.

BUS LED (bottom): Communication status	
Off	Power supply off, DupMacID-Check not ready yet.
Green	The digiCLIP is allocated to the Master, the connection(s) have “Established” status.
Flashing green	The digiCLIP is not allocated to the Master, the connection(s) do not have “Established” status.
Flashing red	The digiCLIP has “Recoverable fault” status, the connection(s) has/have a timeout.
Red	This is a critical, non-recoverable error, communication is not possible.

ATTENTION

If the two LEDs flash red, quickly and alternately, there is a firmware error in the flash memory area, because a firmware update was incomplete, for example. Transfer the firmware again (see Software update, Firmware update). The digiCLIP does not work.
If the two LEDS show permanently red, an internal error is stopping the digiCLIP working. Switch the digiCLIP off and then back on again, to test whether the error is still present. If the error keeps occurring, please contact HBM Technical Support.

5 Commissioning

Mount one or more digiCLIP modules and connect the transducers.

- Activate bus termination resistance for the first and last modules
- Connect the power supply
- Synchronization is performed automatically
- Set the address for each module; addresses must not be duplicated

As delivered, digiCLIP is assigned the address 63 and the bit rate 125 kbit/s. The bit rate can only be modified via the Assistant (see chapter 4.7).

5.1 Operation with the digiCLIP Assistant

The digiCLIP Assistant allows you to parameterize and scaling this measurement system, the display and measured value recording.

The software only shows devices of the digiCLIP product family. All other CAN nodes are ignored.

Procedure

- The digiCLIP must be ready for operation.
- Connect the PC's CAN bus interface to the digiCLIP (this can also be done while operation is ongoing).
- Should the PC not have its own CAN bus interface, you can use the CAN to USB adapter (1−digiCLIP-Setup).
- Make sure that the digiCLIP is not being parameterized from elsewhere at the same time (no further messages Transfer active)
- Start the digiCLIP Assistant.
- When you start the software for the first time, you must choose the CAN network to be used in a window. If you select *Use as standard*, this network will be chosen automatically the next time the system starts up.
- The digiCLIP Assistant finds all the devices and displays them in a list in the Devices area with their CAN address and serial number.
- Start a new search for connected devices via Interface → Devices.

Important

You can obtain the latest version of the relevant Assistant free of charge from http://www.hbm.com/support/ .

5.2 No devices can be found on the DeviceNet

- Check whether the DeviceNet interface is correctly installed on the PC; see the CAN adapter installation instructions and the operating requirements.
- The digiCLIP supports 125 kbit/s, 250 kbit/s and 500 kbit/s for DeviceNet. Check that the specified network uses a permissible bit rate.
- On the DeviceNet, verify for several devices that each digiCLIP has its own CAN address (that there are no duplicate addresses in the network).
  The upper switch on the digiCLIP gives the more significant digit: a setting of 1 above and 2 below corresponds to the decimal address 12.
- Check the digiCLIP address: only addresses between 0 and 63 are permissible, even if other addresses can be set by the rotary switches. At the time of delivery, the digiCLIP has address 63, and uses 125 kbit/s.
- Verify that you are using the correct bit rate (also called the baud rate): the bus LED (lower LED) on the digiCLIP indicates the set bit rate after the device is started up for approx. 1 second: red = 500 kbit/s, yellow = 240 kbit/s and green = 125 kbit/s.
- Check that the termination resistors on the DeviceNet are correctly set: the resistors for the first and last devices on the bus (or PC) must be activated (DIP switches on the digiCLIP). If you are using more than one device, no resistors can be activated on any of the other devices.
- Call the PCANStat program from Peak (Windows **Start** menu, **Programs → PCAN**).
  The program indicates which nodes are available on the CAN bus at the PC end. The entry PCANLight_USB_Client should be available for DeviceNet, the entry HBM_Client for CANopen, and the CAN network selected for CANopen should be displayed.
  It will also be obvious from the status for the PC CAN interface whether the CAN bus is working correctly *(OK)*. BUS HEAVY, for example, can also be caused by a faulty connection. In this situation, check all the cable connections.

NOTE:
The display in the PCANStat window is only updated when the data is transferred. Therefore, once changes have been made, find the devices, so that you can recognize the changes, or reset the CAN bus.

- Call the PCANStat program from Peak (Windows **Start** menu, **Programs →PCAN**).
  Right-click on the area for the PC's CAN interface and reset the CAN bus. Then, in the digiCLIP Assistant, find the devices again: **Interface → Find devices**.

More information can be found in the online Help on the system-CD.

6 Parameterizing via the digiCLIP Assistant

First check that the sensor connection is healthy: Open the Status window by double–clicking on the displayed measured value or with *Device → Show device status*. Red LEDs for *Sensor connection* indicate whether and if so which wiring faults exist.

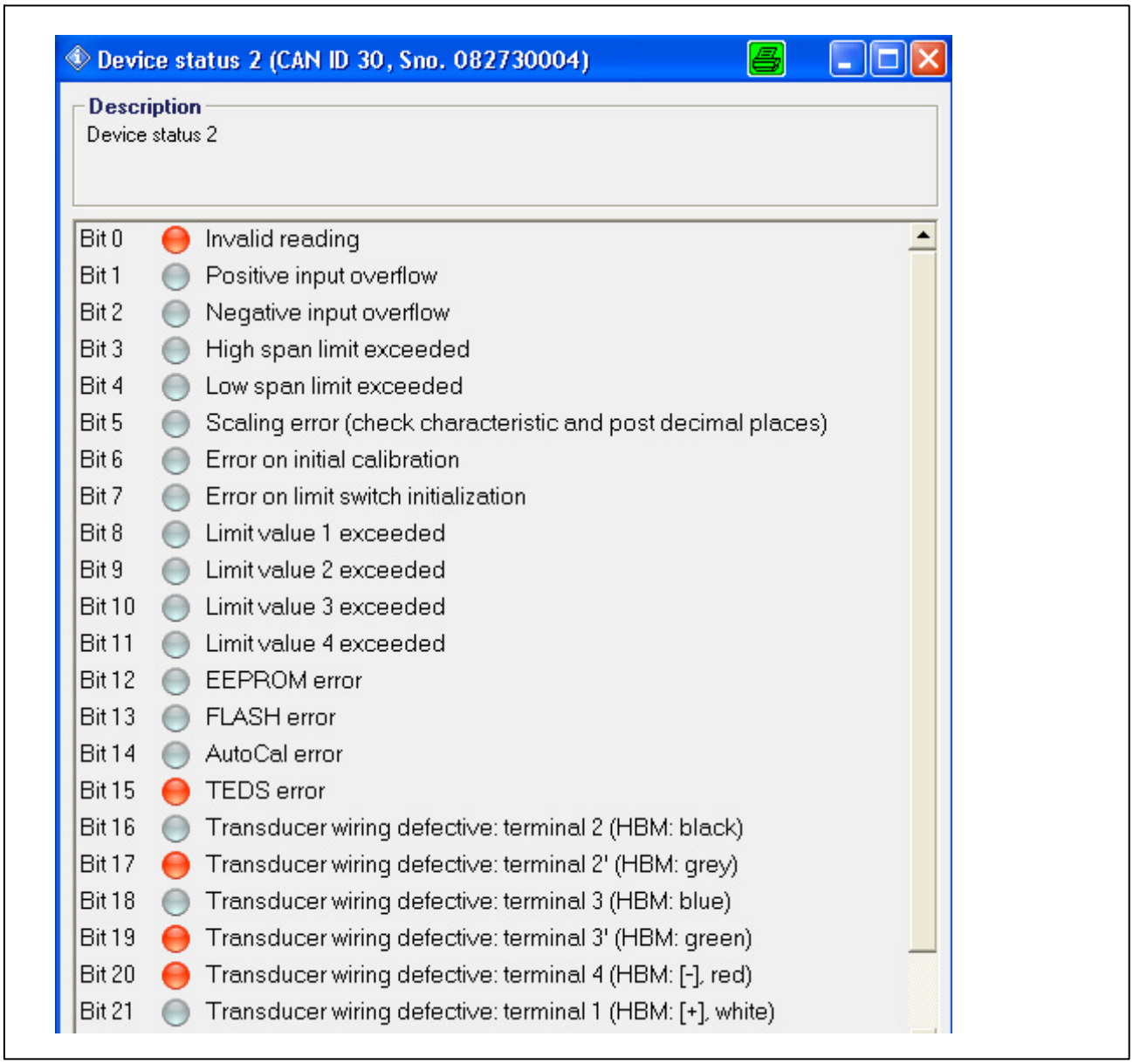


Fig. 6.1: Assistant: Device status

Then use the Assistant menus to set all the other device parameters.

Extensive Help is also available in the Assistant. The parameters are then present in the digiCLIP RAM.

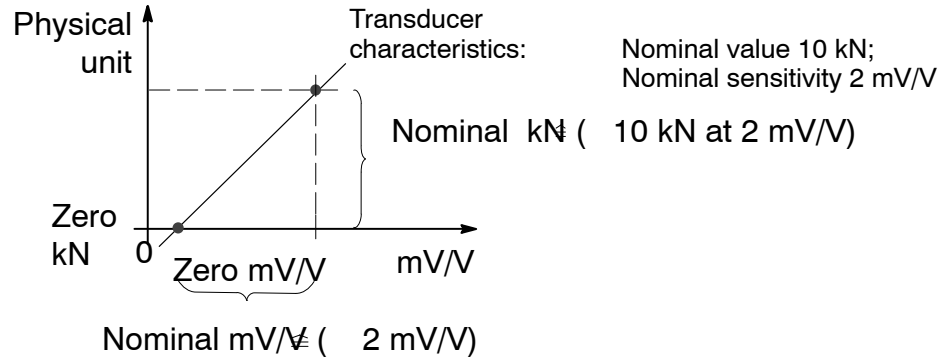
To make them available again after a power failure, they still have to be saved in the digiCLIP EEprom memory (Assistant dialog: Save/load parameters –> Save parameters in device).

After a loss of voltage, or after switching the digiCLIP back on, all the parameters last available in the EEprom are automatically reloaded into the device (RAM).

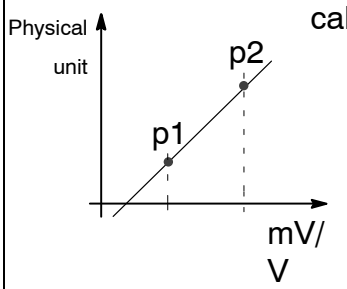
Note

Apart from the factory settings, the digiCLIP only has one parameter set (measurement program) that can be stored in the device. But additional parameter sets can be stored on a PC and then reloaded, using the Assistant. There is no offline mode, that is, creating / changing a parameter set without a connected device.

6.1 Clarification of significant parameters

Scaling	**Scaling in accordance with transducer characteristics** 

Alternative: 2–point scaling



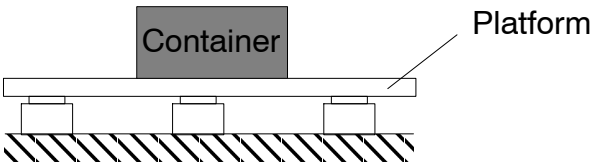
Example: A calibration weight of 4 kg is used to calibrate a 10 kg load cell

1. Relieve the load on the transducer
   Measure point 1 0.0457 mV/V Char. curve point 1
   enter 0 kg physical
2. Load transducer with 4 kg
   Measure point 2 0.873 mV/V
   Char. curve point 2
   enter 4 kg physical

Taring / zeroing

Difference between taring and a zero balance: A zero balance (>0<) affects the gross and the net value. Taring (>T<) only affects the net value.

The difference between a zero balance and taring is made clear in this example:

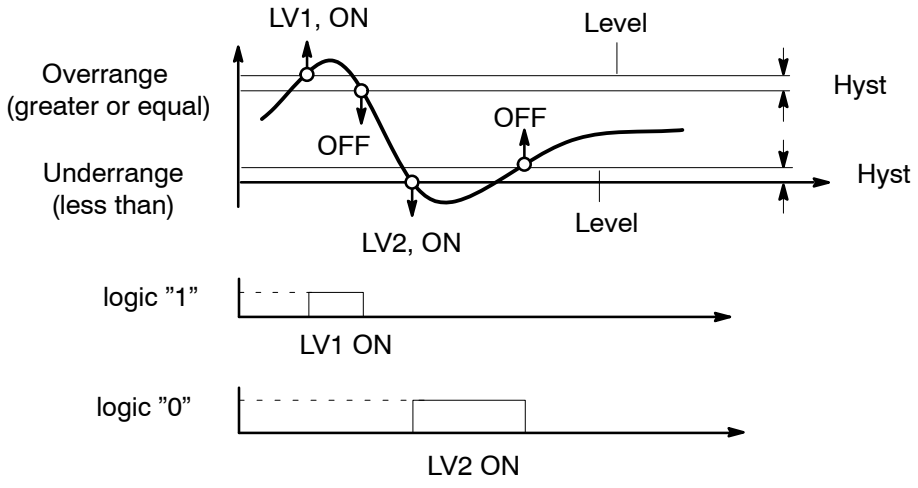


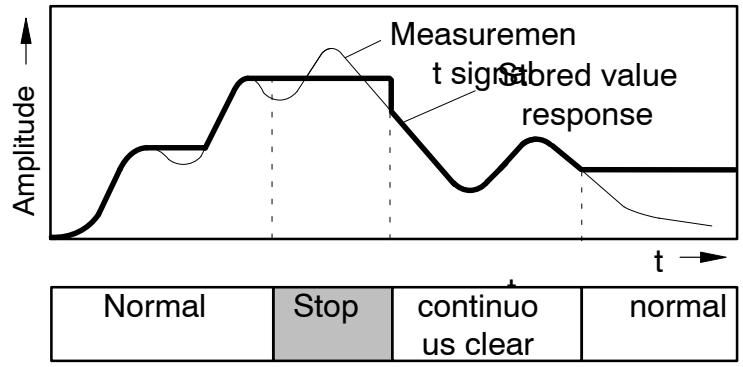
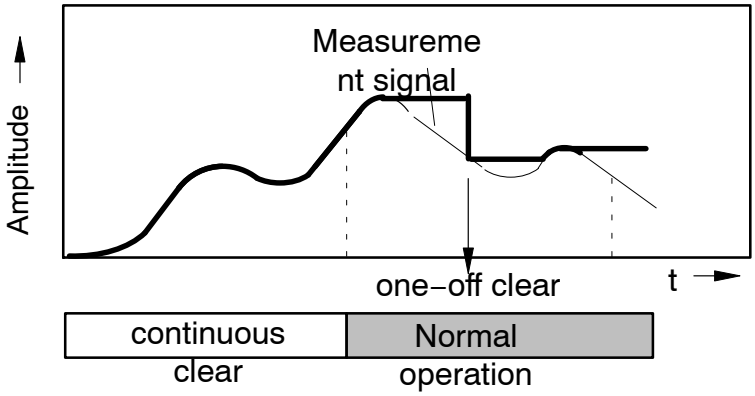
Weighing steps	Action	Display Gross	Net
Put on the platform (35 kg)	> 0<	before 35 kg after 0 kg	before 35 kg after 0 kg
Put on the container (8 kg)	> T<	before 8 kg after 8 kg	before 8 kg after 0 kg

Filters / frequencies	0.05 Hz 1 Hz 20 Hz 0.1 Hz 2 Hz 50 Hz 0.2 Hz 5 Hz 100 Hz 0.5 Hz 10 Hz
Autocal	The Autocal function briefly interrupts the measurement function to link the amplifier input with an internal reference. This evens out errors caused by ageing and temperature. This function is executed **once** on demand.

Limit value switches 1...4	The choice of source for the limit value signal is: gross, net, peak value max/min/peak–to–peak

Limit value functions and parameters



<table>
<tr><td colspan="2">The choice of source for the peak value signal is: gross, net</td></tr>
<tr><td colspan="2">The peak value can be cleared.</td></tr>
<tr><td rowspan="2">Peak values</td><td>Example 1
 
</td></tr>
<tr><td>Example 2
 
</td></tr>
</table>

6.2 Parameterizing with TEDS

6.2.1 Electrical connection with TEDS

TEDS stands for ”Transducer Electronic Data Sheet”.

A transducer with an electronic data sheet as defined in the IEEE 1451.4 standard can be connected to the digiCLIP, making it possible for the amplifier to be set up automatically. A suitably equipped amplifier imports the transducer characteristics (electronic data sheet), translates them into its own settings and measurement can then start.

Six–wire circuitry must be used for TEDS to be connected.

6.2.2 Parameterization with TEDS

If a transducer with TEDS, containing the parameterization data for a full bridge is connected, this can be used to parameterize the amplifier automatically.

When the digiCLIP is activated, it automatically detects whether a TEDS is connected. When the transducer is replaced in the activated state, the new TEDS is also detected automatically.

Check the ”Always use TEDS” box to monitor TEDS functionality and protect scaling from manual intervention. If a transducer is being used without TEDS, this checkmark must be cleared.

To enable the data stored in the TEDS to be used for scaling, a setting must be made in the digiCLIP to indicate the physical unit in which the measured values are to be displayed. The scaling values stored in the TEDS are then automatically converted to the required unit. By specifying this conversion unit, scaling can also take place to a power of ten (e.g. ”kN”) or English units can be used both for the display and in the TEDS.

In the digiCLIP Assistant, in the ”TEDS” area, choose the desired conversion unit from the selection menu. If instead you want to use the unit stored in the TEDS directly, set this value to ”(auto)”.

When the TEDS is activated, its scaling data will be read out and converted to the required physical unit. Should the unit stored in the TEDS and the required conversion unit be incompatible because they describe different quantities (e.g.: torque transducer connected, conversion unit is ”N”), a error message is generated and scaling does not take place.

If automatic activation of TEDS is set (checkmark: ”Always use TEDS”) the TEDS is read out automatically and scaling performed accordingly, whenever the digiCLIP supply voltage is turned on or a new transducer is connected in the on state.

If a scaling error is reported once the TEDS is activated, the reason may be that the value range specified by the two characteristic curve points is so great or so small, that the measured values cannot be displayed with the set decimal places. You then need to adapt the number of decimal places in the ”Scaling” area. It may possibly help to change to a different power of ten, such as. ”N” after ”kN”. To obtain more information, click on ”TEDS error status” in the digiCLIP Assistant. If you have not connected any transducers with TEDS, make sure that the ”TEDS always available” box is not checked. For an accurate analysis, it is advisable to display the data stored in the TEDS. To do this, in the digiCLIP Assistant, click on ”Details” in the ”TEDS” area.

Example 1:

Torque transducer connected, display required in kilonewton meters, ”kNm”

Stored in the TEDS are:

Minimum Force/Weight	1.0 Nm
Maximum Force/Weight	2500.0 Nm
Minimum Electrical Value	0.1 mV/V
Maximum Electrical Value	1.5 mV/V
Reference unit set in the digiCLIP (”kNm”)	03560000 (hex)

After scaling by TEDS, the scaling points are set as follows:

Char. curve point 1, physical	0.001 kNm
Char. curve point 1, electrical	0.1 mV/V
Char. curve point 2, physical	2.5 kNm
Char. curve point 2, electrical	1.5 mV/V

Example 2:

Force transducer connected, display required in English pounds, ”lb”

Stored in the TEDS are:

Minimum Force/Weight	1.0 N
Maximum Force/Weight	1000.0 N
Minimum Electrical Value	–0.1 mV/V
Maximum Electrical Value	4.0 mV/V
Reference unit set in the digiCLIP (”lb”)	00EF0001 (hex)

After scaling by TEDS, the scaling points are set as follows:

Char. curve point 1, physical	0.225 lb
Char. curve point 1, electrical	–0.1 mV/V
Char. curve point 2, physical	224.81 lb
Char. curve point 2, electrical	4.0 mV/V

The data for the minimum and maximum excitation voltage in the TEDS is also checked. If the excitation voltage is too high or too low, it is automatically adapted in the digiCLIP.

If, instead of using the digiCLIP Assistant, you are parameterizing directly by SDO commands, you must use Object 3576 to set the required conversion unit before activating the TEDS. The units available to you correspond to the selection list provided by the digiCLIP Assistant and are defined in accordance with CiA DR303–2. English units of measurement are supplied in accordance with the table below. If value = ”00000000” is set, the unit used for conversion is the one stored in the TEDS.

When the TEDS is successfully activated, objects 3231 and 6131 are also changed accordingly.

The CAN objects for using TEDS are located in Section 7.3.9 .

Note

If several transducer full bridges are connected to a digiCLIP amplifier input in parallel, their TEDS data should not be used for automatic scaling, as in this case, the distribution of the forces could lead to unwanted scaling. Clear the "Always use TEDS" checkmark.

6.2.3 Parameters of the required physical conversion unit

Value (hex)	Required unit	Conversion
FA4B0000	µg	$1 \cdot 10^{-6}$ g
FD4B0000	mg	$1 \cdot 10^{-3}$ g
004B0000	g	
00020000	kg	
03020000	t	1000 kg
00210000	N	
03210000	kN	1000 N
06210000	MN	$1 \cdot 10^{6}$ N
09210000	GN	$1 \cdot 10^{9}$ N
00EF0001	lb	4.44822 N
00EE0001	oz	0.278 N
00ED0001	kgf	9.8 N
FE560000	Ncm	0.01 Nm
00560000	Nm	
03560000	kNm	1000 Nm
06560000	MNm	$1 \cdot 10^{6}$ Nm
00EA0001	ozf–in	$7.06 \cdot 10^{-3}$ Nm
00E90001	ozf–ft	$84.73 \cdot 10^{-3}$ Nm
00E80001	lbf–in	1.12 Nm
00E70001	lbf–ft	1.35 Nm
00E60001	in oz	$7.06 \cdot 10^{-3}$ Nm
00E50001	ozf–ft	$84.73 \cdot 10^{-3}$ Nm
00E40001	in lb	$1.12 \cdot 10^{-1}$ Nm
00E30001	ft lb	1.35 Nm
004E0000	bar	1·105 Pa
034E0000	kbar	1000 bar
FD4E0000	mbar	100.0 Pa
00220000	Pa	
02220000	hPa	100.0 Pa
03220000	kPa	1000 Pa
06220000	MPa	$1 \cdot 10^{6}$ Pa
09220000	GPa	$1 \cdot 10^{9}$ Pa
00AB0000	psi	6894.757 Pa
00010000	m	
FD010000	mm	$1 \cdot 10^{-3}$ m
FE010000	cm	$1 \cdot 10^{-2}$ m
FA010000	µm	$1 \cdot 10^{-6}$ m
00EC0001	in	$25.4 \cdot 10^{-3}$ m
00EB0001	ft	0.3048 m

Value (hex)	Required unit	Conversion
00010300	m/s	
00EB0301	fps	0.304 m/s
00014700	m/min	1.66 m/s
FD550000	mm/s^2	$1 \cdot 10^{-3}$ m/s^2
00550000	m/s^2	
00EB5701	ft/s^2	$3.048 \cdot 10^{-1}$ m/s^2
00EC5701	in/s^2	$2.54 \cdot 10^{-2}$ m/s^2
FA010100	µm/m	$1 \cdot 10^{-6}$ m/m
FE000000	%	
FD000000	‰	0.1 %
FA000000	ppm	$0.1 \cdot 10^{-3}$ %

7 DeviceNet interface description

The digiClip module has an inbuilt DeviceNet interface which can be used both for transmitting measured values and for module parameterization. The choice of bit rate is between 125 kBits/s, 250 kBits/s and a maximum of 500 kBits/s. The interface protocol is adapted from the DeviceNet standard.

DeviceNet was developed by Rockwell Automation and the ODVA (Open DeviceNet Vendor Association) as an open fieldbus standard, based on the CAN protocol. DeviceNet is standardized in European standard EN 50325.

ODVA is responsible for specifying and maintaining the DeviceNet standard http://www.odva.org.

DeviceNet is part of the CIP-based network range that includes ControlNet and EtherNet/IP. The CIP (Common Industrial Protocol) is the common application layer of these three industrial networks. DeviceNet, ControlNet and Ethernet/IP are therefore well-matched, and provide the user with a hierarchized communication for process control level (EtherNet/IP), cell level (ControlNet) and field level (DeviceNet).

DeviceNet is an object-oriented bus system based on the Producer/Consumer model. DeviceNet devices can be a Client (Master) or a Server (Slave), or both. Clients and Servers can be Producers, Consumers or both.

7.1 DeviceNet communication

The DeviceNet protocol is an object-oriented protocol. It is typically used for networking sensors and actuators with the primary automation devices (PLC, IPC). The bandwidth of devices that can be connected via DeviceNet ranges from a simple light barrier to complex automation components.

7.1.1 Predefined Master/Slave Connection Set

The so−called “Predefined Master/Slave Connection Set” is specified for simple DeviceNet Slave devices. This subset of the DeviceNet protocol simplifies the transfer of I/O data between an automation system (PLC) and the distributed peripherals (Slaves).

Implicit and explicit messages, polled I/O, multicast polled I/O and bit−strobed I/O messages from Master to Slave are supported, as are change-of-state/cyclic I/O messages from Slave to Master.

Important

The digiClip module supports:

1. The **"Group2 only, predefined master/slave connection set"**.
2. The addressing scheme 8/8. i.e. classes, instances and attributes are addressed as 8-bit values.

7.1.2 Module properties

The module properties are determined by the Vendor ID, DeviceType and Product code:

Vendor ID	0 x 389,905 HBM
DeviceType	0x00, generic device
Product code	DF30DN = 0x902,2306

7.1.3 EDS files

The module scope of supply includes an EDS file which describes the module properties.

Each manufacturer of a DeviceNet device provides an electronic data sheet for the device, the so-called EDS file. The EDS file contains all the device communication parameters and available objects.

The DeviceNet configuration tool reads the EDS files of the devices in the network and calculates the configuration data, which is then loaded to the DeviceNet nodes. The EDS files for the digiCLIP DeviceNet modules can be found on the digiCLIP system-CD or at www.hbm.com/support.

7.1.4 Connection types

DeviceNet is a connection-oriented protocol. The digiClip module supports the following connection types:

explicit messages	Acyclical service
polled messages	Cyclical service (I/O-message)
bitstrobe messages	Cyclical service (I/O-message)
COS/cyclic messages	Cyclical service (I/O-message)

The various connection types are prioritized via CAN-Identifier. The connections must be parameterized with a suitable software tool; this is not part of the module scope of supply.

7.1.5 Explicit Message

The Explicit Message is used to transmit acyclic data such as e.g. individual measured values and parameters. The message always consists of a request and a response. The Explicit Message format is fixed.

7.1.6 Polled Message

The Polled Message is used to transmit I/O (cyclic) data. The PLC transmits the output data to the module and receives in response the input data. The Polled Message format is HBM-specific. It is possible to change the message format using so-called Assemblies. Information about the available formats can be found in the appendix.

7.1.7 Bitstrobe Message

The Bitstrobe Message is used to transmit I/O (cyclic) data. The PLC can query single or all modules via a command. These then send the input data to the PLC. Transmission of output data to the Digiclip modules is not possible. The Bitstrobe Message format is HBM-specific. It is possible to change the message format using so-called Assemblies. Information about the available formats can be found in the appendix.

7.1.8 COS/cyclic Message

The COS/cyclic Message is used to transmit I/O (cyclic) data. The module can be parameterized with a suitable software tool so that it can transmit the Message in the event of a status change or cyclically. It is possible to change the message format using so-called Assemblies. Information about the available formats can be found in Chapter 7.4.

7.1.9 Assemblies

The I/O data format can be switched, either via the so-called Assembly Object or the HBM class 199 (0xc7). The available formats can be found in the appendix.

7.1.10 Assembly Object

This object describes the Assemblies that are available:

100-102 Assemblies for Poll Output (PLC–>Digiclip)
120-131 Assemblies for Poll Input (Digiclip–>PLC)
140-143 Assemblies for BitStrobe Input (Digiclip –>PLC)
160-163 Assemblies for COS/cyclic Input (Digiclip –>PLC)

The attributes in these objects describe:

- The number of attributes in the Assembly
- The DeviceNet path of the Attributes
- The IO data

Class	Instance	Attribute	Access	Description	Data type
4	100-163	1	R	Number of attributes in the Assembly	U8
4	100-163	2	R	Device path	EPATH
4	100-163	3	RW	Assembly data	ARRAY of U8

The format of the IO data (Polled, BitStrobe and COS/Cyclic connection) can be switched by writing the path of the relevant Assembly Object to the path variable of the Connection Object.

Example:

The user wants to transmit the gross measured value_I32 and the status1_U8 via the Poll connection.

The relevant Assembly Object is Class 4, Instance 105. **(Table 6.2.6).**

Switching is implemented by writing the path information:

0x20,0x04,0x24,0x69,0x30,0x03

in the ConsumedConnectionPath of the Poll connection. Class 5, Instance 2, Attribute 14.

Significance:

0x20,0x04	Class 4
0x24,0x69	Instance 105
0x30,0x03	Attribute 3 Assembly data

It is only permitted to write such paths that are defined in the Assembly Object.

7.1.11 HBM Assembly Object

Class	Inst	Attribute	Get/Set	
199	1	1	Get/Set	Poll Output Assembly
199	1	2	Get/Set	Poll Input Assembly
199	1	3	Get/Set	Strb Input Assembly
199	1	4	Get/Set	Cos Input Assembly

The formats of the individual I/O connections can be switched via the HBM Class 0xc7 (199). Information about the available formats can be found in the appendix.

7.2 Object model for the digiCLIP DF30/31DN module

DeviceNET uses an Object model in which all functions and data of a device are described.

Object	Class	Instance	Description
System Objects			
Identity	0x01	1	Identity: VendorID, P-Number
MessageRouter	0x02	1	Enables the forwarding of Messages
Communication Objects			
DeviceNet	0x03	1	MacId, Bitrate etc.
Assembly	0x04	100-123	Enables switching of the I/O connection
ConnectionClass	0x05	1-5	Describes the explicit and I/O connections and enables their parameterization
Acknowledge Handler	0x2b	1	Coordinates reception of message acknowledgements
User-specific Objects			
VendorSpecific	0x64-0xc7	1-255	HBM-specific Objects, see Chapter **6.3**

Access to these objects is only implemented connection-oriented. Connections must be set up between the nodes and connection objects parameterized. At completion of communication, the connection must be terminated.

General procedure:

1. Allocate Master Slave Connection Set, function code 0x4b
2. Get/Set Attribute, function code 0x0e,0x10
3. Release Master Slave Connection Set, function code 0x4c

7.2.1 Generated error codes

Code (Hex)	Designation according to standard	Description
0x00	DnEC_SUCCESS	Service was successfully performed by the object specified
0x02	DnEC_RESOURCE_UNAVAILABLE	Resource unavailable Resources needed for the object to perform the requested service were unavailable.
0x04	DnEC_PATH_SEGMENT	Path Segment Error The path segment identifier or the segment syntax was not understood by the processing node. Path processing shall stop when a path segment error is encountered.
0x05	DnEC_PATH_DEST_UNKNOWN	Path Destination Unknown The path is referencing an object class, instance or structure element that is not known or is not contained in the processing node. Path processing shall stop when a path destination unknown error is encountered.
0x07	DnEC_CONNECTION_LOST	Connection lost The messaging connection was lost
0x08	DnEC_SERVICE_NOT_SUPPORTED	Service not supported The requested service was not implemented or was not defined for this Object Class/Instance.
0x09	DnEC_INVALID_ATTR_VALUE	Invalid attribute value Invalid attribute data detected
0x0a	DnEC_ATTR_LIST	Attribute list error An attribute in the Get_Attribute_List or Set_Attribute_List response has a non-zero status.
0x0b	DnEC_REQUESTED_MODE	Already in requested node/state The object is already in the mode/state being requested by the service.
0x0c	DnEC_OBJ_STATE	Object state conflict The object cannot perform the requested service in its current mode/state.
0x0d	DnEC_OBJ_DOES_EXIST	Object already exists The requested instance of object to be created already exists.

Code (Hex)	Designation according to standard	Description
0x0e	DnEC_ATTR_NOT_SETTABLE	Attribute not settable A request to modify a non-modifiable attribute was received.
0x0f	DnEC_PRIVILEGE_VIOLATION	Privilege violation A permission/privilege check failed
0x10	DnEC_DEV_STATE	Device state conflict The device's current mode/state prohibits the execution of the requested service.
0x11	DnEC_REPLY_DATA	Reply data too large The data to be transmitted in the response buffer is larger than the allocated response buffer.
0x13	DnEC_NOT_ENOUGH_DATA	Not enough data The service did not supply enough data to perform the specified operation.
0x14	DnEC_ATTR_NOT_SUPPORTED	Attribute not supported The attribute specified in the request is not supported.
0x15	DnEC_TOO_MUCH_DATA	Too much data The service supplied more data than was expected.
0x16	DnEC_OBJ_NOT_EXIST	Object does not exist The object specified does not exist in the device.
0x18	DnEC_STORE_ATTR_DATA	No stored attribute data The attribute data of this object was not saved prior to the requested service.
0x19	DnEC_STORE_OPERATION	Store operation failure The attribute data of this object was not saved due to a failure during the attempt.
0x1c	DnEC_ATTR_LIST_DATA	Missing attribute list entry data The service did not supply an attribute in a list of attributes that was needed by the service to perform the requested behaviour.

Code (Hex)	Designation according to standard	Description
0x1d	DnEC_ATTR_LIST_VALUE	Invalid attribute value list The service is returning the list of attributes supplied with status information for those attributes that were invalid.
0x1f	DnEC_VENDOR	Vendor specific error A vendor specific error has been encountered. The Additional Code Field of the Error Response defines the particular error encountered. Use of this General Error Code should only be performed when none of the Error Codes presented in this table or within an Object Class definition accurately reflect the error.
0x20	DnEC_INVALID_PARAMETER	Invalid parameter A parameter associated with the request was invalid. This code is used when a parameter does not meet the requirements of this specification and/or the requirements defined in an Application Object Specification.
0x27	DnEC_UNEXPECTED_ATTR	Unexpected attribute in list An attempt was made to set an attribute that is not able to be set at this time.
0x28	DnEC_INVALID_MEMBER	Invalid Member ID The Member ID specified in the request does not exist in the specified Class/Instance/Attribute.
0x29	DnEC_MEMBER_NOT_SETTABLE	Member not settable request to modify a non-modifiable member was received.
0x2a	DnEC_GRP2_ONLY_FAIL	Group 2 only server general failure This error code may only be reported by group 2 only servers with 4K or less code space and only in place of Service not supported, Attribute not supported and Attribute not settable.

7.2.2 Data types

Designation according to DeviceNet	Description	Abbreviation in the following tables
USINT	Unsigned byte 8 bits in length	u8
UINT	Unsigned word 16 bits in length	u16
UDINT	Unsigned integer 32 bits in length	u32
SINT	Integer signed in the most significant bit and 8 bits in length	i8
INT	Integer signed in the most significant bit and 16 bits in length	i16
DINT	Integer signed in the most significant bit and 32 bits in length	i32
BOOL	Byte with information in the least significant bit (Bit 0)	b8
STRUCT	Structure	Struct
ARRAY	Field with identical elements	array
PATH	6-byte DeviceNet path, describes an Object 0x20,<Class>,0x24,<Instance>,0x30,<Attribute>	epath
SSTRING	Short String String with explicit length information Length max. 255 bytes	sstring
FLT	Signed floating-point number, 32 bits in length	R32

7.2.3 Identity Object, Class 0x01

Instance	Attribute	Access	Name	Data type	Description	Value
0	1	R	Revision	U16	Revision of the Identity Object	0x0001
1	1	R	VendorID	U16	Identifies the manufacturer	0x389
1	2	R	Device Type	U16	Product type 0x0000 = GenericDevice	0x0000
1	3	R	Product Code	U16	Identifies the product	DF30= 0x0901 DF31= 0x0902
1	4	R	Revision	U8, U8	Revision of the software	0x0800
1	5	R	Status	U16	Status of the module	0-255
1	6	R	Serial Number	U32	P-number of the device	
1	6	R	Product Name	SSTRING	Identification	
1	8	R	Device Status	U8	General Device Status	0-255
1	9	R	Config Value	U16	Configuration value	
1	10	RW	Heartbeat Interval	U8	Heartbeat Interval	0-255

7.2.4 Message Router, Class 0x02

Instance	Attribute	Access	Name	Data type	Description	Value
0	1	R	Revision	U16	Revision of the Message Router Object	1

7.2.5 DeviceNet Object, Class 0x03

Instance	Attribute	Access	Name	Data type	Description	Value
0	1	R	Revision	U16	Revision of the DeviceNet Object	2
1	1	RW	MACID	U8	Node address	0-63
1	2	RW	Bitrate	U8	Bitrate	0-2
1	3	RW	BOI	B8	Busoffl nterrupt	0-1
1	4	RW	Busoff counter	U8	Number of Busoffs	0-255
1	5	R	Allocation Information	Struct of U8,	Masters MacID	0-63
1	6	R	MacId Changed	U8	MacID was changed	0-1
1	7	R	Bitrate Changed	U8	Bitrate was changed	0-1
1	8	R	MacID Switch Value	U8	Value of MAC-ID switch	0-63
1	9	R	Bitrate Switch Value	U8	Value of bitrate switch	0-2

7.2.6 Assembly Object, Class 0x04

Instance	Attribute	Selection HBM	Assembly	Data type	Name	Access
0	1	–		U16	Revision of the Assembly Object	R
100	1	0	Poll Output Assembly	U8	Number of attributes in the Assembly	R
100	2	0	Poll Output Assembly	EPATH	Path Variable	RW
100	3	0	Poll Output Assembly	Array of U8	Data of the Assembly	RW
101	1	1	Poll Output Assembly	U8	Number of attributes in the Assembly	R
101	2	1	Poll Output Assembly	EPATH	Path Variable	RW
101	3	1	Poll Output Assembly	Array of U8	Data of the Assembly	RW

Instance	Attribute	Selection HBM	Assembly	Data type	Name	Access
102	1	2	Poll Output Assembly	U8	Number of attributes in the Assembly	R
102	2	2	Poll Output Assembly	EPATH	Path Variable	RW
102	3	2	Poll Output Assembly	Array of U8	Data of the Assembly	RW
120	1	0	Poll Input Assembly	U8	Number of attributes in the Assembly	R
120	2	0	Poll Input Assembly	EPATH	Path Variable	RW
120	3	0	Poll Input Assembly	Array of U8	Data of the Assembly	R
121	1	1	Poll Input Assembly	U8	Number of attributes in the Assembly	R
121	2	1	Poll Input Assembly	EPATH	Path Variable	RW
121	3	1	Poll Input Assembly	Array of U8	Data of the Assembly	R
122	1	2	Poll Input Assembly	U8	Number of attributes in the Assembly	R
122	2	2	Poll Input Assembly	EPATH	Path Variable	RW
122	3	2	Poll Input Assembly	Array of U8	Data of the Assembly	R
123	1	3	Poll Input Assembly	U8	Number of attributes in the Assembly	R
123	2	3	Poll Input Assembly	EPATH	Path Variable	RW
123	3	3	Poll Input Assembly	Array of U8	Data of the Assembly	R
124	1	4	Poll Input Assembly	U8	Number of attributes in the Assembly	R
124	2	4	Poll Input Assembly	EPATH	Path Variable	RW

Instance	Attribute	Selection HBM	Assembly	Data type	Name	Access
124	3	4	Poll Input Assembly	Array of U8	Data of the Assembly	R
125	1	5	Poll Input Assembly	U8	Number of attributes in the Assembly	R
125	2	5	Poll Input Assembly	EPATH	Path Variable	RW
125	3	5	Poll Input Assembly	Array of U8	Data of the Assembly	R
126	1	6	Poll Input Assembly	U8	Number of attributes in the Assembly	R
126	2	6	Poll Input Assembly	EPATH	Path Variable	RW
126	3	6	Poll Input Assembly	Array of U8	Data of the Assembly	R
127	1	7	Poll Input Assembly	U8	Number of attributes in the Assembly	R
127	2	7	Poll Input Assembly	EPATH	Path Variable	RW
127	3	7	Poll Input Assembly	Array of U8	Data of the Assembly	R
128	1	8	Poll Input Assembly	U8	Number of attributes in the Assembly	R
128	2	8	Poll Input Assembly	EPATH	Path Variable	RW
128	3	8	Poll Input Assembly	Array of U8	Data of the Assembly	R
129	1	9	Poll Input Assembly	U8	Number of attributes in the Assembly	R
129	2	9	Poll Input Assembly	EPATH	Path Variable	RW
129	3	9	Poll Input Assembly	Array of U8	Data of the Assembly	R

Instance	Attribute	Selection HBM	Assembly	Data type	Name	Access
130	1	10	Poll Input Assembly	U8	Number of attributes in the Assembly	R
130	2	10	Poll Input Assembly	EPATH	Path Variable	RW
130	3	10	Poll Input Assembly	Array of U8	Data of the Assembly	R
131	1	11	Poll Input Assembly	U8	Number of attributes in the Assembly	R
131	2	11	Poll Input Assembly	EPATH	Path Variable	RW
131	3	11	Poll Input Assembly	Array of U8	Data of the Assembly	R
140	1	0	Strb Input Assembly	U8	Number of attributes in the Assembly	R
140	2	0	Strb Input Assembly	EPATH	Path Variable	RW
140	3	0	Strb Input Assembly	Array of U8	Data of the Assembly	R
141	1	1	Strb Input Assembly	U8	Number of attributes in the Assembly	R
141	2	1	Strb Input Assembly	EPATH	Path Variable	RW
141	3	1	Strb Input Assembly	Array of U8	Data of the Assembly	R
142	1	2	Strb Input Assembly	U8	Number of attributes in the Assembly	R
142	2	2	Strb Input Assembly	EPATH	Path Variable	RW
142	3	2	Strb Input Assembly	Array of U8	Data of the Assembly	R
143	1	3	Strb Input Assembly	U8	Number of attributes in the Assembly	R
143	2	3	Strb Input Assembly	EPATH	Path Variable	RW

Instance	Attribute	Selection HBM	Assembly	Data type	Name	Access
143	3	3	Strb Input Assembly	Array of U8	Data of the Assembly	R
160	1	0	Cos Input Assembly	U8	Number of attributes in the Assembly	R
160	2	0	Cos Input Assembly	EPATH	Path Variable	RW
160	3	0	Cos Input Assembly	Array of U8	Data of the Assembly	R
161	1	1	Cos Input Assembly	U8	Number of attributes in the Assembly	R
161	2	1	Cos Input Assembly	EPATH	Path Variable	RW
161	3	1	Cos Input Assembly	Array of U8	Data of the Assembly	R
162	1	2	Cos Input Assembly	U8	Number of attributes in the Assembly	R
162	2	2	Cos Input Assembly	EPATH	Path Variable	RW
162	3	2	Cos Input Assembly	Array of U8	Data of the Assembly	R
163	1	3	Cos Input Assembly	U8	Number of attributes in the Assembly	R
163	2	3	Cos Input Assembly	EPATH	Path Variable	RW
163	3	3	Cos Input Assembly	Array of U8	Data of the Assembly	R

7.2.7 Connection Object, Class 0x05

Instance	Attribute	Usage	Access	Name	Data type	Description	Value
0	1	Re-quired	R	Re-vision	U16	Revision of the Connection Object	1

7.2.8 Connection Object, Class 0x05, Instance 1, Explicit connection

Instance	Attribute	Access	Name	Data type	Description
1	1	R	State	U8	Object status
1	2	R	Instance_type	U8	Indicates I/O or Explicit Connection
1	3	R	TransportClassTrigger	U8	Defines connection performance
1	4	R	Produced Connection Id	U8	Can Identifier
1	5	R	Consumed Connection Id	U8	Can Identifier
1	6	R	Inital Communication Characteristic	U8	Connection links
1	7	R	Produced Connection Size	U8	Max. number of transmittable bytes
1	8	R	Consumed Connection Size	U8	Max. number of transmittable bytes
1	9	RW	Expected Packet Rate	U8	Connection timer
1	12	RW	Watchdog TimeOut Action	U8	Behavior with connection failure
1	13	R	Produced ConnectionPathLength	U8	Number of bytes in the Produced ConnectionPath
1	14	RW	Produced ConnectionPath	Array of U8	DeviceNet Path describes the connection elements
1	15	R	Consumed ConnectionPathLength	U8	Number of bytes in the Consumed ConnectionPath
1	16	RW	Consumed ConnectionPath	Array of U8	DeviceNet Path describes the connection elements
1	17	R	Production Inhibit Time	U8	Minimum time between data transmissions

7.2.9 Connection Object, Class 0x05, Instance 2, Polled connection

Instance	Attribute	Access	Name	Data type	Description
1	1	R	State	U8	Object status
1	2	R	Instance_type	U8	Indicates I/O or Explicit Connection
1	3	R	Transport Class Trigger	U8	Defines connection performance
1	4	R	Produced Connection Id	U8	CanIdentifier
1	5	R	Consumed Connection Id	U8	CanIdentifier
1	6	R	Inital Communication Characteristic	U8	Connection links
1	7	R	Produced Connection Size	U8	Max. number of transmittable bytes
1	8	R	Consumed Connection Size	U8	Max. number of transmittable bytes
1	9	RW	Expected Packet Rate	U8	Connection timer
1	12	RW	Watchdog TimeOut Action	U8	Behavior with connection failure
1	13	R	Produced ConnectionPathLength	U8	Number of bytes in the Produced ConnectionPath
1	14	RW	Produced ConnectionPath	Array of U8	DeviceNet Path describes the connection elements
1	15	R	Consumed ConnectionPathLength	U8	Number of bytes in the Consumed ConnectionPath
1	16	RW	Consumed ConnectionPath	Array of U8	DeviceNet Path describes the connection elements
1	17	R	Production Inhibit Time	U8	Minimum time between data transmissions

7.2.10 Connection Object, Class 0x05, Instance 3, BitStrobe connection

Instance	Attribute	Access	Name	Data type	Description
1	1	R	State	U8	Object status
1	2	R	Instance_type	U8	Indicates I/O or Explicit Connection
1	3	R	Transport Class Trigger	U8	Defines connection performance
1	4	R	Produced Connection Id	U8	Can Identifier
1	5	R	Consumed Connection Id	U8	Can Identifier
1	6	R	Inital Communication Characteristic	U8	Connection links
1	7	R	Produced Connection Size	U8	Max. number of transmittable bytes
1	8	R	Consumed Connection Size	U8	Max. number of transmittable bytes
1	9	RW	Expected Packet Rate	U8	Connection timer
1	12	RW	Watchdog TimeOut Action	U8	Behavior with connection failure
1	13	R	Produced ConnectionPathLength	U8	Number of bytes in the Produced ConnectionPath
1	14	RW	Produced ConnectionPath	Array of U8	DeviceNet Path describes the connection elements
1	15	R	Consumed ConnectionPathLength	U8	Number of bytes in the Consumed ConnectionPath
1	16	RW	Consumed ConnectionPath	Array of U8	DeviceNet Path describes the connection elements
1	17	R	Production Inhibit Time	U8	Minimum time between data transmissions

7.2.11 Connection Object, Class 0x05, Instance 4, ChangeOfState / Cyclic connection

Instance	Attribute	Access	Name	Data type	Description
1	1	R	State	U8	Object status
1	2	R	Instance_type	U8	Indicates I/O or Explicit Connection
1	3	R	Transport Class Trigger	U8	Defines connection performance
1	4	R	Produced Connection Id	U8	Can Identifier
1	5	R	Consumed Connection Id	U8	Can Identifier
1	6	R	Inital Communication Characteristic	U8	Connection links
1	7	R	Produced Connection Size	U8	Max. number of transmittable bytes
1	8	R	Consumed Connection Size	U8	Max. number of transmittable bytes
1	9	RW	Expected Packet Rate	U8	Connection timer
1	12	RW	Watchdog TimeOut Action	U8	Behavior with connection failure
1	13	R	Produced ConnectionPathLength	U8	Number of bytes in the Produced ConnectionPath
1	14	RW	Produced ConnectionPath	Array of U8	DeviceNet Path describes the connection elements
1	15	R	Consumed ConnectionPathLength	U8	Number of bytes in the Consumed ConnectionPath
1	16	RW	Consumed ConnectionPath	Array of U8	DeviceNet Path describes the connection elements
1	17	R	Production Inhibit Time	U8	Minimum time between data transmissions

7.2.12 Acknowledge Handler Object, 0x2b

Instance	Attribute	Access	Name	Data type	Description
1	1	RW	Acknowledge Timer	U16	Wait time for Ack
1	2	RW	Retry Limit	U16	Number of repeats for Ack
1	3	R	COS Production Connection Instance	U16	Connection Instance informed by Ack-Handler.

7.3 CAN object dictionary, in function group order

7.3.1 Communication profile

Class	Instance	Sub–Index	Access[1]	Data type[2]	Value	Description	Parameter set[3]
100	1	1	RO	VS	Visible string	Manufacturer device name (20 characters)	–
100	1	2	RO	VS	Visible string	Manufacturer hardware version (13 characters)	–
100	1	3	RO	VS	Visible string	Manufacturer firmware version (8 characters)	–
3	1	1	RO	u8	MAC–ID	Device address	–
1	1	4	RO	u16		Manufacturer firmware	–
3	1	2	RW	u8	0: 125 kbit/s 1: 250 kbit/s 2: 500 kbit/s	Bitrate (also stored in EEPROM), active after next new start	–
1	1	1	RO	u32	HBM: 389 hex	CANopen Vendor–ID	–
1	1	2	RO	u32	0:generic device	Device Type	–
1	1	3	RO	u32	DF30DN: 0901hex	CANopen Product–ID	–
1	1	6	RO	u32		HBM serial number	–
100	1	4	RO	VS	Visible String (12 characters)	HBM serial number	–
100	1	5	RW	VS	Visible String (20 characters) “HBM digiCLIP DF31DN”	Channel name, defined individually by user	A

1) RW: Read and write accessf RO: Read access only
WO: Write access only

2) The format describes the data type, as noted in Section 7.2.

3) Parameter set column: A: Value is stored in the application parameter set; C: Value is stored in the communication parameter set; _ : Value is not stored in a parameter set

7.3.2 Parameter set and factory setting

In the digiCLIP Assistant, click on ”Store parameter in device” or ”Restore factory setting” to read or write the application parameter set. This is marked in the tables by an ”A”. The communication parameter contains the selected bitrate, and the selection of the ”Assemblies” after a new start. The objects concerned are marked in the tables by a ”C”.

Class	Instance	Sub–Index	Access	Data type	Value	Description	Parameter set
100	2	1	RW	u32	Write: 65766173 hex	Store all current application and communication parameters.	–
100	2	2	RW	u32	Write: 65766173 hex	Protect only the current communication parameters.	–
100	2	3	RW	u32	Write: 65766173 hex	Protect only the current application parameters (”A”).	–
100	2	4	RW	u32	Write: 64616F 6Chex	Factory setting: Restore all applications (”A”) and communicationa (”C”) Parameters; Bitrate: 125 kbit/s	–
100	2	6	RW	u32	Write: 64616F 6Chex	Factory setting: Only restore applications parameters (”A”)	–
100	2	5	RW	u32	Write: 64616F 6Chex	Factory setting: Only restore communications (”C”) Parameters; Bitrate: 125 kbit/s	–

7.3.3 Measured values

Class	Instance	Sub–Index	Access	Data type	Value	Description	Parameter set
104	1	1	ROP	i32		Max. peaks measured value	–
104	1	2	ROP	i32		Min. peaks measured value	–
104	1	3	ROP	i32		Peak–to–peak measured value	–
104	2	1	ROP	r32		Max. peaks measured value	–
104	2	2	ROP	r32		Min. peaks measured value	–
104	2	3	ROP	r32		Peak–to–peak measured value	–
104	2	1	ROP	r32		Gross measured value	–
104	2	2	ROP	r32		Net measured value	–
104	1	1	ROP	i32		Gross measured value	–
104	1	2	ROP	i32		Net measured value	–

7.3.4 Device status

Class	Instance	Sub–Index	Access	Data type	Value	Description	Parameter set
101	3	1	ROP	u8	System status 1 Bits 0…2: as footnote [1)] Bit 3: CAN bus error or warning Bits 4…7: Limit value switches 1…4 triggered		–
101	3	2	ROP	u32	System status 2: Bit 0: Measured value invalid Bit 1: Positive measurement input overload Bit 2: Negative measurement input overload Bit 3: Pos. measuring range overshoot Bit 4: Neg. measuring range overshoot Bit 5: Scaling error Bit 6: Incorrect initial calibration values Bit 7: Error when initializing limit value switches Bits 8…11: Limit value switches 1…4 triggered Bit 12: Hardware error: parameter memory (EEPROM) Bit 13: Hardware error: program memory (FLASH) Bit 14: Hardware error: Autocalibration Bit 15: TEDS error[2)] Bits 16…21: Transducer connection faulty: Bit 16: Terminal 2, HBM: black Bit 17: Terminal 2', HBM: gray Bit 18: Terminal 3, HBM: blue Bit 19: Terminal 3', HBM: green Bit 20: Terminal 4 [+], HBM: red Bit 21: Terminal 1 [–], HBM: white Bits 22…31: *reserved*		–
101	3	3	RO	u8	0: digiCLIP is SLAVE 1: digiCLIP is MASTER	Hardware synchronization	–
101	3	4	RO	u8	0: identical 1: not identical	Check whether the current application parameters match the data in the EEPROM	–
107	2	2	ROP	u8		Random count value ("Life counter")	–

1) System status 1: Bit 0 is set when an error occurs which influences the measured value. So when this bit is set, it means that the measured value is invalid. This is the case, for example, with a measurement input overload, a faulty transducer connection, an undefined choice of excitation voltage and for all scaling errors. This bit is cleared when the cause of the error is removed. It is not set when range monitoring is exceeded. Bit 1 is set when the measurement input is overloaded in the positive direction or when the gross measured value exceeds range monitoring in the positive direction. Bit 2 is set accordingly for overshoots in the negative direction.

2) TEDS data availability is only monitored if this has been activated (digiCLIP Assistant: "Always use TEDS" checked)

7.3.5 Device control

Class	Instance	Sub–Index	Access	Data type	Value	Description	Parameter set
101	4	1	RWP	u8	Control byte 1:[1)] Bit 0: Run zeroing Bit 1: Run taring Bit 2: Continuous clear of max. peak–value memory Bit 3: Continuous clear of min. peak–value memory Bit 4: One–off clear of max. peak–value memory Bit 5: One–off clear of min. peak–value memory Bit 6: Stop max. peak–value memory Bit 7: Stop min. peak–value memory Factory setting: all Bits = 0		A[2)]
101	4	2	RW	u8	Bit n = 1: Function enabled Bit n = 0: Function inhibited	Control byte 1 Mask When bit = 1, the corresponding control byte bit (Index 2268) is executed; when bit = 0, the corresponding control byte bit is ignored and assumed to be ”0”.	A
					Factory setting: all Bits = 1		
101	4	3	RWP	u8	Control byte 2[3)] : Bit 0: Run autocalibration (”Auto–Cal”) Bit 1: Run zeroing (”Auto–Zero”) Bit 2: Run taring (”Auto–Tare”) Bit 7: Read out TEDS and trigger TEDS calibration Factory setting: all Bits = 0		–
101	4	4	RW	u8	Bit n = 1: Function enabled Bit n = 0: Function inhibited	Control byte 2 Mask When bit = 1, the corresponding control byte bit 2 is executed; when bit = 0, the corresponding control byte bit is ignored and assumed to be ”0”.	A
101	5	1	WO	u32	696C6163 hex	Trigger one–off autocalibration (”Auto–Cal”)	–
101	5	2	WO	u32	7A65726F hex	Trigger zeroing (”Auto–Zero”)	–
101	5	3	WO	u32	74617261 hex	Trigger taring (”Auto–Tare”)	–

1) If several command bits are set at the same time, this is the sequence that is followed: zeroing, taring, edit peak–value memory. If several bits are set to control the peak–value memory, this is the priority that is applied (the first–named has the highest priority): continuous clear, one–off clear, stop. The functions of bits 0, 1, 4 and 5 are only executed when there is a change of state from logic 0 to 1.

2) Only the state of bits 2, 3, 6 and 7 is protected in the application parameter set.

3) If several command bits are set at the same time, this is the sequence that is followed: zeroing, taring, autocalibration. Bit 7, for calibrating by TEDS, must not be set at the same time as the other control bits of Object 101/4/3.

7.3.6 Peak-value memory control

Class	Instance	Sub–Index	Access	Data type	Parameters	Description	Parameter set
104	3	1	RW	u8	0: Gross meas. value (Factory setting) 1: Net meas. value	Input signal for max. peak-value memory	A
104	3	2	RW	u8	0: Gross meas. Value (Factory setting) 1: Net meas. value	Input signal for min. peak-value memory	A
104	4	1	RW	u8	0: Normal operation (Factory setting) 1: continuous clear	Continuous clear of max. peak-value memory: Peak value follows current measured value	A
104	4	2	RW	u8	0: Normal operation (Factory setting) 1: continuous clear	Continuous clear of min. peak-value memory: Peak value follows current measured value	A
104	4	3	RW	u8	0: Normal operation 1: one-off clear	One–off clear of max. peak-value memory: Next measured value is current max peak value. Read returns = 1 until clearing has been executed in the device	–
104	4	4	RW	u8	0: Normal operation 1: one–off clear	One–off clear of min. peak-value memory: Next measured value is current min. peak value. Read returns = 1 until clearing has been executed in the device	–
104	4	5	RW	u8	0: Normal operation (Factory setting) 1: stop	Stop max. peak–value memory: Peak–value memory remains unchanged, whatever the subsequent measured values	A
104	4	6	RW	u8	0: Normal operation 1: stop	Stop min. peak-value memory: Peak-value memory remains unchanged, whatever the subsequent measured values	A

7.3.7 Digital inputs and outputs (DF31DN only)

Slot C2	Index (hex)	Access	Data type	Value	Description	Parameter set
1	80	RW	u8	Bit 0: input polarity Bit 4: output 1 polarity Bit 5: output 2 polarity	Polarity of digital input and digital outputs: inverting when bit set	A
				Factory settings: all bits = 0		
1	81	RW	u8	Action of digital input: [1] Bit 0: run zero balance Bit 1: run taring Bit 2: continuous clear of max. peak-value memory Bit 3: continuous clear of min. Peak-value memory Bit 4: one−off clear of max. Peak-value memory Bit 5: one−off clear of min. Peak-value memory Bit 6: stop max. Peak-value memory Bit 7: stop min. Peak-value memory Factory settings: all bits = 1		A
1	82	RO	u8	Bit 0: input status Bit 4: output 1 status Bit 5: output 2 status	Electrical status of digital input and digital outputs [2]: bit set when 24 V	–
1	83	RO	u8	Bit 0: input status Bit 4: output 1 status Bit 5: output 2 status	Logic state of digital input and digital outputs, taking polarity into consideration: bit set when action active	–

1) If several bits are set at the same time, this is the sequence that is followed: zero balance, taring, edit peak−value memory. If several bits are set to control the peak-value memory, this is the priority that is applied (first-named has the highest priority): continuous clear, one-off clear, stop. The actions for bit 0, bit 1, bit 4 and bit 5 are carried out precisely when the input voltage changes from the quiescent level to the active level. The actions for bit 2, bit 3, bit 6 and bit 7 are carried out as long as the input voltage corresponds to the active level. The quiescent or active levels are defined with index 2300. The response occurs with the next but one measured value at the latest. The latency time of the electronic digital input can be found in the current data sheet.

2) Short circuit of digital output is not recognized.

Slot C2	Index (hex)	Access	Data type	Value	Description	Parameter set
1	85	RW	u8	Signal source of digital output 1: [3] Bit 0: limit value switch 1 Bit 1: limit value switch 2 Bit 2: limit value switch 3 Bit 3: limit value switch 4 Bit 4: positive out-of-range Bit 5: negative out-of-range Bit 6: overload of input amplifier Bit 7: general error with invalid measured value Factory settings: all bits = 0		A
1	86	RW	u8	Signal source of digital output 2: bit assignment as for digital output 1 Factory settings: all bits = 1		A

3) Several bits can be set simultaneously. The logic states are then assigned ”or-linked” to the digital output. The switching states of bits 0 to 6 are updated with every measured value. The status of bit 7 indicates general errors that lead to invalid measured values, such as transducer, scaling or TEDS errors.
A response time greater than 400 ms must be assumed here. The latency time of the electronic digital input can be found in the current data sheet.

7.3.8 Scaling

There are three scaling methods available: With HBM transducers, the zero value and the span are most often available as scaling data. Two–point scaling, as defined in CANopen, can be used as an alternative. If a transducer with TEDS is connected, the scaling values can also be set with TEDS. The objects for TEDS can be found in Section 7.3.9 . If a scaling value is changed, the scaling values in the other representation are adapted automatically.

Class	Instance	Sub–Index	Access	Data type	Parameters	Description	Parameter set
102	3	1	RW	r32	0.0 (Factory setting)	Span scaling: scaling value: mV/V zero point	A
102	2	1	RW	i32	0.0 (Factory setting)	Span scaling: scaling value: mV/V zero point	A

Class	Instance	Sub–Index	Access	Data type	Parameters	Description	Parameter set
102	3	2	RW	r32	0.0 (Factory setting)	Span scaling: scaling value: phys. zero point	A
102	2	2	RW	i32	0.0 (Factory setting)	Span scaling: scaling value: phys. zero point	A
102	3	3	RW	r32	1.0 (Factory setting)	Span scaling: scaling value: mV/V span	A
102	2	3	RW	i32	1000 (Factory setting)	Span scaling: scaling value: mV/V span	A
102	3	4	RW	r32	1.0 (Factory setting)	Span scaling: scaling value: phys. span.	A
102	2	4	RW	i32	1000 (Factory setting)	Span scaling: scaling value: phys. span.	A
102	4	1	WO	u32	31746573 hex	Two-point scaling: Calibrate X1: set the current internal mV/V meas. value as scaling value point 1	–
102	4	2	WO	u32	32746573 hex	Two-point scaling: Calibrate X2: set the current internal mV/V meas. value as scaling value point 2	–
102	6	1	RW	r32	0.0 (Factory setting)	Two-point scaling: scaling value: mV/V point 1	A
102	5	1	RW	i32	0.0 (Factory setting)	Two-point scaling: scaling value: mV/V point 1	A
102	6	2	RW	r32	0.0 (Factory setting)	Two-point scaling: scaling value: phys. point 1	A
102	5	2	RW	i32		Two-point scaling: scaling value: phys. point 1	A
102	6	3	RW	r32	1.0 (Factory setting)	Two-point scaling: scaling value: mV/V point 2	A
102	5	3	RW	i32	1000 (Factory setting)	Two-point scaling: scaling value: mV/V point 2	A
102	6	4	RW	r32	1.0 (Factory setting)	Two-point scaling: scaling value: phys. point 2	A
102	5	4	RW	i32	1000 (Factory setting)	Two-point scaling: scaling value: phys. point 2	A
102	7	1	RW	u8	0…9 3 (Factory setting)	Decimal point position, the value range can be further restricted, subject to scaling.	A

7.3.9 TEDS

If several transducers with TEDS are connected to an amplifier input, it is only ever the first TEDS to be found that is evaluated. In this case, automatic scaling by TEDS and the ”Always use TEDS” function should be dispensed with.

Class	Instance	Sub−Index	Access	Data type	Parameters	Description	Parameter set
103	1	1	RW	u8		Write: Parameter = 1: Contact the first TEDS and load the data to the device memory[1]. Read: Return value = 1, if the data has been successfully read and is available, otherwise return value = 0	A
103	1	2	RW	u32	0.0 (Factory setting)	Physical reference unit, into which TEDS data are to be converted [2]	A
103	1	3	WO	u32	73646574 hex	Activate scaling by TEDS	–
103	1	4	RO	i16		TEDS: Read out the last calibration date (number of days since January 1, 1998)	–
103	1	5	RO	i16		TEDS: Read out the calibration period	–
103	1	6	RO	VS	Visible string (3 char.)	TEDS: Read out the initials of the calibrator	–
103	1	7	RO	VS	Visible string (45 char.)	TEDS: Read out the transducer comments	–
103	1	8	OS	i16	OctetString (8 bytes)	TEDS: Read out transducer identification (T−ID)	–

1) Whenever a transducer is connected and each time the device is re−started, the TEDS data are read into the device automatically, so that it is not normally necessary to address the TEDS specifically.

2) The physical reference unit is the quantity into which the scaling values are converted, when a TEDS has been read out. This method also allows non−metric units to be supported or a conversion, for example, from newtons (as stored in the TEDS) to kilonewtons (as required in the digiCLIP application). In many cases, the user will set the same unit here, as for displaying the measured values. If a required unit is not compatible with the TEDS data because, for example, a torque transducer has been connected, but newtons, the force transducer unit, have been selected, a CAN error message is returned and scaling does not take place.

Class	Instance	Sub–Index	Access	Data type	Parameters	Description	Parameter set
103	1	9	RW	u8	0: Do not use TEDS automatically (Factory setting) 1: Always use TEDS	Always use TEDS[1)]	A
103	1	10	RO	u8	0: Manual scaling 1: Current scaling corresponds to the TEDS data	Current scaling took place on account of TEDS activation	–
103	1	12	RO	u16		Basic TEDS Template: ”Manufacturer”	–
103	1	13	RO	u16		Basic TEDS Template: ”Model”	–
103	1	14	RO	u8		Basic TEDS Template: ”Version letter”	–
103	1	15	RO	u16		Basic TEDS Template: ”Version number”	–
103	1	16	RO	u32		Basic TEDS Template: ”Serial number”	–

1) ”Always use TEDS” causes the availability of the TEDS data to be monitored, the TEDS activates and scaling takes place in accordance with the TEDS data. Write access to scaling values is then declined.

7.3.10 Transducer settings

Class	Instance	Sub–Index	Access	Data type	Parameters	Description	Parameter set
102	1	1	RW	u8	0: 2.5 V (Factory setting) 1: 1.0 V	Excitation voltage, 2.5 V sets the measuring range to ±4 mV/V, 1.0 V sets the measuring range to ±10 mV/V	A
102	1	2	RO	u8	0:±4 mV/V 1:±10 mV/V	Measuring range	–
107	3	1	RW	u8	0: Normal measurement mode 1: Internal zero signal 2: Internal calibration signal	Selecting the input amplifier signal. Normal measurement mode is always set after a new start.	–

7.3.11 Signal conditioning

Class	Instance	Sub–Index	Access	Data type	Parameters	Description	Parameter set
102	7	4	RW	r32	100.0 (Factory setting)	Write: Choice of filter frequency in Hz. [1] Reading the index returns the actually active filter frequency in Hz.	A
102	7	5	RW	u8	120: 100 Hz, (Factory setting) 119: 50 Hz, 118: 20 Hz, 117: 10 Hz, 116: 5 Hz, 115: 2 Hz, 114: 1 Hz, 113: 0.5 Hz, 112: 0.2 Hz, 111: 0.1 Hz, 110: 0.05 Hz	Filter frequency, Bessel-like	A
102	9	1	RW	r32	0.0 (Factory setting)	Zero point	A
102	8	1	RW	i32	0.0 (Factory setting)	Zero point	A
102	9	2	RW	r32	0.0 (Factory setting)	Tare value	A
102	8	2	RW	i32	0.0 (Factory setting)	Tare value	A
102	7	2	RW	VS	Visible string	Physical unit as a string, exactly 12 characters in length. [2]	A
102	7	3	RW	u32	CiA constant (Factory setting)	Physical unit as CiA constant per DR303–2	A
102	7	1	RW	u8	0…9 3 (Factory setting)	Decimal point position, the value range can be further restricted, subject to scaling.	A

[1] If the required frequency is not available in the device, the next highest possible one is set as the frequency. When a frequency higher than the highest possible one is chosen, the error state is indicated and the previous filter coefficients are not changed. Writing this object resets Object 102/7/5.

[2] These values are only stored in the device, they are not evaluated. If Object 102/7/2 is changed directly by SDO, this does not affect the entry in Object 102/7/3. Conversely, Object 102/7/2 is changed when Object 102/7/3 is written, when there is a text to this effect stored in the device. Scaling by TEDS also causes the entries of these objects to change.

7.3.12 Other device functions

Class	Instance	Sub–Index	Access	Data type	Parameters	Description	Parameter set
107	1	1	RW	u8	”User–Tag” has no effect on the system (Factory setting)	Can be used as a storage cell or for dummy accesses by the user	A
107	1	2	RW	u16	”User–Tag” has no effect on the system (Factory setting)	Can be used as a storage cell or for dummy accesses by the user	A
107	1	3	RW	u32	”User–Tag” has no effect on the system (Factory setting)	Can be used as a storage cell or for dummy accesses by the user	A
107	2	1	RW	u32	Write: 746F6F62 hex Read: 0: Normal operation, 1: System in restart	Write: Run a system restart; Read: System state	–
107	3	2	RW	u32	CiA date format (number of days since January 1, 1984)	Date of the last calibration; write with password protection	–

7.3.13 Range monitoring

Range monitoring does not lead to an error message when the limit value is exceeded. Instead, corresponding status bits are set to ”measuring range monitoring”.

Class	Instance	Sub–Index	Access	Data type	Parameters	Description	Parameter set
106	12	1	RW	r32	$-1 \cdot 10^{10}$ (Factory setting)	Gross measured value range monitoring: Lower limit	A
106	12	2	RW	r32	$+1 \cdot 10^{10}$ (Factory setting)	Gross measured value range monitoring: Upper limit	A
106	11	1	RW	i32	–2147483648	Gross measured value range monitoring: Lower limit	A
106	11	2	RW	i32	+2147483647	Gross measured value range monitoring: Upper limit	A

7.3.14 Limit value monitoring

Class	Instance	Sub–Index	Access	Data type	Parameters	Description	Parameter set
106	1	1	RW	u32	Compare with: *Gross meas. value:* 61300120 hex or 91300120 hex (Factory setting) *Net measured value:* 61400120 hex or 91400120 hex *Max. peak meas. value:* 20020120 hex or 30020120 hex *Min. peak meas. value:* 20030120 hex or 30030120 hex *Peak-to-peak meas. value:* 20040120 hex or 30040120 hex	Measured value source for limit value switch 1	A
106	1	2	RW	u8	inactive: 0 (Factory setting) greater or equal: 2 less: 3	Level reference for limit value switch 1	A
106	1	7	RW	r32	0.0 (Factory setting)	Threshold value for limit value switch 1, physical quantity	A
106	1	5	RW	i32	0.0 (Factory setting)	Threshold value for limit value switch 1, physical quantity	A

Class	Instance	Sub–Index	Access	Data type	Parameters	Description	Parameter set
106	1	8	RW	r32	Value >= 0 (Factory setting)	Hysteresis for limit value switch 1, physical quantity	A
106	1	6	RW	i32	0.0 (Factory setting)	Hysteresis for limit value switch 1, physical quantity	A
106	1	3	RO	b8	0: not triggered 1: triggered	State of limit value switch 1	–
106	1	4	WO	b8	0: no action 1: clear	Clear hysteresis state of limit value switch 1	–
106	2	1	RW	u32	see Index 6503	Measured value source for limit value switch 2	A
106	2	2	RW	u8	see Index 6508	Level reference for limit value switch 2	A
106	2	7	RW	r32	0.0 (Factory setting)	Threshold value for limit value switch 2	A
106	2	5	RW	i32	0.0 (Factory setting)	Threshold value for limit value switch 2	A
106	2	8	RW	r32	Value >= 0	Hysteresis for limit value switch 2	A
106	2	6	RW	i32	0.0 (Factory setting)	Hysteresis for limit value switch 2	A
106	2	3	RO	b8	0: not triggered 1: triggered	State of limit value switch 2	–
106	2	4	WO	b8	0: no action 1: clear	Clear hysteresis state of limit value switch 2	–

Class	Instance	Sub–Index	Access	Data type	Parameters	Description	Parameter set
106	3	1	RW	u32	see Index 6503	Measured value source for limit value switch 3	A
106	3	2	RW	u8	see Index 6508	Level reference for limit value switch 3	A
106	3	7	RW	r32	0.0 (Factory setting)	Threshold value for limit value switch 3	A
106	3	5	RW	i32	0.0 (Factory setting)	Threshold value for limit value switch 3	A
106	3	8	RW	r32	Value >= 0 0.0 (Factory setting)	Hysteresis for limit value switch 3	A
106	3	6	RW	i32	0.0 (Factory setting)	Hysteresis for limit value switch 3	A
106	3	3	RO	b8	0: not triggered 1: triggered	State of limit value switch 3	–
106	3	4	WO	b8	0: no action 1: clear	Clear hysteresis state of limit value switch 3	–
106	4	1	RW	u32	see Index 6503	Measured value source for limit value switch 4	A
106	4	2	RW	u8	see Index 6508	Level reference for limit value switch 4	A
106	4	7	RW	r32	0.0 (Factory setting)	Threshold value for limit value switch 4	A
106	4	5	RW	i32	0.0 (Factory setting)	Threshold value for limit value switch 4	A
106	4	8	RW	r32	Value >= 0 0.0 (Factory setting)	Hysteresis for limit value switch 4	A

Class	Instance	Sub–Index	Access	Data type	Parameters	Description	Parameter set
106	4	6	RW	i32		Hysteresis for limit value switch 4	A
106	4	3	RO	b8	0: not triggered 1: triggered	State of limit value switch 4	–
106	4	4	WO	b8	0: no action 1: clear	Clear hysteresis state of limit value switch 4	–
106	10	1	ROP	u8	Bit 0 = switch 1 ...Bit 3 = switch 4	State of limit value switches 1…4	A
106	10	2	ROP	b8	0: no switch triggered 1: min. one switch triggered	Overall state of all limit value switches	A
106	10	3	WOP	b8	0: no action 1: clear all switches	Clear hysteresis states of all limit value switches	A

7.4 Format of cyclic data

7.4.1 Format Poll Output Data (PLC ⇒ module)

Selection by setting Class 199, Instance 1, Attribute 1 (HBM) or writing the Assembly Path in Class 5, Instance 2, Attribute 16

HBM selection	Assembly Path	Number of bytes	Element 1 Offset Class Instance Attribute	Element 2 Offset Class Instance Attribute
0	4,100,2	1	+0 101,4,1 Control byte1 u8	–
1	4,101,2	1	+0 101,4,3 Control byte2 u8	–
2	4,102,2	2	+0 101,4,1 Control byte1 u8	+1 101,4,3 Control byte2 u8

7.4.2 Format Poll Input Data (module ⇒ PLC)

Selection by setting Class 199, Instance 1, Attribute 2 (HBM) or writing the Assembly Path in Class 5, Instance 2, Attribute 14

HBM selection	Assembly Path	Number of bytes	Element 1 Offset Class Instance Attribute	Element 2 Offset Class Instance Attribute	Element 3 Offset Class Instance Attribute	Element 4 Offset Class Instance Attribute	Element 5 Offset Class Instance Attribute
0	4,120,2	6	+0 101,1,1 Meas. val. Gross i32	+4 101,3,1 Status1 U8	+5 105,3,2 DigInput U8		
1	4,121,2	6	+0 101,1,2 Meas. val. Gross R32	+4 101,3,1 Status1 U8	+5 105,3,2 DigInput U8		
2	4,122,2	6	+0 101,1,2 Meas. val. Net i32	+4 101,3,1 Status1 U8	+5 105,3,2 DigInput U8		
3	4,123,2	6	+0 101,1,2 Meas. val. Net R32	+4 101,3,1 Status1 U8	+5 105,3,2 DigInput U8		
4	4,124,2	14	+0 101,1,1 Meas. val. Gross i32	+4 104,1,2 Peak Min I32	+8 104,1,1 Peak Max I32	+12 101,3,1 Status1 U8	+13 105,3,2 DigInput U8
5	4,125,2	14	+0 101,1,2 Meas. val. Gross R32	+4 104,2,2 Peak Min R32	+8 104,2,1 Peak Max R32	+12 101,3,1 Status1 U8	+13 105,3,2 DigInput U8
6	4,126,2	10	+0 101,1,1 Meas. val. Gross i32	+4 104,1,3 Peak Peak I32	+8 101,3,1 Status1 U8	+9 105,3,2 DigInput U8	

HBM selec-tion	Assembly Path	Number of bytes	Element 1 Offset Class Instance Attribute	Element 2 Offset Class Instance Attribute	Element 3 Offset Class Instance Attribute	Element 4 Offset Class Instance Attribute	Element 5 Offset Class Instance Attribute
7	4,127,2	10	+0 101,1,2 Meas. val. Gross R32	+4 104,2,3 Peak Peak R32	+8 101,3,1 Status1 U8	+9 105,3,2 DigInput U8	
8	4,128,2	14	+0 101,1,2 Meas. val. Net i32	+4 104,1,2 Peak Min I32	+8 104,1,1 Peak Max I32	+12 101,3,1 Status1 U8	+13 105,3,2 DigInput U8
9	4,129,2	14	+0 101,1,2 Meas. val. Net R32	+4 104,2,2 Peak Min R32	+8 104,2,1 Peak Max R32	+12 101,3,1 Status1 U8	+13 105,3,2 DigInput U8
10	4,130,2	10	+0 101,1,2 Meas. val. Net i32	+4 104,1,3 Peak Peak I32	+8 101,3,1 Status1 U8	+9 105,3,2 DigInput U8	
11	4.131,2	10	+0 101,1,2 Meas. val. Net R32	+4 104,2,3 Peak Peak R32	+8 101,3,1 Status1 U8	+9 105,3,2 DigInput U8	

7.4.3 Format BitStrobe Input Data (module ⇒ PLC)

Selection by setting Class 199, Instance 1, Attribute 3 (HBM) or writing the Assembly Path in Class 5, Instance 3, Attribute 14

HBM selection	Assembly Path	Number of bytes	Element 1 Offset Class Instance Attribute	Element 2 Offset Class Instance Attribute	Element 3 Offset Class Instance Attribute
0	4,140,2	6	+0 101,1,1 Meas. val. Gross I32	+4 101,3,1 Status1 U8	+5 105,3,2 DigInput U8
1	4,141,2	6	+0 101,1,2 Meas. val. Gross R32	+4 101,3,1 Status1 U8	+5 105,3,2 DigInput U8
2	4,142,2	6	+0 101,1,2 Meas. val. Net I32	+4 101,3,1 Status1 U8	+5 105,3,2 DigInput U8
3	4,143,2	6	+0 101,1,2 Meas. val. Net R32	+4 101,3,1 Status1 U8	+5 105,3,2 DigInput U8

7.4.4 Format ChangeOfState / Cyclic Input Data (module ⇒ PLC)

Selection by setting Class 199, Instance 1, Attribute 4 (HBM) or writing the Assembly Path in Class 5, Instance 4, Attribute 14

HBM selection	Assembly Path	Number of bytes	Element 1 Offset Class Instance Attribute	Element 2 Offset Class Instance Attribute	Element 3 Offset Class Instance Attribute
0	4,160,2	6	+0 101,1,1 Meas. val. Gross I32	+4 101,3,1 Status1 U8	+5 105,3,2 DigInput U8
1	4,161,2	6	+0 101,1,2 Meas. val. Gross R32	+4 101,3,1 Status1 U8	+5 105,3,2 DigInput U8
2	4,162,2	6	+0 101,1,2 Meas. val. Net I32	+4 101,3,1 Status1 U8	+5 105,3,2 DigInput U8
3	4,163,2	6	+0 101,1,2 Meas. val. Net R32	+4 101,3,1 Status1 U8	+5 105,3,2 DigInput U8

7.5 Communication without a DeviceNet Master

This section helps you understand the DeviceNet examples in section 7.6.

7.5.1 Introduction

This section describes how DeviceNET devices can be operated without a DeviceNET Master.

DeviceNET defines a number of services:

- **Change-of-State/Cyclic Message**

  Used to convey alarms. The service is parameterized with the “Connection Object”. This service is not dealt with here.

- **Bit-Strobe Message**

  This service is used to request master data from one slave or from a group of slaves. The service is parameterized with the “Connection Object”. This service is not dealt with here.

- **Poll Messages**

  Poll Messages are used to transmit cyclic data. The service is parameterized with the “Connection Object”. Chapter 7.5.3 deals with this service.

- **Explicit Messages**

  Used to convey acyclic data, such as configuration parameters. Chapter 7.5.4 deals with this service.

7.5.1 .1 Declarations

The following declarations are made to simplify communication:

1. MasterMACID=0, Slave MACIDs=1..63
2. Attribute format 8/8
3. Only those of “Group2 only, predefined Connection Set” are used.

7.5.2 Network management functions

7.5.2 .1 Duplicate MACID check.

After being reset, the Slave twice transmits the so-called “Duplicate MACID check” message. This message is received by all the other Slaves.

If a Slave detects its own MACID, it transmits a “Duplicate MACID check” message with a set R/R bit. This allows the Master to recognize MACIDs assigned twice.

Data direction Slave –> Master

Connection ID	Byte Offset	Content	Explanation
Dup MAC ID Check	0	0x00/0x80	Message Header with R/R field
	1	0x89	Vendor ID Low Byte
	2	0x03	Vendor ID High Byte 0x0389=HBM
	3	xx	Serial Number Low Byte
	4	xx	
	5	xx	
	6	xx	Serial Number High Byte

xx: according to the application

The Connection ID is calculated to:

Connection ID = 0x407 + (SLAVEMACID*8)

7.5.2 .2 “Open Explicit Messaging Connection Request”

After the reset, only a so–called “Unconnected Port” is open at the Slave. The Master now has to establish a connection:

Data direction Master –> Slave

Connection ID	Byte Offset	Content	Explanation
Slaves Unconnected Port ID	0	0x00/0x40	Message Header with Toggle Flag
	1	0x4b	Service Code
	2	0x03	Class ID
	3	0x01	Instance ID
	4	0x03	Allocation Choice (bit–by–bit) 0x01 – Explicit 0x02 – Polled 0x04 – Bitstrobe 0x08 – COS/Cyclic
	5	0x00	Allocators MACID

The Connection ID is calculated to:

Connection ID = 0x406 + (SLAVEMACID*8)

The Slave responds with “Success Response to Allocate_Master/Slave_Connection_Set Request”.

Connection ID	Byte Offset	Content	Explanation
Slaves Explicit Response ID	0	0x00/0x40	Message Header with Toggle Flag
	1	0xcb	Response Code
	2	0x00	Message Body Format

The Connection ID is calculated to:

Connection ID = 0x403 + (SLAVEMACID*8)

7.5.2 .3 “Release_Master/Slave_Connection Set Request Message”

This message closes a connection.

Data direction Master –> Slave

Connection ID	Byte Offset	Content	Explanation
Slaves Unconnected Port ID	0	0x00/0x40	Message Header with Toggle Flag
	1	0x4c	Service Code
	2	0x03	Class ID
	3	0x01	Instance ID
	4	0x03	Release Choice (bit–by–bit) 0x01 – Explicit 0x02 – Polled 0x04 – Bitstrobe 0x08 – COS/Cyclic

The Connection ID is calculated to:

Connection ID = 0x406 + (SLAVEMACID*8)

The Slave responds with “Success Response to Release_Master/Slave_Connection_Set Request”.

Data direction Slave –> Master

Connection ID	Byte Offset	Content	Explanation
Slaves Explicit Response ID	0	0x00/0x40	Message Header with Toggle Flag
	1	0xcc	Response Code

The Connection ID is calculated to:

Connection ID = 0x403 + (SLAVEMACID*8)

Note

The “Expected Packet Rate" attribute in the Connection Object (Class: 5, Instance:1, Attribute: 9), should be set to 0. This deactivates timeout monitoring of the connection. Section 4.2 deals with the attribute settings.

7.5.3 Poll Messages

7.5.3 .1 Master’s I/O Poll request

This is the cyclic data transmitted from the Master to the Slave.

The Slave responds with the Slave’s I/O Poll Response.

Data direction Master –> Slave

Connection ID	Byte Offset	Content	Explanation
Masters I/O Poll Request Connection ID	0	x	Cyclic data Low Byte

xx: according to the application

In this situation, the Master transmits 1 byte to the Slave. The telegrams described here should be understood as examples; the number and data content can be parameterized by the user. Quantity and mapping can be taken from the “Connection Object” of the poll connection.

The Connection ID is calculated to:

Connection ID = 0x405 + (SLAVEMACID*8)

7.5.3 .2 Slave's I/O Poll Response

This is the cyclic data transmitted from the Slave to the Master.

Data direction Slave –> Master

Connection ID	Byte Offset	Content	Explanation
Slaves I/O Poll Response Connection ID	0	0x00	Fragmentation protocol
	1	xx	Byte 1 data
	2	xx	Byte 2 data
	3	xx	Byte 3 data
	4	xx	Byte 4 data
	5	xx	Byte 5 data

xx: according to the application

The Connection ID is calculated to:

Connection ID = 0x3c0 + SLAVEMACID

In this situation, the Slave transmits 6 bytes to the Master. If there are more than 8 bytes to be transmitted, a so-called fragmentation protocol is applied. The fragmentation protocol is not explained in this document. The telegrams described here should be understood as examples; the number and data content can be parameterized by the user. Quantity and mapping can be taken from the "Connection Object" of the poll connection.

7.5.4 Explicit Connections

7.5.4 .1 Reading attributes

7.5.4 .1.1 Masters Explicit Get Attribute Request

Data direction Master –> Slave

Connection ID	Byte Offset	Content	Explanation
Masters Explicit Request Connection ID	0	0x00/0x40	Message Header with Toggle Flag
	1	0x0e	Request Code – Read Attribute Single
	2	Class	Class
	3	Instance	Instance
	4	Attribute	Attribute

This telegram is used to request data at the Slave. The Slave responds either with “Error Response Explicit Message” or “Slave Explicit Response Message”.

The Connection ID is calculated to:

Connection ID := 0x404 + (SLAVEMACID*8)

7.5.4 .1.2 Error Response Explicit Message

Data direction Slave –> Master

Response in the event of an error

Connection ID	Byte Offset	Content	Explanation
Slaves Explicit Response Connection ID	0	0x00/0x40	Message Header with Toggle Flag
	1	0x94	Service Code – Error Response
	2	xx[1)]	General Error Code
	3	xx[1)]	Additional Information

1) For the error codes, see the DeviceNET Specification, Appendix H.

The Connection ID is calculated to:

Connection ID = 0x403 + (SLAVEMACID*8)

7.5.4 .1.3 Slave Explicit Response Message

Data direction Slave –> Master unfragmented

Connection ID	Byte Offset	Content	Explanation
Slaves Explicit Response Connection ID	0	0x00/0x40	Message Header with Toggle Flag
	1	0x8e	Service Code
	2	xx	Byte 1 Response (Low Byte)
	3	xx	Byte 2 Response
	4	xx	Byte 3 Response
	5	xx	Byte 4 Response
	6	xx	Byte 5 Response
	7	xx	Byte 6 Response

xx: according to the application

The number of attribute data items is provided by the DLC Code of the CAN Identifier:

Number of bytes = DLC – 2

The Connection ID is calculated to:

Connection ID = 0x403 + (SLAVEMACID*8)

7.5.4 .1.4 Slave Explicit Response Message fragmented

When there are > 6 bytes of attribute data, the data must be fragmented. In the situation, the “Slaves Explicit Response Message” looks like this:

Data direction Slave –> Master>6 bytes of attribute data

Connection ID	Byte Offset	Content	Explanation
Slaves Explicit Response Connection ID	0	xx	Header Byte
	1	xx	Fragmentation Protocol
	2	0x8e	Service Code
	3	xx	Byte 1 Length code for strings
	4	xx	Byte 2
	5	xx	Byte 3
	6	xx	Byte 4
	7	xx	Byte 5

xx: according to the application

The Connection ID is calculated to:

Connection ID = 0x403 + (SLAVEMACID*8)

The Master acknowledges each fragment with the “Acknowledge Fragmentation Message”

Connection ID	Byte Offset	Content	Explanation
Masters Explicit Request Connection ID	0	xx	Header Byte
	1	xx	Fragmentation Protocol
	2	0/1	ACK Status 0=ok, 1=error

The Connection ID is calculated to:

Connection ID := 0x404 + (SLAVEMACID*8)

Header byte

Bit number								Explanation
7	6	5	4	3	2	1	0	
F								Fragmentation bit = 1
	T							Toggle Flag
		0	0	0	0	0	0	

Fragmentation protocol

Bit number								Explanation
7	6	5	4	3	2	1	0	
F	F							Fragment Type 00 = 1St Fragment 01 = Middle Fragment 10 = Last Fragment 11 = Frag Ack
		N	N	N	N	N	N	Fragment Counter

Note

The “Expected Packet Rate" attribute in the Connection Object (Class: 5, Instance:2, Attribute: 9), should be set to 0. Section 7.5 deals with the attribute settings.This deactivates timeout monitoring of the connection.

7.5.4 .2 Writing Objects

7.5.4 .2.1 Masters Explicit Set Attribute Request unfragmented

Data direction Master –> Slave (1–2 bytes of attribute data)

Connection ID	Byte Offset	Content	Explanation
Masters Explicit Request Connection ID	0	0x00/0x40	Message Header with Toggle Flag
	1	0x10	Request Code – Set Attribute Single
	2	xx	Class
	3	xx	Instance
	4	xx	Attribute
	5	xx	Low Byte data
	6	xx	Byte 2 data
	7	xx	Byte 3 data

xx: according to the application

This telegram is used to write data at the Slave. The Slave responds either with “Error Response Explicit Message” or “Slave Explicit Set Attribute Response Message”.

The Connection ID is calculated to:

Connection ID := 0x404 + (SLAVEMACID*8)

The DLC Code of the CAN Identifier follows from the number of data items to be transmitted:

DLC = number of data items + 4

Slaves Explicit Set Attribute Response

Data direction Slave –> Master (1–2 bytes of attribute data)

Connection ID	Byte Offset	Content	Explanation
Slaves Explicit Response Connection ID	0	0x00/0x40	Message Header with Toggle Flag
	1	0x90	Reponse Code
	2	xx	Low Byte data
	3	xx	Low Byte 2 data

The Connection ID is calculated to:

Connection ID = 0x403 + (SLAVEMACID*8)

7.5.4 .2.2 Masters Explicit Set Attribute Request fragmented

Data direction Master –> Slave (>=4 bytes of attribute data)

Connection ID	Byte Offset	Content	Explanation
Masters Explicit Request Connection ID	0	0x80	Frag Header
	1	0x00	Message Header
	2	0x10	Request code – Set Attribute single
	3	xx	Class
	4	xx	Instance
	5	xx	Attribute
	6	xx	Low Byte data
	7	xx	Byte 2 data

xx: according to the application

This telegram is used to write data at the Slave. The Slave responds either with “Error Response Explicit Message” or “Acknowledge Fragmentation Message”.

The Connection ID is calculated to:

Connection ID := 0x404 + (SLAVEMACID*8)

Slaves Explicit Set Attribute Response

Data direction Slave –> Master (4 bytes of attribute data)

Connection ID	Byte Offset	Content	Explanation
Slaves Explicit Response Connection ID	0	0x80	Frag Header
	1	xx	Fragmentation protocol
	2	xx	ACK Status 0=ok, 1=error

The Connection ID is calculated to:

Connection ID = 0x403 + (SLAVEMACID*8)

The Master now transmits additional records.

Connection ID	Byte Offset	Content	Explanation
Masters Explicit Request Connection ID	0	0x80	Frag Header
	1	xx	Fragmentation protocol
	2	xx	Data byte n
	3	xx	Data byte n+1
	4	xx	Data byte n+2
	5	xx	Data byte n+3
	6	xx	Data byte n+4
	7	xx	Data byte n+5

xx: according to the application

Number of bytes = DLC−2

The Slave acknowledges:

Data direction Slave −> Master (4 bytes of attribute data)

Connection ID	Byte Offset	Content	Explanation
Slaves Explicit Response Connection ID	0	0x80	Frag Header
	1	xx	Fragmentation protocol
	2	xx	ACK Status 0=ok, 1=error

xx: according to the application

7.6 DeviceNet example

Important

For an explanation of these examples, read section 7.5 “Communication without a DeviceNET Master". This section describes the calculation of the identifiers and the meaning of the individual bytes of the CAN telegram.

Cyclic data traffic (Poll connection)

Assumption: Digiclip at address 5
Toggle: Change toggle byte 0x00/0x040
X: Byte is not transferred

1st step PLC –>Digiclip (Allocate Master Slave Connection Set)

Open polled and explicit Connection

Identi-fier	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
0x42e	Toggle	0x4b	0x03	0x01	0x03	0x00	X	X
		Service code	Class	Instan ce	Allocation Choice (Expl and Poll)	Master MACI D		

Digiclip –> PLC (Acknowledgement, all OK)

Identi-fier	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
0x42b	Toggle	0xcb	00	X	X	X	X	X
		Response code	Messa ge– body format					

2nd step PLC –> Digiclip (Class 5, Instance 1, Attribute 9) expected packet rate = 0

Deactivate Timeout of the explicit connection (option). With Timeout monitoring, the connection is closed after 2500mSec, if not used.

Identifier	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
0x42c	Toggle	0x10	0x05	0x01	0x09	0x00	0x00	X
		Service code	Class	Inst	Attribute	Timeout time		

Digiclip –> PLC (Echo of current Attribute value)

Identifier	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
0x42b	Toggle	0x90	00	00	X	X	X	X
		Response code	Timeout time					

3rd step PLC –> Digiclip (Class 5, Instance 2, Attribute 9) expected packet rate = 0

Deactivate Timeout of the explicit connection (option). With Timeout monitoring, the connection is closed after 2500mSec, if not used.

Identifier	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
0x42c	Toggle	0x10	0x05	0x02	0x09	0x00	0x00	X
		Service code	Class	Inst	Attribute	Timeout time		

Digiclip –> PLC (Echo of current Attribute value

Identifier	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
0x42b	Toggle	0x90	00	00	X	X	X	X
		Response code	Timeout time					

4th step PLC –> Digiclip (PLC output data). Here 1 byte, content according to Digiclip parameterization. In this case:

Class 101, Instance 4 Attribute 3, Control byte (0x00).

Identifier	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
0x42d	0x00	X	X	X	X	X	X	X
	Control byte							

5th step Digiclip−>PLC. (PLC input data). Here 6 bytes. The content is subject to the Digiclip parameterization. In this case, it includes the telegram:

Class 101, Instance 1, Attribute 1, Gross measured value_I32 (0xffffc41a)

Class 101, Instance 3 Attribute 1, Status (0x05).

Class 105, Instance 3 Attribute 2, Status of digital inputs (0x00).

Identifier	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
0x3c5	0x1a	0xc4	0xff	0xff	0x05	0x00	X	X
	MeasValue_I32_Gross				Status	Status Dig. inputs		

6th step PLC−>Digiclip (Close the Explicit and Poll connection) or continue from Step 4.

Identifier	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
0x42e	Toggle	0x4c	0x03	0x01	0x03	X	X	X
		Service code	Class	Inst	Release choice			

Digiclip −> PLC (Acknowledgement)

Identifier	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
0x42b	Toggle	0xcc	X	X	X	X	X	X
		Response code						

Acyclic data traffic (Explicit connection)

This section uses examples to deal with reading and writing attributes

1st step PLC −>Digiclip (Allocate Master Slave Connection Set)

Open explicit Connection

Identi-fier	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
0x42e	Toggle	0x4b	0x03	0x01	0x01	0x00	X	X
		Service Code	Class	Inst	Allocation Choice (Explicit)	Master MACID		

Digiclip –> PLC (Acknowledgement, all OK)

Identi-fier	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
0x42b	Toggle	0xcb	00	X	X	X	X	X
		Response code	Message – body format					

2nd step PLC –> Digiclip (Class 5, Instance 1, Attribute 9) expected packet rate = 0

Deactivate Timeout of the explicit connection (option) With Timeout monitoring, the connection is closed after 2500mSec, if not used.

Identifier	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
0x42c	Toggle	0x10	0x05	0x01	0x09	0x00	0x00	X
		Service code	Class	Inst	Attribute	Timeout time		

Digiclip –> PLC (Echo of current Attribute value

Identifier	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
0x42b	Toggle	0x90	00	00	X	X	X	X
		Response code	Timeout time					

3rd step PLC –> Digiclip (Read Class 101, Instance 1, Attribute 1) Measured value132_Gross

Identifier	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
0x42c	Toggle	0x0e	0x65	0x01	0x01	X	X	X
		Service code	Class	Inst	Attribute			

Digiclip –> PLC (Measured value)

Identifier	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
0x42b	Toggle	0x8e	0x29	0x5e	0x24	0x00	X	X
		Response code	Measured value_I32_Gross					

Class 101, Instance 1, Attribute 1, Measured value I32_Gross (0x00245e29)

4th step PLC –> Digiclip (Write Class 102, Instance 7, Attribute 5)

Sets filter frequency to 100Hz

Identifier	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
0x42c	Toggle	0x10	0x66	0x07	0x05	0x78	X	X
		Service code	Class	Inst	Attribute	Parameter 120=100Hz		

Digiclip –> PLC (Acknowledgement)

Identifier	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
0x42b	Toggle	0x90	X	X	X	X	X	X
		Response code						

5th step PLC –> Digiclip (Close Explicit connection)

Identifier	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
0x42e	Toggle	0x4c	0x03	0x01	0x01	X	X	X
		Service code	Class	Inst	Release choice			

Digiclip –> PLC (Acknowledgement)

Identifier	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
0x42b	Toggle	0xcc	X	X	X	X	X	X
		Response code						

8 DigiCLIP data memory in the sensor

This functionality is not available until the following firmware versions:

DF30DN: as of version 1.10

DF31DN: as of version 1.10

14 data memories are available. Each data memory has a maximum data length of 32 bits of user data. The data format is ”unsigned”.

The read or write request only returns once the request has been processed in full. Should an error occur here, because, for example, the object has been incorrectly addressed, there is a transmission error that could not be automatically corrected, or the data memory is damaged, the request returns with an error message. Reading from the sensor usually takes less than 500 ms, writing to the sensor about 1 second. Processing can last up to 3 seconds if there is no sensor with TEDS connected, or in rare cases, when there are transmission faults.

Objects are available to set a data memory to a defined value, as well as objects to automatically increment the current value in the data memory with an incremental value communicated as a parameter.

Before it can be set, the data memory must first be unlocked. Then a constant value can be written once to a data memory. To set a data memory once again, locking first has to be deactivated again. This prevents the wrongful setting of a data memory that is used as an incremental counter.

The relevant object can be directly used to increment the counter in the data memory. There is no active locking for this. If the type of increment that is written would exceed the 32-bit number range, FFFFFFFF (hex) is written to the data memory and no error is generated.

AVERTISSEMENT

As access to the data memories in the sensor is via the measuring lead, there can be no measurement while the data memories in the sensor are being accessed. In this case, the measured values are not updated. The data memory in the sensor is then accessed when the data memory is to be written, or when the data memory is to be read out after the module is switched on, after a sensor replacement or wire break. If the reading of a data memory is repeated, the numerical value stored temporarily in the digiCLIP module is transmitted. Consequently, measurement is not disturbed by repeated readout.

– Suitable measures are applied to increase data security in the digiCLIP module. The aim here is to make sure that module power failure, or removal of the sensor while writing a data memory does not destroy it. However, the reliability of this process cannot be assured to the extent that it is suitable for applications relevant to safety.

– Users must consider that a maximum of 50,000 write accesses can be expected if you add up all the write accesses to the data memories. There is no established limit to the number of read accesses.

– If a tool that has not been approved by HBM is used to write the TEDS data in the sensor, it is possible that data memories will be overwritten. We therefore strongly advise that you only use modules and software from HBM

8.1 Objects for DeviceNet

UINT32: unsigned integer 32 bit; RW: read and write; WO: write only

Class	Instance	Attribute	Access	Data type	Description
103	2	1	RW	u32	Data memory 1 Write: The parameter value is added as a positive increment to the existing numerical value and stored in the data memory. Should incrementation cause the 32-bit number range to be exceeded, the value **FFFFFFFF** (hex) is written and no error is generated. Read: The parameter delivers the current numerical value in the data memory.
103	2	2	RW	u32	Data memory 2
103	2	…	RW	u32	…
103	2	14	RW	u32	Data memory 14

Class	Instance	Attribute	Access	Data type	Description
103	3	1	RW	u32	Data memory 1 Write: The parameter value is written to the data memory. Object **103** / **4** / **1** must first have been transmitted to deactivate write locking. After writing, locking is automatically reactivated. Read: The parameter delivers the current numerical value in the data memory.
103	3	2	RW	u32	Data memory 2
103	3	…	RW	u32	…
103	3	14	RW	u32	Data memory 14

Class	Instance	Attribute	Access	Data type	Description
103	4	1	WO	u32	Data memory 1 Deactivating locking to set a constant value in a data memory (see **103** / **3** / **1** …**14**). Write: Locking is only deactivated when the parameter value is **6B636C75** (hex). All the other parameter values generate an error message.

9 Examples

The following example uses a measurement task to illustrate device functionality and the requisite settings.

Problem:

The forming process in a press is to be monitored, in order to achieve uniform product quality. The maximum force exerted by the press in each cycle is to be recorded. To safeguard the production process, this maximum force must lie between the lower (F1) and the upper (F2) force limit values.

Solution:

The force response measured by a strain gage force transducer (such as the C9B/10 kN; 1 mV/V) is amplified and assessed by the digiCLIP. The maximum force is recorded with the aid of the (maximum) peak−value memory and assessed with two limit value switches with regard to the upper and lower limits.

The state of limit value switches 1 to 4 is regularly read with Object Class 10, Instance 3, Attribute 1.

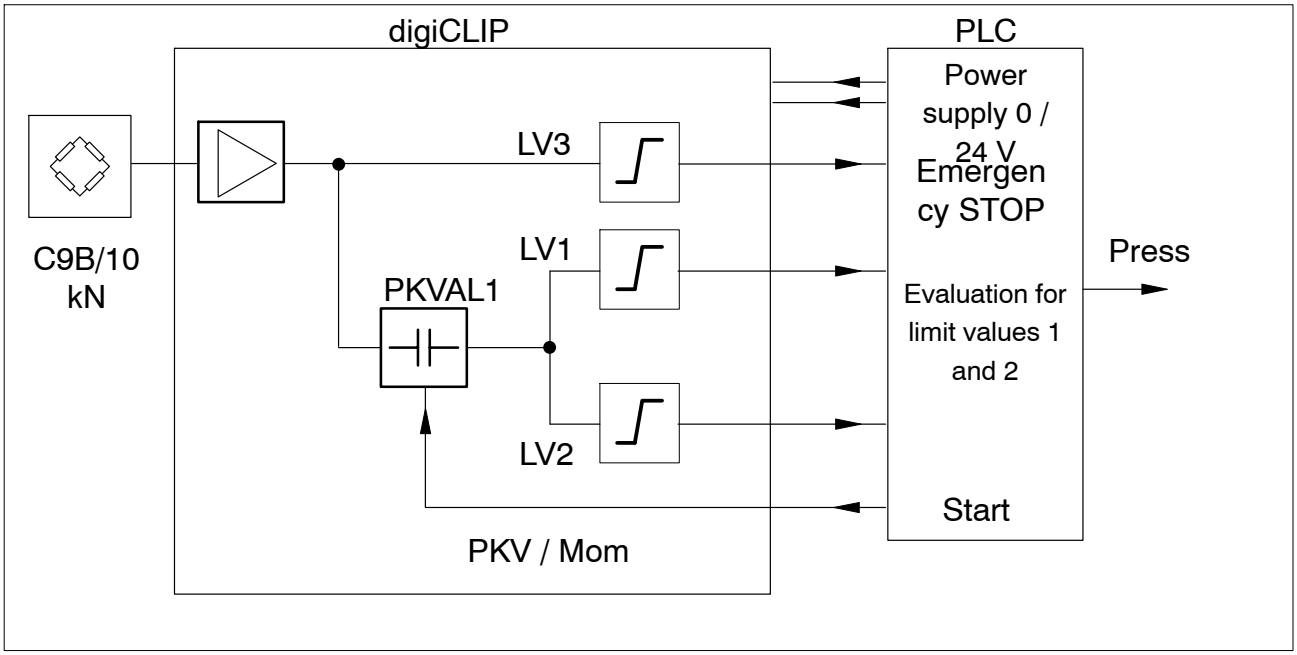
This Object can be mapped in the Poll file via the Assembly ”HBM selection 0 or DeviceNet Class 4, Instance 120, Attribute 2”.

LV1 source = net measured value

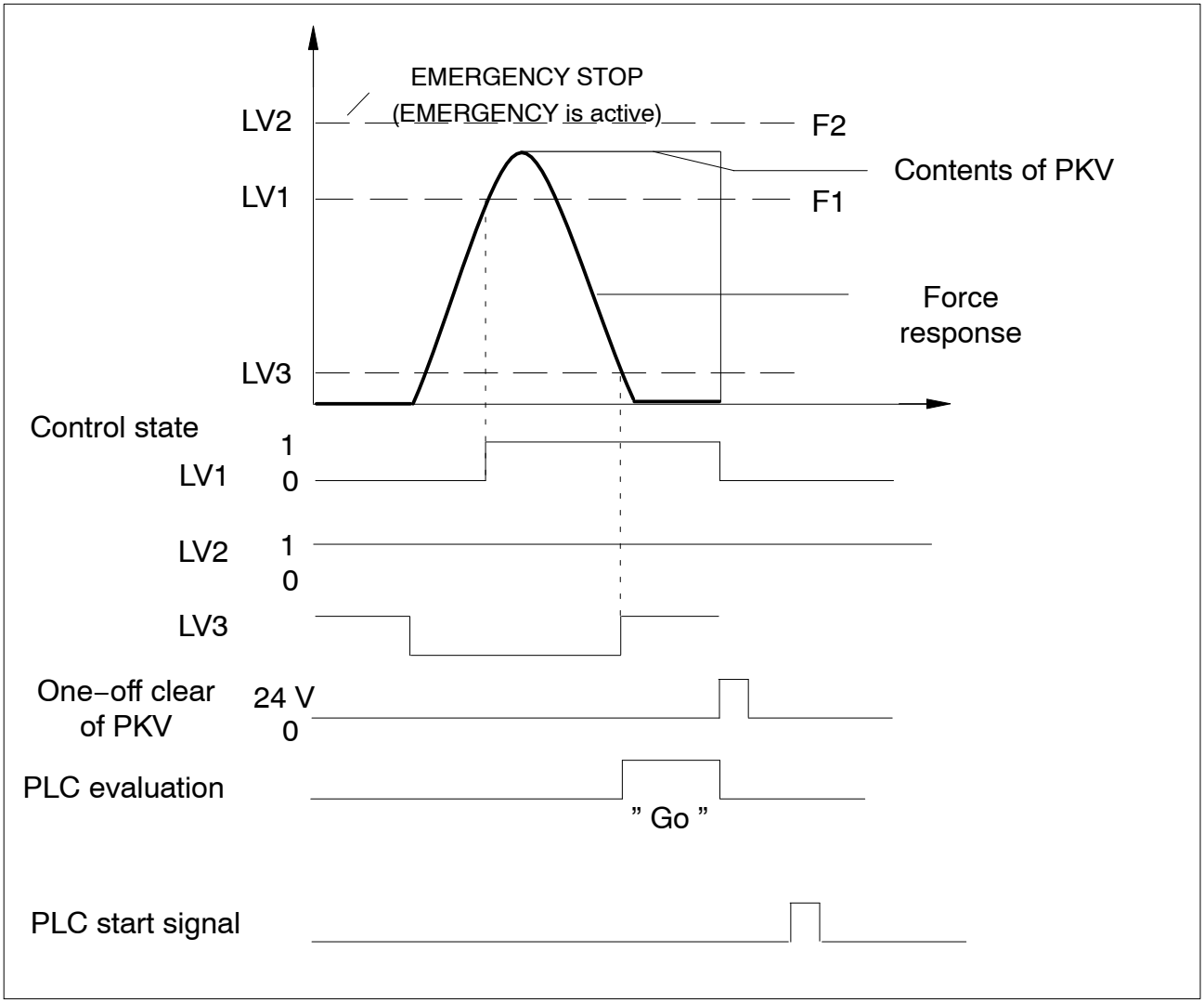
LV2 = gross measured value (machine protection)

A PLC takes over process control. As well as the control commands for the press, it gives the digiCLIP a start signal when the press cycle begins and once the process has run, enlists the help of the limit value outputs for ”Go/No−Go testing”.
The PLC start signal clears the contents of the peak−value memory via a digiCLIP control input.

Wiring diagram:



Timing diagram:



The following settings must be chosen:

LV1 Checks whether the lower force limit (F1) has been reached. The input signal is the output of the peak−value memory (maximum value). If limit LV1 is exceeded, a High signal is generated. A positive actuating direction with positive output logic must be set for this.

LV2 Checks whether the maximum load limit for the machine is exceeded (Cutoff function). The input signal is the gross measured value. If limit LV2 is exceeded, a PDO signal is generated. This is read immediately by the PLC and ensures that the press is quickly shut down.

LV3 Checks whether the press has returned to its starting position. Only then can the PLC start its ” Go / No−Go testing”.

PKV Records the maximum peak value of the force response. The input signal is the net measured value. The PKV is cleared by sending the relevant SDO.

PLC evaluation of the limit value report:

	Go	**Reject**	
LV1	1	0	1
LV2	1	1	0

10 Technical support

Should you have any questions when working with the PMX measuring amplifier system, HBM's technical support can provide:

E−mail support

support@hbm.com

Extended support can be obtained through a maintenance contract.

Fax support

06151 803−288 (within Germany)

+49 6151 803−288 (international)

The following options are also available:

HBM on the Internet

http://www.hbm.de

Download software updates from HBM

http://www.hbm.com/Software

Headquarters world-wide

Europe

Hottinger Baldwin Messtechnik GmbH:
Im Tiefen See 45, 64293 Darmstadt, Deutschland
Tel. +49 6151 8030, Fax +49 6151 8039100
E−Mail: info@hbm.com

www@hbm.com

North and South America

HBM, Inc., 19 Bartlett Street, Marlborough, MA 01752, USA
Tel. +1−800−578−4260 / +1−508−624−4500,
Fax +1−508−485−7480
E−Mail: info@usa.com

Asia

Hottinger Baldwin Measurement (Suzhou) Co., Ltd.
106 Heng Shan Road, Suzhou 215009, Jiangsu, VR China
Tel. (+86) 512 68247776, Fax (+86) 512 68259343
E−Mail: hbmchina@hbm.com.cn

Up to date addresses of representatives can also be found on the Internet under: : www.hbm.com/Contact/Worlwide−Sales−offices

Hottinger Baldwin Messtechnik GmbH
Im Tiefen See 45 • 64293 Darmstadt • Germany
Tel. +49 6151 803–0 • Fax: +49 6151 803–9100
Email: info@hbm.com • www.hbm.com

7–Nr. 2002.3159
A3159–2.0 en

measure and predict with confidence