Automation Solutions

User Manual

LOCC-Box-Net – LOCC-Pads

Description of LOCC-Box-Net, Software LOCC-Pads, Components (Gateway) and Accessories.

Version 1.31

LUTZE Systematic Technology

The user manual is part of the product and contains important information about the handling and the safety. To avoid hazardous situations read the manual before installing the product and using it.
Lütze reserves the right to change its products in the interest of technical progress.
These alterations need not to be documented in every case.

This manual and the contained information have been arranged with the utmost care. The Friedrich Lütze GmbH disclaims liability for literal mistakes and other errors or resulted damages.
The named brands and product names in this document are trademarks or registered trademarks by title holder.

© Copyright 2011 by Friedrich Lütze GmbH All rights reserved.

Friedrich Lütze GmbH
Post office box 1224
D-71366 Weinstadt - Großheppach
Germany

Phone: +49/ (0)7151/ 6053-0
Fax: +49/ (0)7151/ 6053-277
E-Mail: automation@luetze.de
Internet: http://www.luetze.com

Content

1 General Information

1.1 Symbol Description

The manual contains several safety messages. Each safety message contains a defined signal word and a color. The color and the word are referring to an alert level. There are 4 levels. The safety messages point out hazardous situations and give information to avoid those.

⚠ DANGER	Indicates a hazardous situation which, if not avoided will result in death or serious injury.
⚠ WARNING	Indicates a hazardous situation which, if not avoided could result in death or serious injury.
⚠ CAUTION	Indicates a hazardous situation which, if not avoided could result in minor or moderate injury.
NOTICE	Is used to address practices not related to personal injury.

1.2 Copyright

This manual is intended for the operator and his staff. It is forbidden to give the content to a third party, to duplicate, exploit or impart it. The Friedrich Lütze GmbH has to allow it explicit in writing.
General data, text, images and drawings are copyrighted and are liable to the industrial property right. Contravention can be prosecuted criminally. The named brands and product names in this document are trademarks or registered trademarks by titleholder.

1.3 Disclaim of Liability

We have verified the contents of this manual regarding to the conformity of the described hardware and software. Nevertheless divergence may be possible and we disclaim warranty for the complete agreement. The information in this manual will be verified periodically and corrections will be in the next issue.
We would appreciate any kind of suggestion and contributions on your part.
All warranty and liability claims shall be excluded by Friedrich Lütze GmbH in case of damages caused by missing or insufficient knowledge of the operating instructions. Therefore the user company is recommended to have a confirmation in writing about the instruction of the employees.
Modifications or functional alternations on the modules are not allowed due to safety reasons. Any modification on the modules not explicitly authorized by the manufacturer will result in loss of any liability claims to Friedrich Lütze GmbH. The same applies if non authorized parts or equipment are used.

1.4 Safety

1.4.1 Content of Manual

Read and follow the manual before using the product the first time.
This applies to every person which is getting in touch with the product. Trained employees and experts especially qualified persons which had worked with similar products before have to read and understand the manual.

1.4.2 Intended Use

The usage as agreed upon includes the operation in accordance with the operating instructions. The LOCC-Box System is allowed to be used according to the described applications within the technical documents only and in combination with the recommended authorized foreign devices and components only.

1.4.3 Operating Employee

Only highly trained employees are allowed to do the following work on the modules:

- Installation
- Commissioning
- Operating
- Maintenance.

Regarding the safety-related notes qualified employees are people who are allowed to operate with the modules, systems and the current circuits and to ground and mark those according to the safety standards. The operating employees have to be instructed and trained.

1.4.4 Maintenance

The modules are maintenance free. Therefore for continuous operation no inspection or maintenance intervals are necessary.

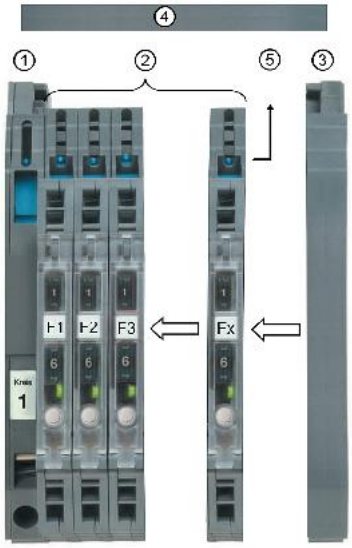
1.4.5 Decommissioning and Deposal

In case of decommissioning and disposal of the modules the user has to observe the valid environmental guidelines of the respective country for user's location.

2 LOCC-Box-Net (Control Unit), 716410

2.1 Installation

2.1.1 System Assembly



For assembly mount the products on a top hat rail according to EN 50022. There are two possibilities:

A) Operating with supply set 716425 (Supply terminal and End block)

❶ Snap on the supply terminal on the left side. The terminal includes the end stop function. The supply terminal is connected to the PSU (DC 12/24V, 40A).

❷ Alongside of the supply terminal (right side) the needed number of control devices 716410 can be mounted.

❸ Mount the end stop terminal.

❹ To get an electrical connection between supply terminal and control device cut the copper rail 716426 to a suitable length**.** Calculation: **18mm + 8,1mm x n** The copper rail has to be snapped into the relevant slots of the supply terminal and end stop terminal.

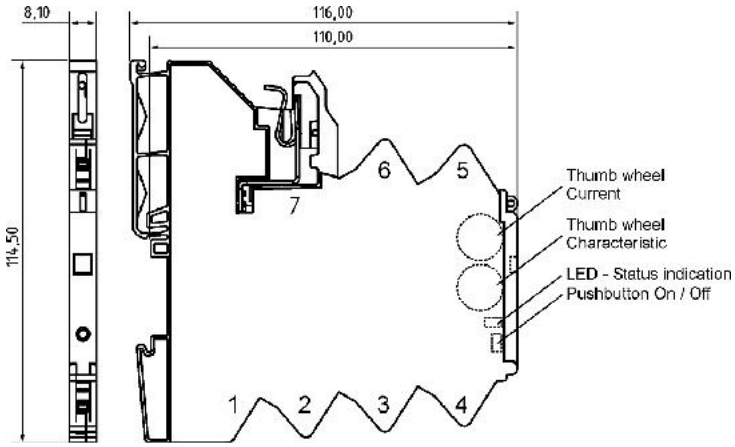
❺ To contact the single control modules, move the contact slide on the copper rail by using a screw driver. The load is connected to termination point (1). The minus potential of the load must be connected directly to the users power supply. This also applies to the minus potential (5). For all other functions, see the function plan on the next page.

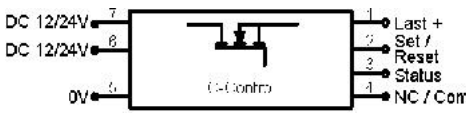
❻ If required, the copper rail can be covered by a plastic cover (716427) between supply terminal and end terminal. The pre cut part is based on the calculation: **6mm + 8,1mm x n**.

B) Direct voltage supply of the control unit

Instead of powering the system via the supply terminal, it is possible to power the system via the termination point (6). A power supply of DC 12/24 V is needed. All other functionalities correspond to description A).

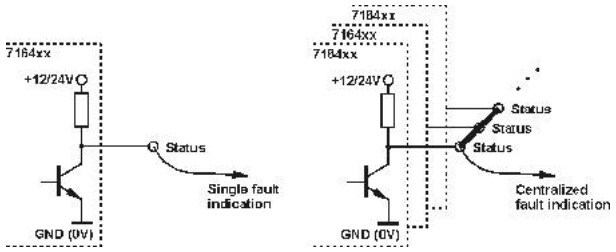
2.1.2 Dimension and Pin assignment





1: + Output
2: Control input (Set / Reset)
3: Status output
4: Communication
5: 0V
6: + Supply (alternative)
7: + Supply

2.1.3 Function, Displays and Settings

Functionality	PIN	Description
Output + , Load	1	+ potential load
Control input - Set /Reset	2	Binary DC 12/24V input to switch on/ off the control unit. Parametrizable with software LOCC-Pads - chapter 3.5.8 Default: Off: DC 12/24V (falling edge, pulse > 100ms, < 800ms) On: DC 12/24V (falling edge, pulse > 1sec)
Single or centralized fault indication	3	Binary output DC 12/24V for fault indication. Parametrizable with software LOCC-Pads - chapter 3.5.8 Status 1: Level High: LOCC-Box switched on ; nominal operation (Default) Level Low: LOCC-Box switched on ; nominal operation Level Low : LOCC-Box blown ; load circuit open Status 2: Level High: LOCC-Box switched on ; nominal operation Level High: LOCC-Box switched on ; nominal operation Level Low : LOCC-Box blown ; load circuit open 
Communication	4	1 wire bus for Gateway
Input -	5	0V LOCC-Box device. Connect it directly to the power supply. On this pin a minus potential of the load is not allowed.
Input + DC 12/24V (alternative)	6	DC 12/24V, max. 10A; LOCC-Box Under consideration of a max. total current DC 6A the jumper comb 716428-716430 (8-pole) or 716438-716440 (16-pole) can be used to distribute the potential.
Contact slide, + input	7	Voltage supply of the module/ Load DC 12/24V, maximum DC 10A

Status indication	Operation	Description
LED green	ON	Function ok – output is active. #
LED red	OFF	The module is switched off or acknowledged via the button or software.#
LED red with short off	OFF	The module is switched off or acknowledged via the remote input.. #
LED green, flashing1Hz	Overload	Load over 90% of I_{nom}
LED green, flashing 5 Hz	Overload	Load over 100% of I_{nom}
LED red, flashing 1Hz	Over current	Output switched off. Reason: over load. #
LED red, with short on	Over current	Output switched off . Reason: short circuit. #
LED red, flashing 5Hz	Error	Wiring fault, internal device error

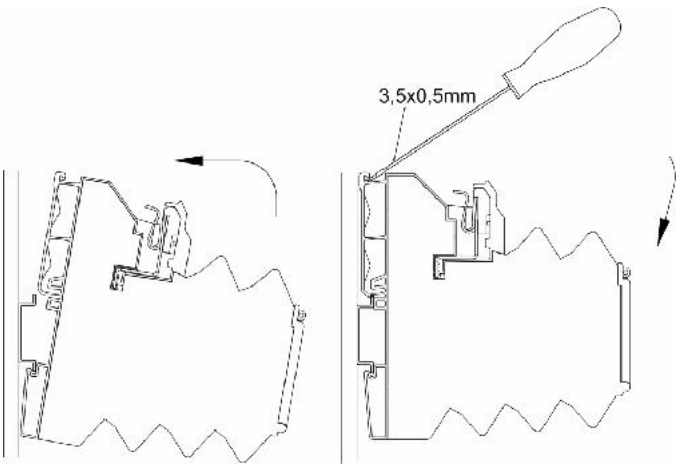
Adjustments	Operation	Description
Thumb wheel - **I** *	Current	Setting the rated current in 1 amp steps.
Thumb wheel - **C** *	Characteristic	1: quick acting, 2: middle acting, 3: slow acting_1, 4: slow acting_2, 5: slow acting_3 (*chapter 2.3*)
Push button **	On / Off	Nominal operation : On / Off Fuse tripped : 1. press: Acknowledge / 2. press: On

* Settings are active after a restart of the modules. Use the push button for the restart.
** The push button has a master function. A module which was switched off by the button can only be restarted by the push button, LOCC-Pads or the field bus.
\# If an interruption of the operating voltage occurs the last status is stored (default).

NOTICE

The status output, the remote input and the switch on behavior after interrupt of the power supply is adjusted of “default”.
Adjustments can be made via LOCC-Pads, *see chapter 3.5.8.*

2.1.4 Installation



	3,5x0,5 mm
10	0,25 ... 2,5 mm²
10	0,25 ... 2,5 mm²
10	0,25 ... 1,5 mm²
AWG	23 to 14

2.2 Technical data

2.2.1 LOCC-Box-Net (Control unit), 716410

Input	
Nominal voltage	DC 12/24V
Voltage range	DC 10 – 32V
Nominal current	DC 10A max.
System current	DC 40A via copper bar 10x3mm
Polarization protection	internal electronic
Terminal	screw less contact slide
Remote input (SET/RESET)	
Signal level	DC 12/24V (EN 61131)
Switch off time (Reset)	> 100ms, < 800ms (falling edge) **#**
Switch on time (Set)	≥ 1sec (falling edge) **#**
Output	
Switching technology	MosFET
Output current	DC 10A max.
Voltage drop	< 170mV (10A)
Status indication	green, red (*see chapter 2.1.3)*
Switch on capacity	10.000 µF
Rated switch off current	DC 1A – DC 10 A (via switch, in steps of 1A)
Switch off times	1= fast acting, 2= middle acting, 3=slow acting_1, 4=slow acting_2, 5 - 9= slow acting_3, Curve 10 is via LOCC-Pads programmable!
General Data	
Housing material	PA 6.6 (UL 94 V0; NFF I2,F2)
Mounting	snap on TS 35 (acc. EN 50022)
Protection level	IP 20
Mounting position	any
Installation technology	spring termination 0.25mm^2 – 2.5mm^2 all types of wires until 2.5mm² without end sleeve, until 1.5mm² with end sleeve
Operation temperature	-25°C … +50°C
Store temperature	-40°C … +85°C
Dimension (WxHxD)	8.1mm x 116mm x 114,5mm
Weight	0.12 kg
Approvals	cULus 508, files E135145, vol. 2, sec. 8
Standards	EN 60950-1; EN61131-1,2; EN 60898; EN 60947-4-1; EN 50081

2.2.2 Supply set (Supply terminal and end block), 716425

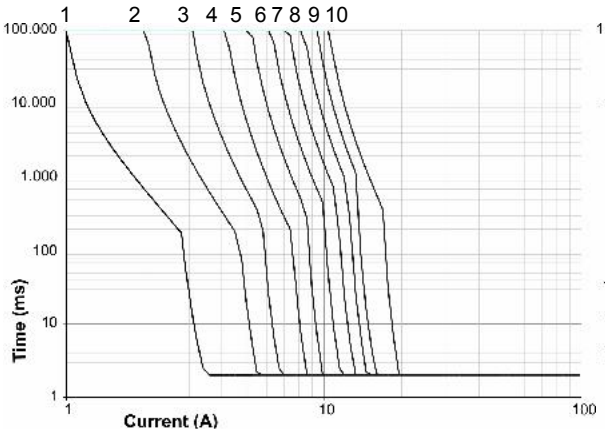
Input	
Nominal voltage	DC 12/24V
Nominal current	DC 40A max.
General Data	
Housing material	PA 6.6 (UL 94 V0; NFF I2,F2)
Mounting	snap on TS 35 (acc. EN 50022)
Protection level	IP 20
Mounting position	any
Installation technology	spring termination 0.33mm^2 – 10.0mm^2 0.33mm² – 6.0mm²
Operation temperature	-25°C … +60°C
Store temperature	-40°C … +85°C
Dimension (WxHxD)	10.0 mm x 63.7mm x 119.4mm
Weight	0.035 kg
Approvals	cULus 508, files E135145, vol.2, sec. 8

Default-settings, and other operations modes can be set via LOCC-Pads

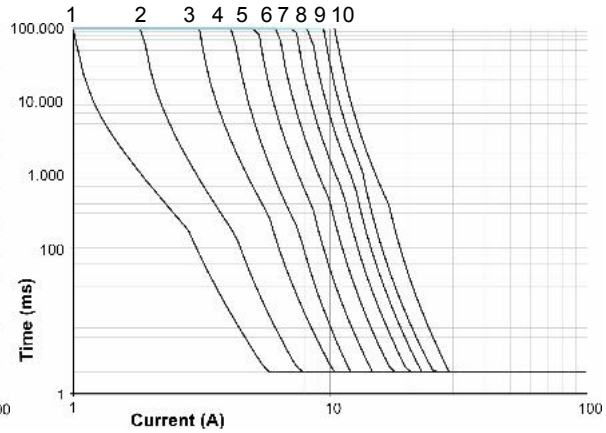
2.3 Characteristic Curves

The LOCC-Box / LOCC-Box-Net offer the possibility to implement 10 characteristics. Up to the product state E06 the first 3 switch positions (C) are realized as: 1 = fast acting, 2 = middle acting and 3 = slow acting_1. With the product state E07 (end of August, 2009) and higher we provide additional the switch position 4 = slow acting_2 and 5 = slow acting_3. All other switching positions are like the last one.

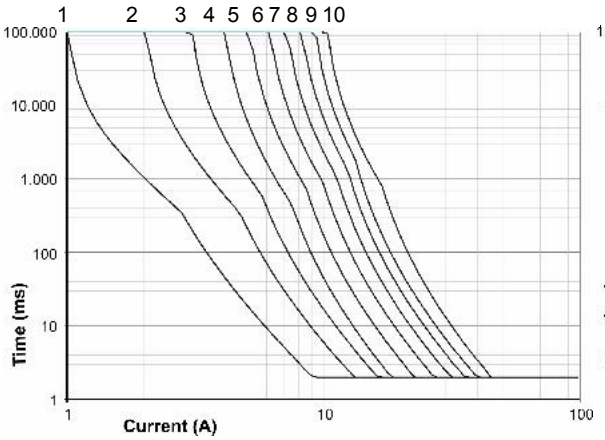
1. Switch setting, characteristic fast acting



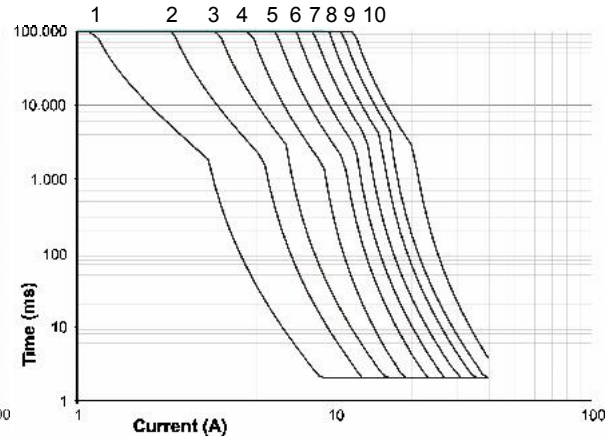
2. Switch setting, characteristic middle acting



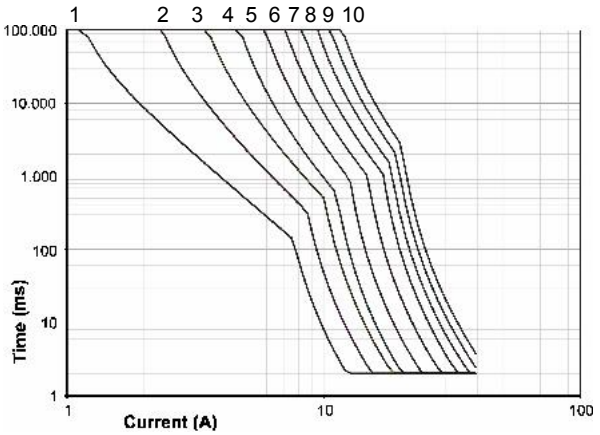
3. Switch setting, characteristic slow acting_1



4. Switch setting, characteristic slow acting_2



5. Switch setting, characteristic slow acting_3



3 LOCC-Pads (Software) [2)]

3.1 Introduction

This section describes the installation and the functionality of the LOCC-Pads software. It also describes the connection between LOCC-Pads (716410) and the Gateway (716459). The software and all components are suitable to the LOCC-Box-Net modules. The software is compatible with following operating systems:

- Windows 2000 [1)]
- Windows XP, 32 Bit [1)]
- Windows Vista, 32 Bit [1)]
- Windows 7, 32 Bit[1)]

The software is designed to parameterize, to analyze and to diagnose the LOCC-Box-Net and all connecting loads. Connect the module via the USB interface to the computer, *see chapter* **Fehler! Verweisquelle konnte nicht gefunden werden.**. For the communication between LOCC-Box-Net and LOCC-Pads a Gateway is required.

NOTICE

Connect the Gateway after installing the software LOCC-Pads, otherwise an installation of the device driver is not possible. The LOCC-Pads software is also needed for the communication via RS232 and CANopen in order to set the required parameters such as parity or baud rate. You can find detailed information in *chapter* **Fehler! Verweisquelle konnte nicht gefunden werden.** *and* **Fehler! Verweisquelle konnte nicht gefunden werden.**.

3.2 Installation

The Software package LOCC-Pads is needed for installation. The software can be downloaded free of charge on the Lütze website:
www.luetze.de > Deutschland > Downloads > Eplan/Macros/Software

To install the software proceed as follows:

1. Download the file ***LOCC-Pads_x.x.x.x.zip*** and save it in any directory. The directory can be deleted after the installation.

2. Extract the Zip-Files in the same directory.

3. Start the installation by double clicking on the file ***LOCC-Pads_setup_xxxx.exe***. Choose a setup language (German / English). Confirm by clicking ***OK.***

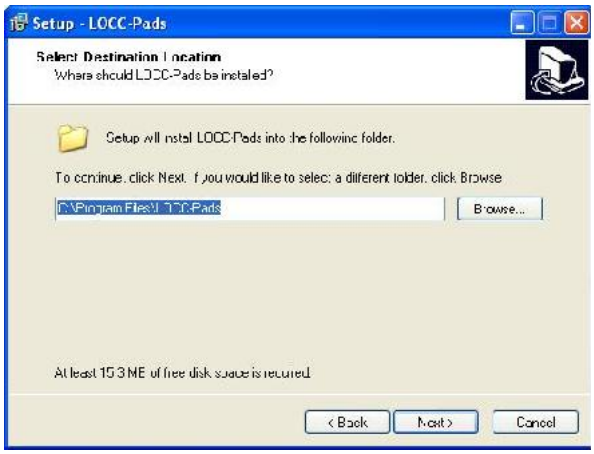
4. Windows[1)] starts the setup assistant. The Assistant guides through the installation. Follow the instructions and confirm with ***Next*** to continue or ***Cancel*** to abort.

5. Licensing Agreement

   Please read the licensing agreement carefully and choose ***I accept the Agreement*** or ***I reject the agreement***. In case of a rejection it is not possible to proceed with the installation. Confirm with ***Next***.

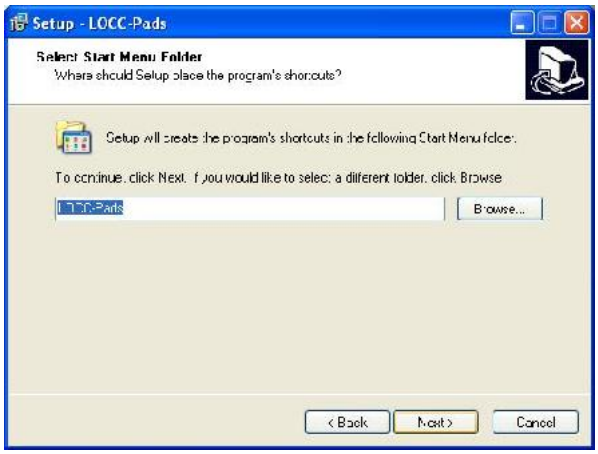
6. Choose a Target Folder

   The setup chooses automatically the presettled folder where LOCC-Pads have to be installed. Click ***Next*** to continue or ***Browse***, if you wish to choose another folder. Note the necessary disc space.



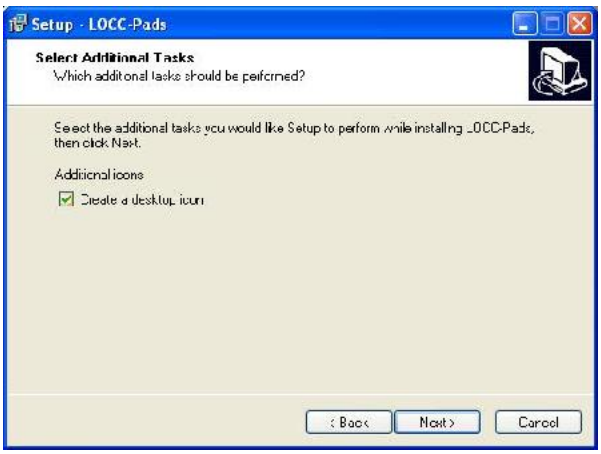
7. Choose the Start Menu Folder

   The setup will create automatically the program linkage in the presettled start menu folder. Click ***Next*** to continue or ***Browse…***, if you wish to choose another folder.



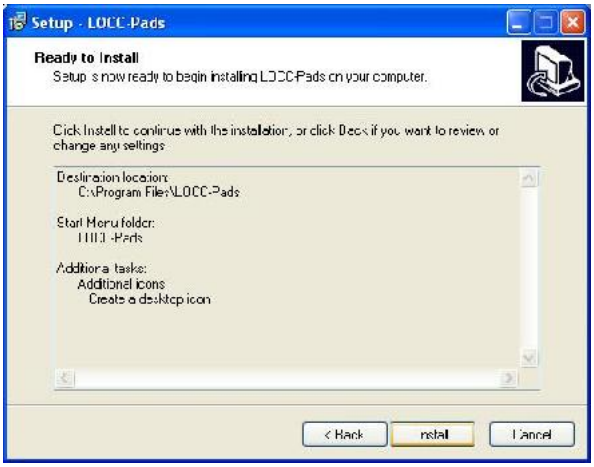
8. Additional functions

   Mark the checkbox ***Create a desktop icon*** to create a shortcut on the desktop. To confirm click ***Next***.



9. Installation

   All adjustments are displayed. Click ***Install*** to start the installation. Click ***Back*** if you wish to modify your adjustments.



10. End

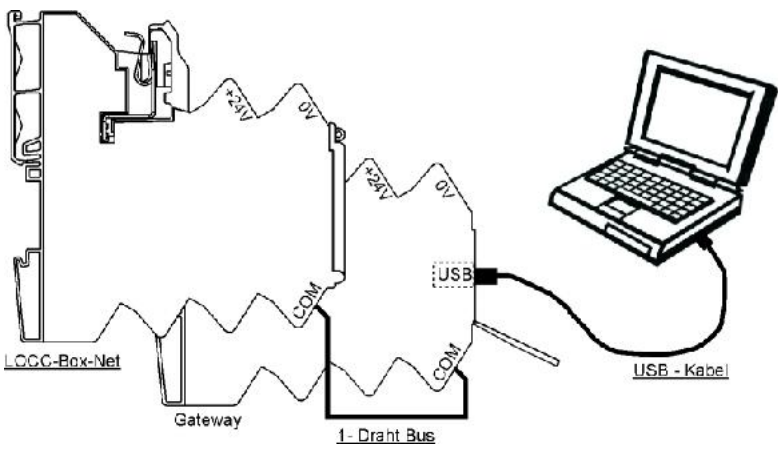
    Mark the checkbox ***Launch LOCC Pads*** to start LOCC Pads after installation. Click ***Finish*** to complete the installation.

3.3 USB connection

The picture shows the connection between LOCC-Box-Net and a computer via the USB connection. The connection to CANopen, Profinet and EtherCAT is described in *chapter* **Fehler! Verweisquelle konnte nicht gefunden werden.** *and* **Fehler! Verweisquelle konnte nicht gefunden werden.**. The Gateway is used in the example below. The USB cable is in the gateway scope of delivery. Please make sure that the software LOCC-Pads is already installed!

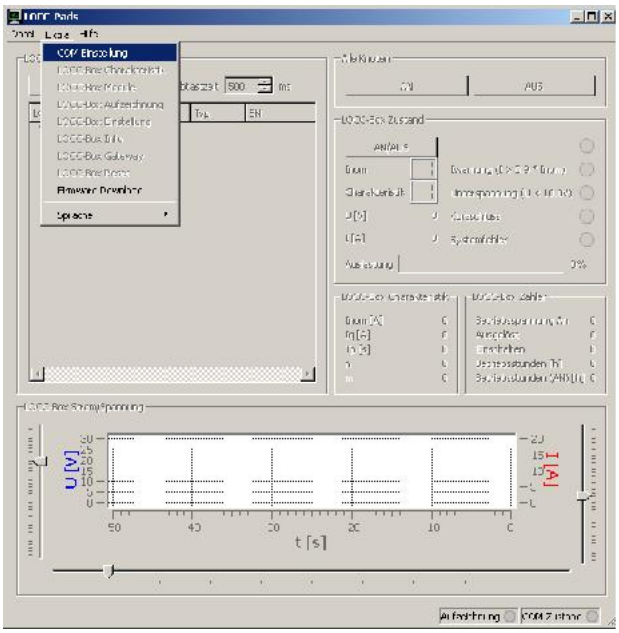
1. Supply the module, the LOCC-Box-Net and the Gateway with an operating voltage of DC 12/24V.

2. Connect all the COM connectors of the LOCC-Box-Net module to the Gateway. The jumper combs (*see chapter* **7***, Accessories*) are suitable for this purpose.



3. Connect the USB cable to the Gateway or rather to the computer. The computer will recognize the Gateway as ***New Hardware*** and, if necessary, the ***Assistant to search a new hardware*** will start. Further details can be found in the *chapter 4.1.6, 5.1.6* or *6.1.6*..

3.4 Adjustments



After connecting the modules as described in *chapter* **Fehler! Verweisquelle konnte nicht gefunden werden.**, start the software by double clicking on the LOCC-Pads symbol on the desktop or by ***Start >Programs>LOCC-Pads>LOCC-Pads.***
LOCC-Pads starts in an inactive mode. With LOCC-Pads it is possible to make adjustments in the COM settings, to download the firmware and to select a language.

NOTICE

If an error message occurs during the start, *see chapter* **Fehler! Verweisquelle konnte nicht gefunden werden.**.*1*.

3.4.1 Language, COM Config

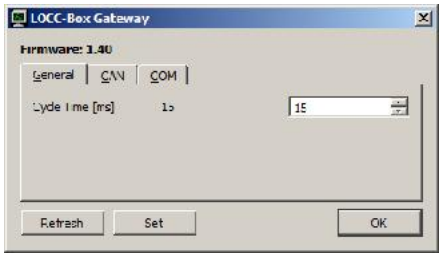
Click ***Extra > Language*** to choose a language (English or German).

To create a communication between computer and the Gateway, it is necessary to define the connection USB-Comport. The adjustment has to be made by clicking ***Extra > COM Config.***

-> The active state is indicated in the status line (COM-state) on the right bottom of the window.

3.4.2 Configuration LOCC-Box Gateway for CANopen (RS232) and EtherCAT

In order to use the Gateway for CANopen (RS232) and EtherCAT, it is necessary to make some basic setting.
Click ***Extra > LOCC-Box Gateway***.



The first line display the currently Firmware from the connected module.

- **General**
  - Cycle time (ms): Time for the cyclic inquiry of the Connected LOCC-Box-Net modules
- **CAN**
  - Node number: Assignment of the node number in the CANopen Network, maximal 254
  - Baud rate (Kbit/s): Adjustment of the data rates, min. 10 / max. 1000
- **COM**
  - Baud rate (bit/s): Adjustment of the data rates, min. 600 / max. 115200

All the values in gray can be read after the call of the window and show the present adjustments of the Gateway. With the button „Refresh“ it is possible to update the values and to save them with „Set“. “OK“ is used to close the window - new adjustments are not saved. At the top the currently firmware is additionally indicated.

3.4.3 Firmware Download

The Firmware Download is for updating the gateway software. It is only for service purposes (updates). This function is password protected.

3.5 Function via USB

The software LOCC-Pads is subdivided in following windows:

“LOCC-Box Nodes“,

“All Nodes“,

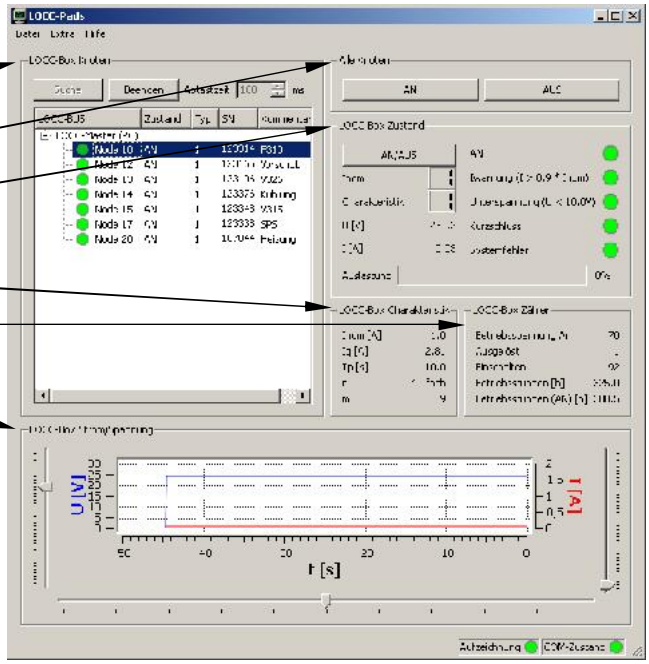
“LOCC-Box Status“,

“LOCC-Box Characteristic“,

“LOCC-Box Counter“ and

“LOCC-Box Current/Voltage“.

Further options and adjustments can be found in the menu item ***Extra***. The options are described below.



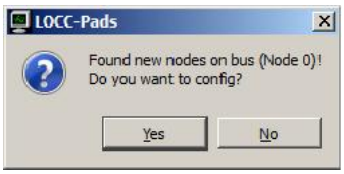
3.5.1 Window - LOCC-Box Nodes (Node Number Assignment)

This window offers the possibility to search modules, to adjust the start/stop record of the module data and the scan time. Also the node number, the device status, the type, the serial number and an editable comment are displayed.

Searching new modules with node number 0 (delivery status)

Modules with default settings always have the node number 0. To use the modules they have to be assigned by a node number between 1 and 254.

After clicking ***Search***, the internal bus is checked for connected modules. After the check the message ***Found new nodes...*** appears.
To configure those nodes click ***Yes***, to cancel click ***No***.



If choosing ***Yes*** all found modules are in the configuration mode.
The modules are flashing green. As a standard number 1 is suggested.
The number can be changed to any other number. To assign any module to that number press the push button of that module for 1 sec. The displayed number increments by 1 and

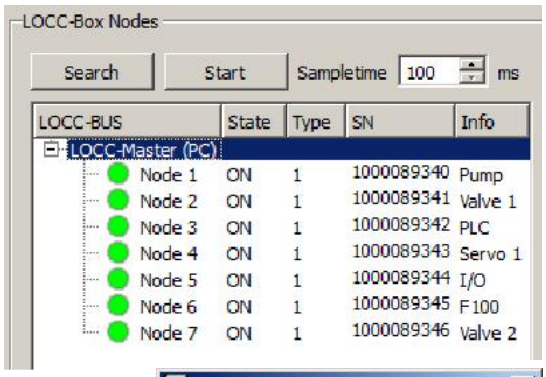
the next module can be assigned. This process has to be repeated till all modules are assigned to a node number.

NOTICE Assigned modules can be reset by choosing ***Extra > LOCC-Box Reset.*** See chapter 3.5.11.

Searching modules with a node number > 0

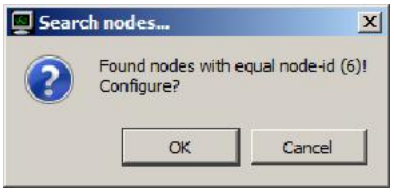
After clicking ***Search***, the internal bus is checked for connected modules. All modules with a node number between 1 and 254 will be displayed in the ***LOCC-Box Node*** window.

Modules with the node number 0 or the same node number are also recognized. These modules have to be configured.



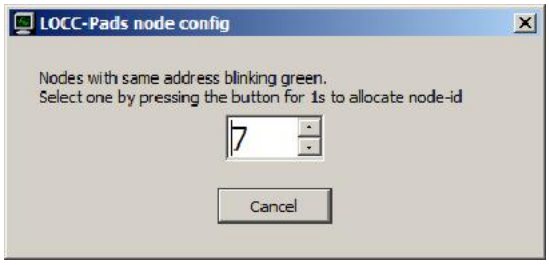
The window ***Search Node*** appears.

In the following example two modules with the node number 6 were found.



After configuring node number 7 the search restarts. This process will be repeated till all modules are assigned to a valid node number (accept 0).

To cancel click ***Cancel***.



NOTICE To change node numbers afterwards. The modules have to be in the stopping mode.

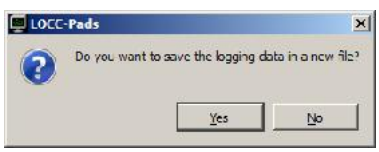
Scan time (cycle time)
The scan time (cycle time) can be adjusted to min.100 ms. This time indicates in which time period the modules are queried by the Gateway. Current and voltage runs which are smaller than the scan time are not recorded and therefore not indicated.

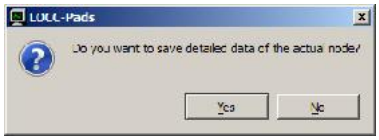
Start / End
This button is to start and stop the recording. After the start phase, all fields are active and the data are indicated by selected nodes. The status ***Logging*** is signalized in the status line.

When the Logging is finished, all windows are disconnected and are no longer updated. Confirm with ***Yes*** to stop the logging or with ***No*** to continue.

With ***Yes*** it is possible to confirm if the record has to be saved in Excel Format (+.CSV), clicking ***No*** for not saving.

With ***Yes*** it is possible to save the data of the current node in Excel Format (*.log). By clicking ***No*** the possibility of saving does not exist.



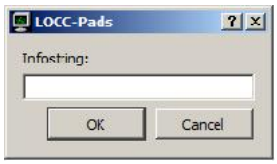


Status
Shows the status of the unit ***ON***, ***OFF***, ***Blown***, ***OFF (Switch)*** or ***EXT. OFF***.

Type
Shows the unit type. For example 716410 is type 1.

SN
Indicates the serial number of the modules.

Info string
With this option it is possible to enter an information for each node. The adjacent window can be opened by a double click on the suitable info field. This can be written or modified only if the recording is stopped. The information can have a length of max. 32 characters. The information is saved in the LOCC-Box. *See chapter* ***Fehler! Verweisquelle konnte nicht gefunden werden.***.



3.5.2 Windows – All Nodes

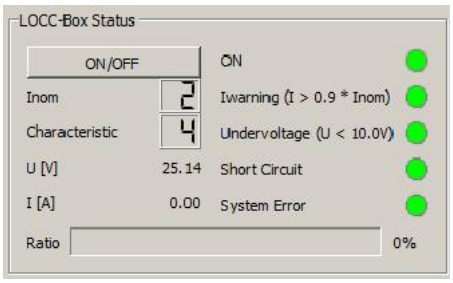
With this option it is possible to switch all modules ***ON*** and ***OFF*** at the same time.
Confirm the security query by clicking ***Yes*** to start the procedure or with ***No*** to abort it.

NOTICE

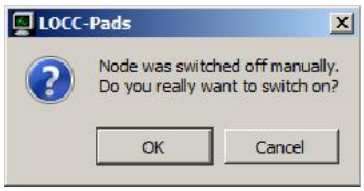
Nodes which are switched off manually cannot be activated with this function. These nodes can only be switched ***ON/OFF*** in the field ***LOCC-Box Status*** (*see chapter* ***Fehler! Verweisquelle konnte nicht gefunden werden.***) or alternately via the On/OFF button on the modules.

3.5.3 Windows - LOCC-Box Status

The following status and information are only related to the marked module. The represented lights show the condition of the status display of the module. The numbers in the fields ***Inom*** and ***Characteristic*** showing the settings of the current range and the characteristic curve. The adjacent output voltage and the currently flowing current are indicated in the field ***U[V]*** or ***I[A]***. The bar ***Ratio*** represents the chronological sequence of 0 – 100% of the load monitoring according to the current flow. This means the bigger the over current, the smaller the time until the LOCC-Box-Net switches off the load.



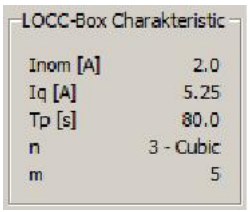
On / Off
Provides the possibility to switch the modules ON/OFF or acknowledge the modules after ***Blown***.



NOTICE

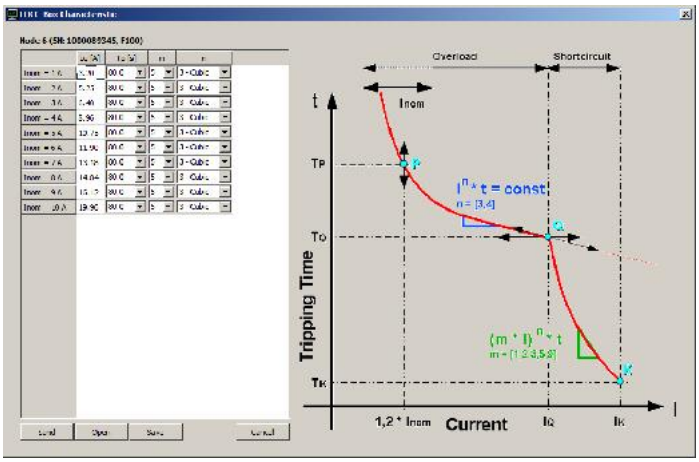
Modules which are switched off manually can only be switched on after confirming the security query with ***OK***.

3.5.4 Windows - LOCC-Box Characteristic



This field shows the parameters of the selected modules for the adjusted characteristic curve or rather the current range. Modifications by turning the selection switch ***Current Range*** or ***Characteristic*** are only updated after a restart of the module. The switch off / switch on can occur by the remote input, the software or by the module button.

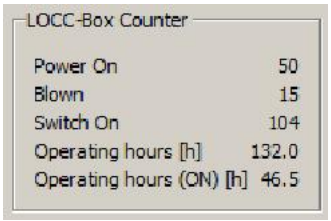
Under ***Extra > LOCC-Box Characteristic*** it is possible to program a characteristic curve and to send it to position 10 (Characteristic) on the LOCC-Box-Net. Besides that option it is possible to save and open the characteristic curves. By modifying the 4 parameters speedy or lethargically characteristic curves are possible.
The shown diagram is for better comprehension of the possible settings. The diagram is not adapted to the adjustments.



3.5.5 Window - LOCC-Box Counter

The LOCC-Box modules have internal counters, which are read out by the software and displayed. These counters cannot be deleted.

Power ON
Indicates how many times the LOCC-Box has been connected to the supply voltage.



Blown
Indicates how many times the device has switched off because of overload or short circuit.

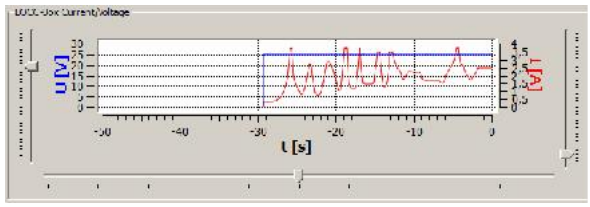
Switch On
This is the total of how many times the module has switched on (External / Software / Pushbutton).

Operating hours [h]
Shows the total of the operating hours how long the LOCC-Box has been switched on to the supply voltage.

Operating hours (ON) [h]
Indicates the total time how long the load was or is switched on.

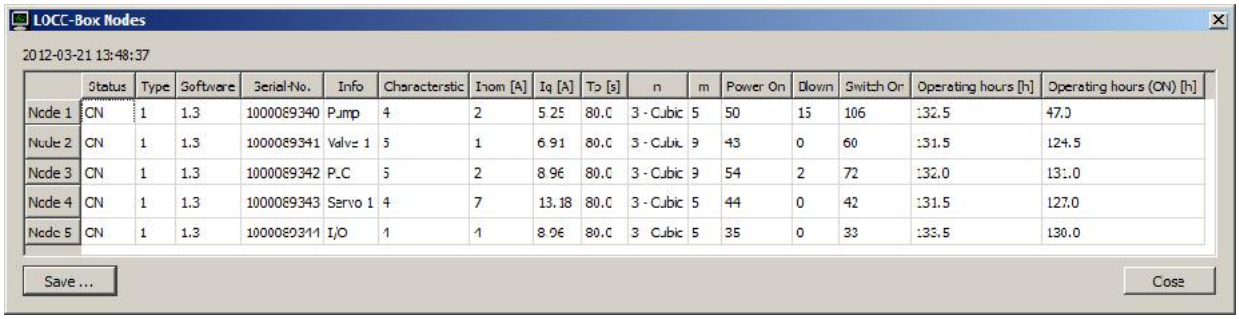
3.5.6 Window - LOCC-Box Current/Voltage

The represented ***Plotter*** shows the chronological sequence of output voltage and current for the selected nodes (module). The diagrams for current, voltage and time can be adjusted by the sliders. It is possible to save the value – see chapter 3.5.1



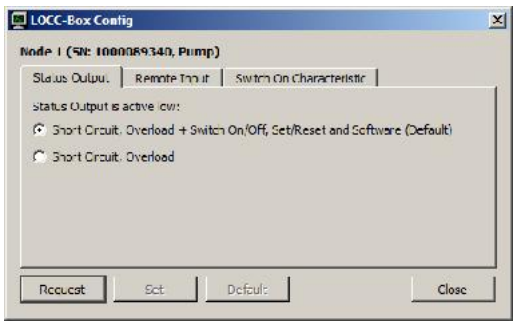
3.5.7 Extra / LOCC-Box Modules

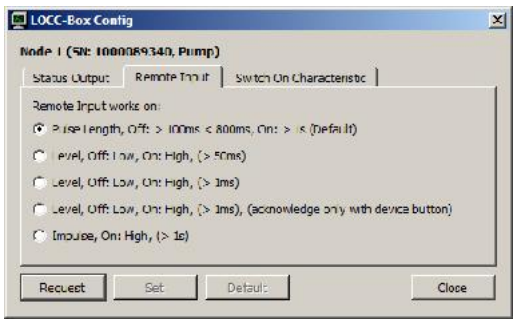
This menu item displays a list of all connected and identified devices at the moment of the search run. These information are updated only during the search run and can be saved as a CSV file which is compatible with Excel.

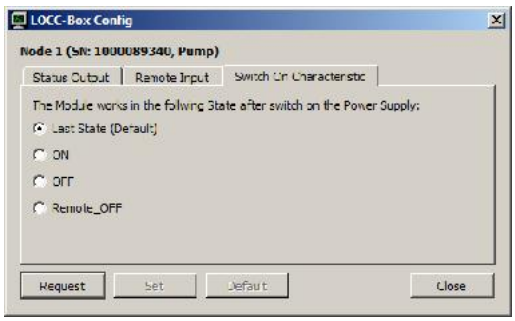


	Status	Type	Software	Serial-No.	Info	Characteristic	Inom [A]	Iq [A]	Tb [s]	n	m	Power On	Blown	Switch On	Operating hours [h]	Operating hours (ON) [h]
Node 1	ON	1	1.3	1000089340	Pump	4	2	5.25	80.0	3 - Cubic	5	50	15	106	132.5	47.0
Node 2	ON	1	1.3	1000089341	Valve 1	5	1	6.91	80.0	3 - Cubic	9	43	0	60	131.5	124.5
Node 3	ON	1	1.3	1000089342	P_C	5	2	8.96	80.0	3 - Cubic	9	54	2	72	132.0	131.0
Node 4	ON	1	1.3	1000089343	Servo 1	4	7	13.18	80.0	3 - Cubic	5	44	0	42	131.5	127.0
Node 5	ON	1	1.3	1000089311	I/O	1	1	8.96	80.0	3 - Cubic	5	35	0	33	133.5	130.0

3.5.8 Extra / LOCC-Box Config







➢ **Status Output**

The status output is suitable for the error recording and can be parameterized for two operating modes. The High level is available if the output is switched on. The status changes to low if the output is switched off by overload or short circuit. The additional mode for changes to low is the pushbutton, remote input or the software. (Default, as standard type 716401)

➢ **Remote Input**

The remote input provides 5 adjustments for switching the modules on/off. This is possible by adjusting the impulse length or the level.
The impulse length on/off corresponds the default setting.(716400, 716401, 716410)

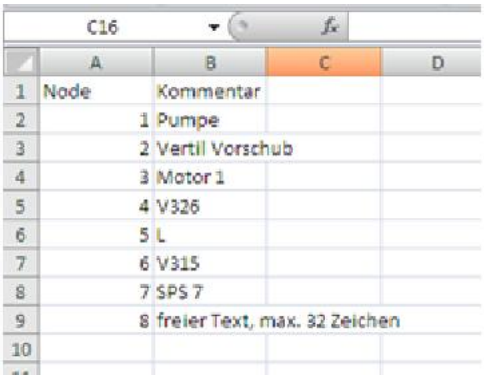
➢ **Switch ON Characteristic**

This adjustment describes the status after switching on the power supply. The default adjustment is the last state.

By clicking the button ***Request*** the current adjustments are displayed. By clicking ***Set*** the settings are send on the LOCC-Box. By clicking ***Default*** the default values will be write in the LOCC-Box.

3.5.9 Extra / LOCC-Box Info

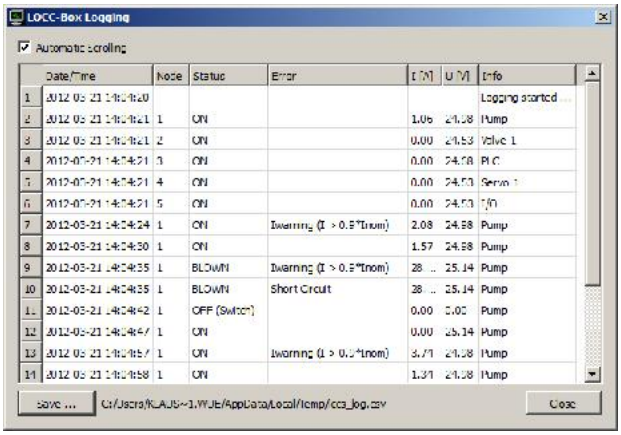
LOCC-Box Info provides the import of comments. A description how to change comments can be found *in chapter 3.5.1*. For the import a CSV file with any name is needed.



	A	B	C	D
1	Node	Kommentar		
2	1	Pumpe		
3	2	Vertil Vorschub		
4	3	Motor 1		
5	4	V326		
6	5	L		
7	6	V315		
8	7	SPS 7		
9	8	freier Text, max. 32 Zeichen		
10				

3.5.10 Extra / LOCC-Box Logging

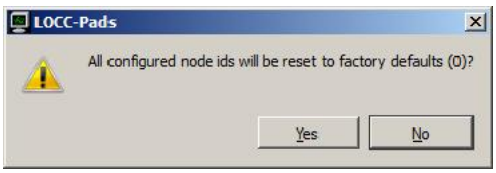
After starting the window ***LOCC-Box Logging***, all events are saved in a temporary file on the computer. The file can be opened via the menu and can be saved as a CSV file. If shutting down LOCC-Pads an inquiry for saving the file appears. Choose ***Yes*** or ***No***.
All events like over current, switching on/off for nodes with date/time, status, error current/voltage values and comments are logged.
The start of logging is displayed by a green light in the status line.



3.5.11 Extra / LOCC-Box Reset

Configured modules with a node number >0 can be set to the default status (0) by resetting. Comments, settings and counters are not reset. It is useful if using the modules with node numbers in a new order. This operation is password protected to avoid an accidentally reset.
The password is: ***Reset Luetze***.

Following alert window /confirmation message appears:



The modules have to be disconnected 5 s from the DC power supply after confirming by clicking ***Yes***.

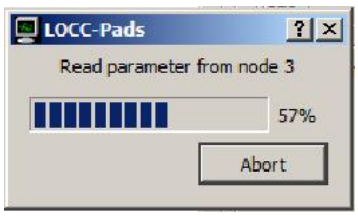
3.6 Configuration Saving / Open

With LOCC-Pads it is possible to save configurations and open existing ones.

Configurations are:
- Adjustments of the status output, remote input and the behavior of the switch on procedure - *see chapter 3.5.8*
- Info strings, *see chapter 3.5.1*
- Settings of the characteristic curve of position 10, please *see chapter 3.5.4*
- Setting of the rotary switches for the ***Current Range (I)*** and the ***Characteristics (C)***.
- Modules with Node number

3.6.1 Saving

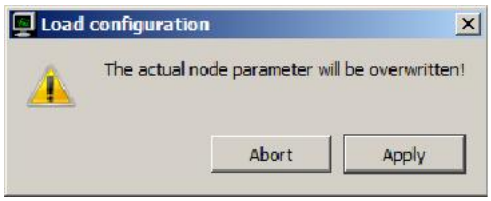
At least one module has to be connected and the modules have to be in the stop mode. Click File>Save Configuration



3.6.2 Open

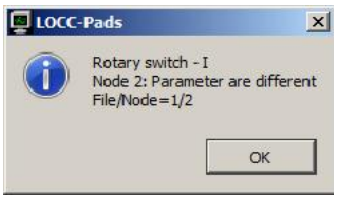
To open an existing configuration at least one bus node has to be connected and the modules have to be in the stop mode. Click ***File > Open Configuration.***

Confirm the loading of the configuration by clicking ***Apply***. To cancel the loading, click ***Abort***.

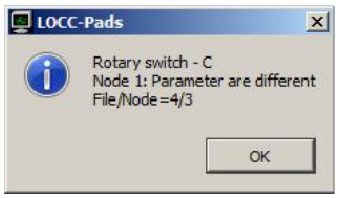


Invalid configuration like rotary switch settings of the current or characteristics or not existing node numbers are alert by a message window.

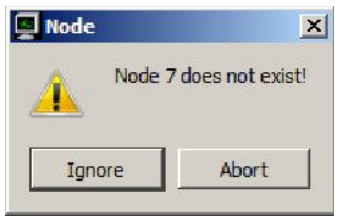
Example: The rotary switch **I** is set to 2 on node 1, but 1 is saved in the configuration file.
→ Set the rotary switch to 1 and switch the module off and on again.



The rotary switch **C** is set to 3 on node 1, but 4 is saved in the configuration file.
→ Set the rotary switch to 4 and switch the module off and on again.



The module with the node number 7 is saved in the configuration, but it does not exist.



NOTICE Modules which are not saved in the configuration file are not considered when loading.

4 Gateway – CANopen, 716459

The LOCC-Box Gateway is an electronic part which distributes and transforms the data and messages of the serial LOCC-Box-Net interface to 3 further communication interfaces USB, CANopen or RS232.

4.1 General Information

4.1.1 Explanation

The serial LOCC-Box-interface (LOCCbus) is a 1 wire–communication interface according to the LIN specification. The protocol on this interface is programmed according to the Multidrop Protocol. The detailed configuration of the LOCC-Box-Interface can be found in *chapter* **Fehler! Verweisquelle konnte nicht gefunden werden.** *(RS-232) and 4.4 (CANopen).*

The Gateway supports following interfaces:

- Full-Speed USB-interface with a max Bitrates of 12 MBit/s according to USB 2.0
- CAN-Interface according to ISO 11898-1, physical transmission layer is the CAN-High-Speed-Layer according to ISO 11898-2
- serial interface RS232

The USB-Interface is for connecting the LOCC-Box-Net to a common computer or notebook. The USB-interface is recognized by Windows XP [1)] or Windows Vista [1)] as a serial COM-Interface. With LOCC-Pads the interface is used for the initial operation and configuration of the LOCC-Box-Net boards (716410).

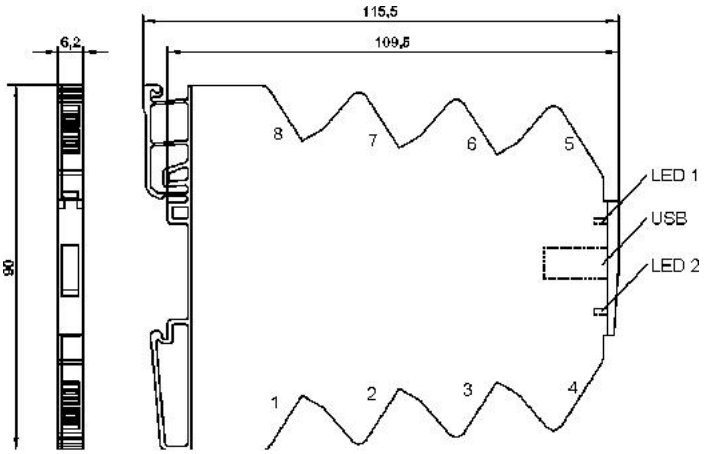
The CANopen and serial RS232 interface are used for connecting the LOCC-Box-net to any programmable logic controller (PLC) of different manufacturers for example.

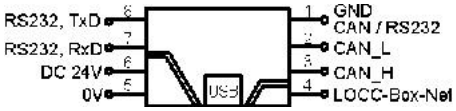
A simultaneous operation mode of the USB- and CANopen-interface is not possible. In this case the communication via the USB interface always has priority. The communication via the CANopen-interface is switched Off, the LOCC-Box-Net Interface is not available as electrical CANopen-user (no CAN-Acknowledge!).

The CANopen-interface functions according to the CANopen-Protocol CiA DS301; as CANopen-Profile the *Generic IO Profile* according to the CiA DS401 is used. The baud rate and the CANopen-nodes-Addresses (Node-ID) can be adjusted in LOCC-Pads.

The communication connection of the 3 Gateway interfaces (RS232, USB and CANopen) is exclusively positioned on the interface of the LOCC-Box-Net modules. The modules are not designed for a transverse communication (USB <-> RS232 <-> CANopen) among each other.

4.1.2 Dimensions and Connections:





1:	GND - CANopen / RS232
2:	CANopen connection - CAN_L
3:	CANopen connection - CAN_H
4:	1 wire Bus (LOCC-Box-Net)
5:	0V (supply Gateway)
6:	DC 12/24V (+supply Gateway)
7:	RS232 connection - RxD
8:	RS232 connection – TxD

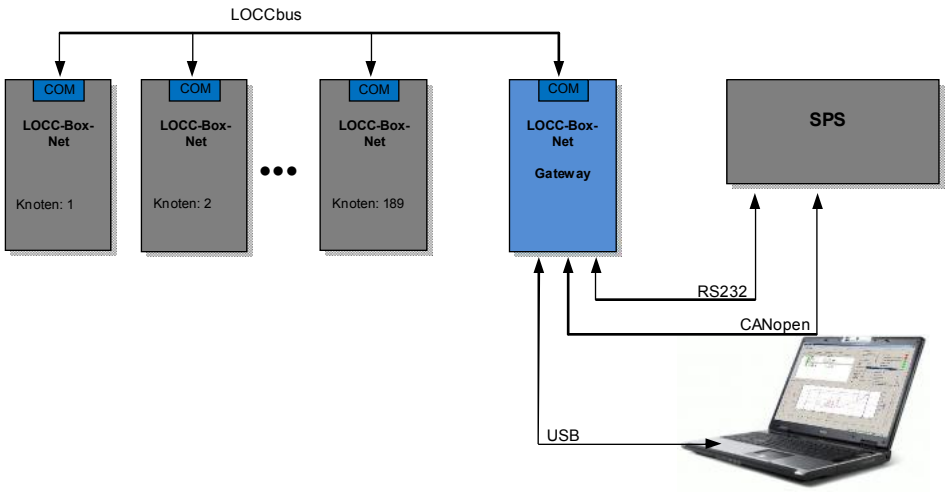
USB:	USB connection
LED 1:	USB status in case of data Exchange via UART
LED 2:	CANopen status

4.1.3 Function and Display

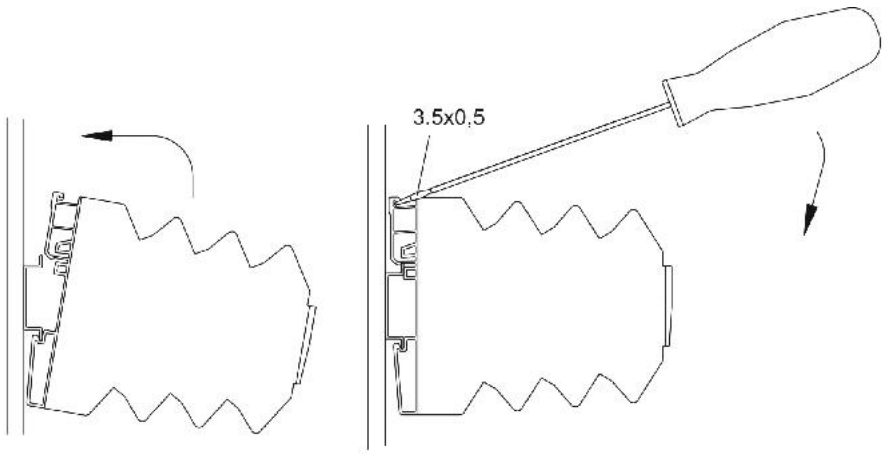
Function	PIN	Description
GND	1	Ground potential for CANopen and RS232, connection for shielding.
CAN_L	2	Connection of the CAN_Low cable, load resistance typical 120R, must be set externally
CAN_H	3	Connection of the CAN_High cable, load resistance typical 120R, must be set externally
1 wire bus	4	1 wire communication bus for the connection of typical 40 and max 84 LOCC-Box-Net modules, see topology. The bus length is up to 40m.
- connection, own supply	5	Has to be connected directly to the voltage supply.
+ connection, own supply	6	Has to be connected directly to the voltage supply.
RS232_RxD	7	RS232 received data cable
RS232_TxD	8	RS232 transmission data cable

Display	Function	Description
LED 1, green – lighting	USB	USB is active
LED 1, Green – with short off	USB + RS232	USB is active. If an additional data exchange occurs through the RS232 interface, the LED for this time is blanked.
LED 1, green – with short on	USB + RS232	USB is not active. If a data are exchanged via the RS232 interface, the LED flashes for this time.
LED 1, rot – lighting	Firmware	The Firmware has been deleted. If the LED 2 flashes in red as well, the Gateway is in the Firmware-Update-Mode.
LED 2, green – blinking	CANopen	CANopen is active – pre-operational mode
LED 2, green – with short on	CANopen	CANopen is active - stop mode
LED 2, green - lighting	CANopen	CANopen is active – operation mode
LED 2, red – with short on	CANopen	CANopen is active - BUS_WARN message
LED 2, red - lighting	CANopen	CANopen is active - BUS_OFF message, together with LED1 red lighting the gateway is in the firmware update mode.

4.1.4 Topology and Structure



4.1.5 Mounting



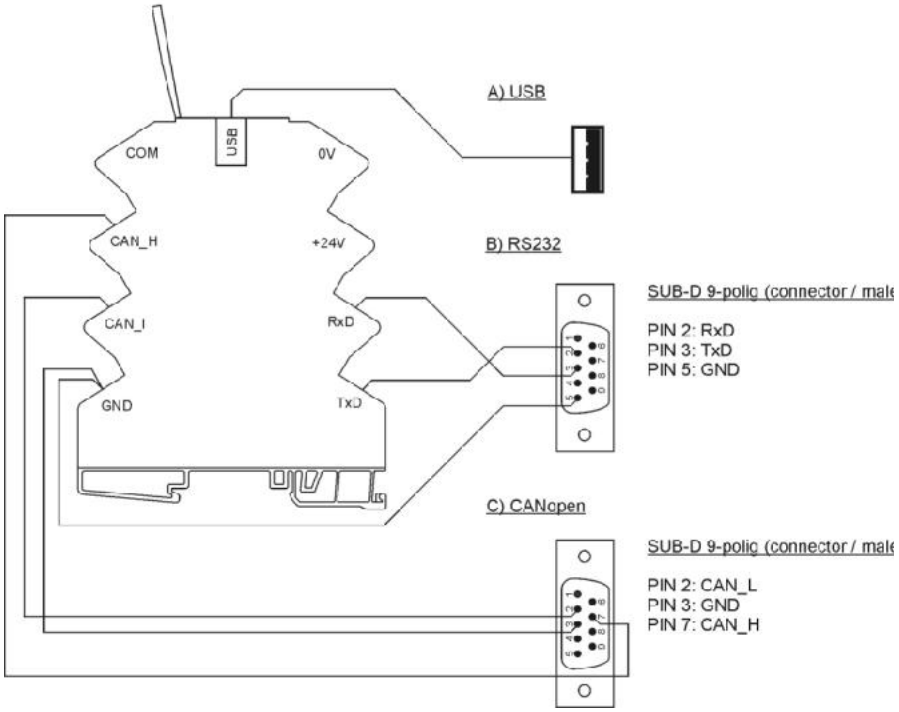
10	0,25 ... 2,5 mm²
10	0,25 ... 2,5 mm²
10	0,25 ... 1,5 mm²
AWG	23 to 14

4.1.6 Installation

1. Supply the Gateway and all LOCC-Box-Net modules with an operating voltage of DC 12/24V

2. Connect the ***COM*** connectors of the modules to the Gateway. For this purpose jumper combs can be used. *See chapter* **7**.

3. Connect the Gateway as follows:
   A) USB
   B) RS232
   C) CANopen



Connection to USB

Connect the Gateway to the computer by using the provided USB cable.
At the initial connection, the Gateway will find a new Hardware ***LOCC-Box-Net Interface 716459*** and the ***Found new Hardware wizard*** will prompt.

Choose ***Install the software automatically*** and confirm by clicking ***Next***. Follow the instructions of the wizard, which searches and installs the ***loccbn.inf*** driver.

A safety Warning will prompt. Confirm by clicking ***Continue Anyway***.

4.2 Communication via USB

See chapter **Fehler! Verweisquelle konnte nicht gefunden werden.** *LOCC-Pads.*

4.3 Communication via RS232

The UART interface RS232 allows the communication between the Gateway (LOCC-Box-Net) and a terminal program, for example Microsoft Hyper Terminal, or a PLC. The telegrams begin with a <STX> (ASCII 0x02) and end with a <ETX> (ASCII 0x03).
<STX> is the key combination ***Strg+B*** and <ETX> ***Strg+C***. The single numbers are transmitted as ASCII-HEX. Use the following setup:

- Baud rate: 600 - 115200 bit/s
- Data bit: 8
- Parity: none
- Stop bits: 1
- Flow control: none

The communication allows to write and to read values and the status. The Information corresponds to the version as in the Software LOCC-Pads. The following information can be read:

- Module type
- Module state / configuration:
  - Module ***Off***, ***On***, ***Blown*** and ***EXT.Off***
  - New Modules on the BUS
  - Current warning (I>90%), switching off due to over current
  - Under voltage
  - switching off due to short circuit
  - System failure
  - Switch setting of current range
  - Switch setting of characteristic
- Input voltage
- Output voltage
- Current
- Adjustments of the characteristic curves
- Software version
- Serial number
- Counter reading

It is possible to write the operating status ON / OFF.

Example: HyperTerminal

1. Start the program. Click ***Start>Programs>Accessories>Communication> HyperTerminal*.**

2. Enter a name for the new connection and assign an icon to it. For example the Lütze icon.
   Confirm by clicking ***OK*** to continue or ***Cancel*** to abort.

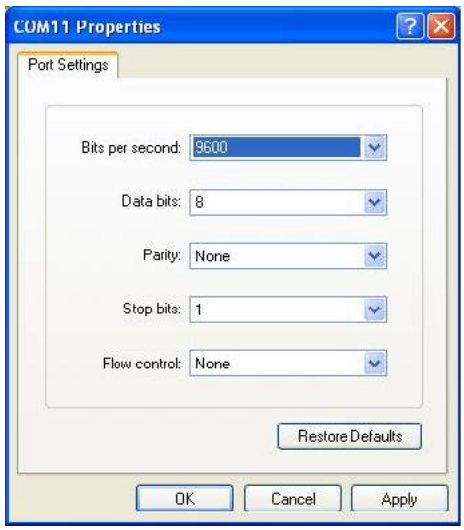
3. From the drop down list ***Connect using*** choose the COM1 interface where the Gateway is connected to.

NOTICE

In case you do not know which connection is the right one, check it with the ***Device-Manager***.
Open the folder ***System*** by clicking ***Start > Settings > Control Panel.*** The ***Device Manager*** can be found under ***Hardware***.

4. Enter all properties for the communication. Use following adjustments:

   - Baud rate: 600 - 115200 bit/s
   - Data bit: 8
   - Parity: none
   - Stop bits: 1
   - Flow control: none
   -



NOTICE

The baud rate has to be the same as the baud rate of the Gateway.
See chapter **Fehler! Verweisquelle konnte nicht gefunden werden.** *"Configuration LOCC-Box Gateway for CANopen and RS232".*

5. Adjustment of the local Echo

   - Stop the connection by clicking ***Disconnect***.

   - Under ***File>Properties*** the menu Properties prompts. Click the slide ***Settings***.

   - Under ***File>Properties*** the menu Properties prompts. Click the slide ***Settings***.

4.3.1 How to read values and status

Query	Input - command	Output
	***<STX>* = Strg+B / *<ETX>* = Strg+C**	
Software version	<STX> nn 50 <ETX> Module command (Nodes)	<STX> XX YY <ETX> = MSB (most signification bit) = LSB (least significant bit) MSB, LSB (Hex) = Version, with dot 1 step to the left side
	Example: Input: <STX>0150<ETX>, module query (nodes) 1 Output: <STX>04 00<ETX> = 00 04 = *decimal value* = 0.4	
Serial number	<STX> nn 51 <ETX> Module command (Nodes)	<STX> WW XX YY ZZ <ETX> = MSB (most signification bit) = LSB (least significant bit) MSB … LSB (Hex) = conversion into *decimal value* = Serial number
	Example: Input: <STX>0151<ETX>, module query (nodes) 1 Output: <STX>EE E1 01 00<ETX> = 00 01 E1 EF = *decimal value* = 123375	
LOCC-Box Counter Operating voltage ON	<STX> nn 52 <ETX> Module command (Nodes)	<STX> XX YY ZZ <ETX> = MSB (most signification bit) = LSB (least significant bit) MSB … LSB (Hex) = conversion into *decimal value* = counter
	Example: Input: <STX>0B52<ETX>, module query (nodes) 11 Output: <STX>0C 01 00<ETX> = 00 01 0C = *decimal value* = 268	
LOCC-Box Counter “Operating hours (h)“	<STX> nn 53 <ETX> Module command (Nodes)	<STX> XX YY ZZ <ETX> = MSB (most signification bit) = LSB (least significant bit) MSB … LSB (Hex) = conversion into *decimal value* / 2 = Operating hours (h)
	Example: Input: <STX>0C53<ETX>, module query (nodes) 12 Output: <STX>60 01 00<ETX> = 00 01 60 = *decimal value* / 2 = 176 h	
LOCC-Box Counter “Operating hours ON (h)“	<STX> nn 54 <ETX> Module command (Nodes)	<STX> XX YY ZZ <ETX> = MSB (most signification bit) = LSB (least significant bit) MSB … LSB (Hex) = conversion into *decimal value* / 2 = Operating hours (h)
	Example: Input: <STX>0D54<ETX>, module query (nodes) 13, output: <STX>FB 08 00<ETX> = 00 08 FB = *decimal value* / 2 = 1149,5 h	

Query	Input – Command	Output
	***<STX>* = Strg+B / *<ETX>* = Strg+C**	
LOCC-Box Counter "Blown"	<STX> nn 55 <ETX> Module (Nodes) command	<STX> XX YY ZZ <ETX> = MSB (most signification bit) = LSB (least significant bit) MSB … LSB (Hex) = conversion into *decimal value* = Blown
	Example: Input: <STX>1155<ETX>, module query (nodes) 17, Output: <STX>28 00 00<ETX> = 00 00 28 = *decimal value*= 40	
LOCC-Box Counter "Switch On"	<STX> nn 56 <ETX> Module (Nodes) command	<STX> XX YY ZZ <ETX> = MSB (most signification bit) = LSB (least significant bit) MSB … LSB (Hex) = conversion into *decimal value* = Switch On
	Example: Input: <STX>0E56<ETX>, module query (nodes) 14, Output: <STX>11 00 00<ETX> = 00 00 11 = *decimal value* = 17	
Module state/ Configuration	<STX> nn 04 <ETX> Module (Nodes) command	<STX> X_2 X_1 Y_2 Y_1 <ETX> = decimal 0-9 = current range 1-10A = decimal 0-9 = characteristic 1-10 (binary) 0 0 0 0 00: OFF (Switch / LOCC-Pads) 01: ON 10: Blown 11: EXT. OFF reserved reserved (binary) 0 0 0 0 1: Iwarnung (I > 0.9 * Inom) 1: Under voltage (U < 10.0V) 1: Short circuit 1: System failure
	Example: Input: <STX>0204<ETX>, module query (nodes) 2 Output: <STX>4 2 9 1<ETX> = decimal 1 = current range 2A = decimal 9 = characteristic 10 0 0 1 0 10: Blown 0 1 0 0 1: Short circuit	
Module type	<STX> nn 00 <ETX> Module (Nodes) command	<STX> X_2X_1 <ETX> = decimal = Type number
	Example: Input: <STX>0A00<ETX>, module query (nodes) 10 Output: <STX>01<ETX> = decimal = Type 1	

Query	Input – Command	Output
	***<STX>* = Strg+B / *<ETX>* = Strg+C**	
Input voltage	<STX>nn 0503<ETX> Module (Nodes) command	<STX> XX YY <ETX> = MSB (most signification bit) = LSB (least significant bit) MSB, LSB (Hex) = conversion into *Decimal value* The max measuring value is 1024 and equates 39V. The result is the following rule of three: Input voltage = $\frac{\textit{Decimal value} \times 39V}{1024}$
	Example: Input: <STX>140503<ETX>, module query (nodes) 20 Output: <STX>84 02<ETX> = 02 84 = *Decimal value* = 644 Input voltage = $\frac{644 \times 39V}{1024}$ = 24,53V	
Output voltage	<STX>nn 0502<ETX> Module (Nodes) command	<STX> XX YY <ETX> = MSB (most signification bit) = LSB (least significant bit) MSB, LSB (Hex) = conversion into *Decimal value* The max measuring value is 1024 and equates 39V. The result is the following rule of three: Output voltage = $\frac{\textit{Decimal value} \times 39V}{1024}$
	Example: Input: <STX>210502<ETX>, module query (nodes) 33 Output: <STX>87 02<ETX> = 02 87 = *Decimal value* = 647 Output voltage = $\frac{647 \times 39V}{1024}$ = 24,64 V	
Current measurement	<STX>nn 0511<ETX> Module (Nodes) command	<STX> XX YY <ETX> = MSB (most signification bit) = LSB (least significant bit) MSB, LSB (Hex) = conversion into *Decimal value* The max measuring value is 1024 and equates 32,75A. The result is the following rule of three: Current = $\frac{\textit{Dezimalwert} \times 32,75A}{1024}$
	Example: Input: <STX>0F0511<ETX>, module query (nodes) 15 Output: <STX>1F 00<ETX> = 00 1F = *decimal value* = 31 Current = $\frac{31 \times 32,75A}{1024}$ = 0,99 A	

Query	Input - Command	Output
	***<STX>* = Strg+B / *<ETX>* = Strg+C**	
Adjustment of the characteristic curves	<STX>nn 051D<ETX> Module (Nodes) command	<STX> XX Y Y <ETX> (binary) 0 0 0 0 0 0 0 0 00: m = 2 01: m = 3 10: m = 5 11: m = 9 0: n = 3-Cubic 1: n = 4-Quad 000: Tp(s) = 1,25 001: Tp(s) = 2,5 010: Tp(s) = 5,0 011: Tp(s) = 10,0 100: Tp(s) = 20,0 101: Tp(s) = 40,0 110: Tp(s) = 80,0 reserved conversion into *Decimal value,* The max measuring value is 256 and equates 32,75A. The result is the following rule of three: $Iq = \frac{(256 - \text{Decimal value}) \times}{32{,}75A}$
	Example: Input: <STX>0A051D<ETX>, module query (nodes) 10 Output: <STX>99 1F<ETX> 00 01 1 1 11 11: m = 9 1: n = 4-Quad 011: Tp(s) = 10,0 = *Decimal value* = 153 $Iq = \frac{(256 - 153) \times 32{,}75A}{256} = 13{,}18\ A$	

4.3.2 Writing On / Off status

Query	Input - Command	Output
	***<STX>* = Strg+B / *<ETX>* = Strg+C**	
Status On / Off	<STX> nn 84 xx <ETX> Module (Nodes) Command (write) Instruction (On / Off) **XX:** 00 = Module switch off 01 = Module switch on	As answer the new operation status is restored. The answer can be found according „module status / configuration“ on page 25.
	Example: Input: <STX>088401<ETX>, switch On module (Node) 8 Output: <STX>09 25<ETX>, 09 = The module is on. 25 = Current range (5+1), Characteristic 3 (2+1)	

4.4 CANopen Communication

On the CAN Bus the Gateway performs as a CANopen node according to DS301 with a profile following DS401 (Generic I/O).

4.4.1 Terms and Definitions

CANopen	CANopen is a protocol which has been developed originally for industrial control systems. www.can-cia.org
COB	Communication Object
Data rate	The data rate is the number of data which can be transmitted in a certain time.
EDS-Files	Electronic Data Sheet Is provided by the manufacturer of a CANopen device. It has a standard format for the description of devices. The EDS-file contains information about the description of the file, about general device information and about the description of the supported objects.
Emergency-Id	Emergency Data Object
ID	Identifier
Node number	Within a CANopen-Network it is possible to identify every device by its node number (Node-ID). The allowed node numbers are in the range from 1 to 127 and can appear only one time in a network.
LEN	Length - Length of the data
NMT	Network Management (Master)
PDO	Process Data Objects The PDOs are used for the transmission of the process data. In 'Receive'-PDO (RxPDO) the Process Data are received from the LOCC-Box Gateway. Via the 'Transmit'-PDO (TxPDO) the Gateway sends data to the CANopen net.
PDO-Mapping	Die The PDOs are used for the transmission of the process data. In 'Receive'-PDO (RxPDO) the Process Data are received from the LOCC-Box Gateway. Via the 'Transmit'-PDO (TxPDO) the Gateway sends data to the CANopen net.
RTR	Remote Transmit Request
Rx	Receive
SDO	Service Data Object The SDOs are used for the transmission of the configuration and parameter data for example the output current. In contradiction to the PDOs, the SDO messages will be confirmed. After writing or reading request on a data object, a confirmation or a failure telegram follows.
Sync	Sync (frame) Telegram
Tx	Transmit

4.4.2 NMT-Boot-up

The Gateway can be initialized with the CiA-Draft Standard 301, as described in chapter 9.4 'Minimal - Boot-up'. In this example the Gateway is used with the NodeID 5.

Instruction	Identifier [Hex]	Instruction code	Node-ID of the Gateway
NMT - "Operation Mode"	0000	01	05_h
NMT - "Pre-Operation Mode"	0000	80	05_h

The Operation Mode support additional Process-Data-Objects (PDO)!

4.4.3 The CANopen Directory

The object directory is an assorted grouping of objects, which is accessible via the network. Each object in every directory is addressed with a 16-Bit-Index. In the object directory the index is indicated in a hexadecimal form.

The index can be a 16-Bit-Parameter code according to the CANopen specification (CiA DS-301), or a manufacturer specific code. With the more significant bits of the index, it is possible to identify in which object group the parameter belongs to. Following parameters belonging to the object directory:

Index [Hex]	Object	Example
0001 … 009F	Definition of data types	-
1000 … 1FFF	Communication Profile Area	1001_h : error-register
2000 … 5FFF	Manufacturer Specific Profile Area	2000_h : module-type
6000 … 9FFF	Standardized Device Profile Area	according to the application profile DS-40x
A000 … FFFF	Reserved	-

4.4.3.1 Access to the Object Directory via SDOs

With the SDOs (Service Data Objects) it is possible to access the object directory of a device. An SDO represents a channel to access the parameter of the device. The access is possible in the operational and pre-operational mode. The SDOs are transmitted with ID ***600_h + NodeID*** (Request). The receiver confirms the parameters with ID ***580_h + NodeID*** (Response).

The SDO is configured as follows:

Identifier (600_h+ Node-ID)	Com-mand code	Index (low)	Index (high)	Sub-Index	LSB	Data field	MSB

Example - Sending (Request):

605_h (Node-ID 5 Gateway)	40_h (read)	00_h	20_h	01 (LOCC-Box node 1)	00_h	00_h	00_h	00_h
		(Index=2000_h) (Transmit-SDO-module type)						

Example - Receiving (Response):

585_h (Node-ID 5 Gateway)	$4F_h$ (1 Byte)	00_h	20_h	01 (LOCC-Box node 1)	01_h	00_h	00_h	00_h
		(Index=2000_h) (Receive-SDO-module type)			value = 00 00 00 01 (MSB … LSB)			

Description of the SDOs:

Identifier: The parameters are transmitted with ID ***600_h + NodeID*** (Request). The receiver confirms the parameters with ID ***580_h + NodeID*** (Response).

Instr. Code: The instruction code consists the instruction specifies and the length.
The most needed combinations for example are:
$40_h = 64_d$: Read Request, this means a parameter has to be read
$23_h = 35_d$: Write Request with 32 Bit data, this means a parameter has to be placed
The Gateway responds to each received telegram with an answer telegram.
This can include the following instruction codes:
$43_h = 67_d$: Read Response with 32 Bit data, this telegram contains the required parameters
$60_h = 96_d$: Write Response, this means a parameter has been set successful
$80_h = 128_d$: Error Response, this means the CAN module reports a failure.

The following table gives an overview about the most used instructions codes. The command frames must always include 8 data bytes. Information about syntax and further instruction codes are contained in the CiA DS-301.

Instruction	Count of data bytes	Command code [Hex]
Write Request (Initiate Domain Download)	1 2 3 4	2F 2B 27 23
Write Response (Initiate Domain Download)	-	60
Read Request (Initiate Domain Upload)	-	40
Read Response (Initiate Domain Upload)	1 2 3 4	4F 4B 47 43
Error Response (Abort Domain Transfer)	-	80

Index, Sub-Index Index and Sub-Index are described in *chapter Device Profile Area,* **Fehler! Verweisquelle konnte nicht gefunden werden.** and "*Manufacturer Specific Profile Area,* **Fehler! Verweisquelle konnte nicht gefunden werden.**" of this manual.

Data field … The data field can be maximum 4 bytes long. The low-order byte (LSB) comes first and the more important byte (MSB) last. The low-order byte is always in ***Data 1***, in case of 16-Bit-values the more important byte (bits 8...15) is in ***Data 2***, and in case of 32-Bit-values the MSB is (bits 24...31) in ***Data 4***.

4.4.3.2 Failure codes of the SDO-Domain-Transfers

The following failure codes can occur during the application (according to CiA DS-301, chapter *Abort SDO Transfer Protocol*):

Failure code [Hex]	Explanation
05030000	Toggle Bit unchanged
05040001	Wrong command specified
06010002	Writing access is wrong
06020000	Wrong index
06040041	The object cannot be mapped onto PDO
06040043	The parameters are incompatible
06060000	No access – hardware failure
06070010	Wrong number data bytes
06070012	Length of the service parameters is too big
06070013	Length of the service parameters is too small
06090011	Wrong Sub-Index
06090030	Parameter has been sent out of the last value range
08000000	Failure cause not defined
08000020	Data cannot be transmitted or saved
08000022	With the present device status, the data cannot be transmitted or saved
08000024	Access to flash failed

4.4.4 Overview of the CANopen-Identifier used

Function	Identifier [Hex]	Remarks
Network management	0	NMT (operation-/ pre-operation mode)
SYNC	80	Sync to each one, (configurable by Object 1005_h)
Emergency Message	80 + *Node-ID*	Configurable by Object 1014_h
Tx-PDO1	180 + *Node-ID*	PDO1 from the Gateway (Tx) (Object 1800_h)
Tx-PDO2	280 + *Node-ID*	PDO2 from the Gateway (Tx) (Object 1801_h)
Tx-PDO3	380 + *Node-ID*	PDO3 from the Gateway (Tx) (Object 1802_h)
Tx-PDO4	480 + *Node-ID*	PDO4 from the Gateway (Tx) (Object 1803_h)
Rx-PDO1	200 + *Node-ID*	PDO1 to the Gateway (Rx) (Object 1400_h)
Rx-PDO2	300 + *Node-ID*	PDO2 to the Gateway (Rx) (Object 1401_h)
Rx-PDO3	400 + *Node-ID*	PDO3 to the Gateway (Rx) (Object 1402_h)
Rx-PDO4	500 + *Node-ID*	PDO4 to the Gateway (Rx) (Object 1403_h)
Tx-SDO	580 + *Node-ID*	SDO from the Gateway (Tx)
Rx-SDO	600 + *Node-ID*	SDO to the Gateway (Rx)
Node Guarding	700 + *Node-ID*	configurable by Object $100E_h$

NodeID: Adjusted CANopen-Address [1_h...$7F_h$]

4.4.4.1 Adjustment of the COB-ID

The COB-IDs, which can be adjusted (with exception of the SYNC), are deviated from the adjustment of the Node-ID via LOCC-Pads *(chapter* **Fehler! Verweisquelle konnte nicht gefunden werden.***)*. If the COB-IDs are described by a SDO, the adjustment is valid, even if another Node-ID in LOCC-Pads is adjusted. In order to allow the takeover of the Default-COB-ID, the COM defaults or all defaults must be loaded (Object 1011h).

4.4.5 Setting and reading the LOCC-Box-Net

4.4.5.1 LOCC-Box-Net Message

The transmission modes can be distinguished into:

- acyclic, synchronic: The transmission is done after receiving a SYNC-Message (PDO –transmission mode 0), if the data have changed
- acyclic, synchronic: The transmission is done after receiving a certain number of SYNC-Messages (PDO-transmission mode 1...240).
- synchronic, remote request: The status of the inputs is saved with each SYNC message and is sent after receiving an RTR frame (PDO transmission mode 252).
- asynchronous, remote request: After receiving an RTR frame, the last calculated status of the inputs is sent (PDO transmission mode 253).
- process controlled, asynchronous: The transmission takes place when the status of defined LOCC Box-Net modules (PDO transmission modes 254, 255) changes.

4.4.5.2 Switching on the LOCC-Box-Net

The LOCC-Box-Net modules are sent as soon as an object to switch on is received (for example Object 6200h through Rx-PDO).

4.4.5.3 Supported transmission modalities according to DS-301

Transmission type	**PDO - transmission**					**Supported from the Gateway**
	Cyclic	Acyclic	Synchronic	asynchronous	RTR	
0		X	X			X
1 … 240	X		X			X
241 … 251	reserved					-
252			X		X	X
253				X	X	X
254				X	X	X
255				X	X	X

4.4.6 Implemented PDO's

The gateway supports TX- and RX-PDO's in the activated ***Operation Mode***. *See chapter* **Fehler! Verweisquelle konnte nicht gefunden werden.**.

4.4.6.1 Tx – PDO

The transmit PDO sends the information which module is switched on and which is switched off to the Gateway. Following identifier are used:

- 180 + *Node-ID,* for node number 1 – 64
- 280 + *Node-ID,* for node number 65 – 128
- 380 + *Node-ID,* for node number 129 – 192
- 480 + *Node-ID,* for node number 193 – 254

At each change of the status a PDO will be send. Changes of status are:

- Modules were switched on or off
- Modules are blown
- Modules were acknowledged
- The output current is bigger than 90% of the selected current range
- The output current is bigger than 100% of the selected current range

Example:

A status change was recognized on a module with the node number 1-64. The gateway with the node number 5 sends a PDO on the identifier 185_h (180 + NodeID) to the PLC.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			LSB	…	…	…	…	…	…	MSB
185_h	0_h	8_h	FF_h	00_h	$F9_h$	$0A_h$	$9B_h$	01_h	30_h	00_h
			read out value: 00 30 01 9B 0A F9 00 FF_h							

8 Data bytes are transmitted and evaluated binary. 0=OFF, 1=ON.

Binary: 0000 0000 0011 0000 0000 0001 1001 1011 0000 1010 1111 1001 0000 0000 1111 1111

Module: 64 … 57 56 … 49 48 … 41 40 … 33 32 … 25 24 … 17 16 … 9 8 … 1

4.4.6.2 Rx – PDO

Because of the receive PDO the gateway can receive information which module/node should be switched on/off. Following identifier are used:

- 200 + *NodeID,* for node number 1 – 64
- 300 + *NodeID,* for node number 65 – 128
- 400 + *NodeID,* for node number 129 – 192
- 500 + *NodeID,* for node number 193 – 254

Example:
The Gateway module 5 receives the information for node number 1-64. 8 data bytes are transmitted.
0 = OFF, 1 = ON

Binary: 0000 0000 0011 0000 0000 0001 1001 1011 0000 1010 1111 1001 0000 0000 1111 1111

Module: 64 … 57 56 … 49 48 … 41 40 … 33 32 … 25 24 … 17 16 … 9 8 … 1

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			LSB	…	…	…	…	…	…	MSB
205_h	0_h	8_h	FF_h	00_h	$F9_h$	$0A_h$	$9B_h$	01_h	30_h	00_h
			written value: 00 30 01 9B 0A F9 00 FF_h							

NOTICE

Blown modules will be not acknowledged if switching off. The command ***Switching On*** leaps the acknowledgment and switches the module on.

4.4.7 Implemented CANopen – Objects (1000_h … $1FFF_h$)

A detailed description of the objects can found in the CiA DS-301.

Index	Sub-Index	Description	Data type	R/W	Default-value	Example in Chapter
1000_h	-	Device type	unsigned 32	ro	00030191_h	4.4.7.1
1001_h	-	Error register	unsigned 8	ro	00_h	4.4.7.2
1003_h	16	Error-field	unsigned 32	ro	00_h	4.4.7.3
1005_h	-	COB-ID-Sync	unsigned 32	rw	80_h	4.4.7.4
1006_h	-	Communication cycle period	unsigned 32	ro	0000000_h	4.4.7.5
1008_h	-	Device name	visible string	ro	LOCC-Box-Net	4.4.7.6
1009_h	-	Hardware version	visible string	ro	x.y	4.4.7.7
$100A_h$	-	Software version	visible string	ro	x.y	4.4.7.8
$100C_h$	-	Guard time	unsigned 16	rw	0000_h	4.4.7.9
$100D_h$	-	Life time factor	unsigned 8	rw	00_h	4.4.7.9
$100E_h$	-	Node guard COB-ID	unsigned 32	rw	700_h + Node-ID	4.4.7.10

1010_h	1	Store parameters	unsigned 32	rw	1_h	4.4.7.11
1011_h	1	Restore default parameters	unsigned 32	rw	1_h	4.4.7.12
1014_h	-	COB-ID emergency message	unsigned 32	rw	80_h + Node-ID	4.4.7.13
1015_h	-	Inhibit time emergency	unsigned 16	rw	00_h	4.4.7.14
1016_h	1	Consumer heartbeat time	unsigned 32	rw	00_h	4.4.7.15
1017_h	-	Producer heartbeat time	unsigned 16	rw	00_h	4.4.7.16
1018_h	4	Identity object	unsigned 32	ro	-	4.4.7.18
	1	Vendor –ID			64_h	
	2	Product code, decimal			716459	
	3	Revision number			x.y	
	4	Serial number			0_h	
1020_h	2	Verify configuration	unsigned 32	rw	00_h	4.4.7.18
1029_h	1	Error behavior	unsigned 8	rw	00_h	4.4.7.19

Index	Sub-Index	Description	Data type	R/W	Example in Chapter
1400_h	2	1. Receive PDO-Parameter	PDO CommPar	rw	4.4.7.20
1401_h	2	2. Receive PDO-Parameter	PDO CommPar	rw	4.4.7.20
1402_h	2	3. Receive PDO-Parameter	PDO CommPar	rw	4.4.7.20
1403_h	2	4. Receive PDO-Parameter	PDO CommPar	rw	4.4.7.20
1600_h	8	1. Receive PDO-Mapping	PDO Mapping	ro	4.4.7.21
1601_h	8	2. Receive PDO-Mapping	PDO Mapping	ro	4.4.7.21
1602_h	8	3. Receive PDO-Mapping	PDO Mapping	ro	4.4.7.21
1603_h	8	4. Receive PDO-Mapping	PDO Mapping	ro	4.4.7.21
1800_h	5	1. Transmit PDO-Parameter	PDO CommPar	rw	4.4.7.22
1801_h	5	2. Transmit PDO-Parameter	PDO CommPar	rw	4.4.7.22
1802_h	5	3. Transmit PDO-Parameter	PDO CommPar	rw	4.4.7.22
1803_h	5	4. Transmit PDO-Parameter	PDO CommPar	rw	4.4.7.22
$1A00_h$	8	1. Transmit PDO-Mapping	PDO Mapping	ro	4.4.7.23
$1A01_h$	8	2. Transmit PDO-Mapping	PDO Mapping	ro	4.4.7.23
$1A02_h$	8	3. Transmit PDO-Mapping	PDO Mapping	ro	4.4.7.23
$1A03_h$	8	4. Transmit PDO-Mapping	PDO Mapping	ro	4.4.7.23

ro = only read
rw = read and write

4.4.7.1 Device Type (1000_h)

Index	Name	Data type	R/W	Default-value
1000_h	*Device type*	Unsigned 32	ro	0003 0191_h

The value of the *Device type is:* 0003.0191_h

Digital input/output: 0003_h
Digital profile number: 0191_h

Read-out of the device type

The CANopen Master sends with the identifier ***605_h*** (600_h + Node-ID) the read request to the Gateway (module No. 5) (Node-ID=5_h):

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
605_h	0_h	8_h	40_h	00_h	10_h	00_h	00_h	00_h	00_h	00_h
			Read Request	Index=1000_h						

The Gateway (module No. 5) sends a read response (43_h = 4 data bytes) to the master via the identifier ***585_h*** (580_h + Node-ID) with the value of the device Type:

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
585_h	0_h	8_h	43_h	00_h	10_h	00_h	91_h	01_h	03_h	00_h
			Read Response	Index=1000_h			read out value: 00 03 01 91_h			

The read out value matches the device type: 0003.0191_h (*see above*).

4.4.7.2 Error Register (1001_h)

Index	Name	Data type	R/W	Default-value
1001_h	*Error Register*	Unsigned 8	ro	00_h

The Gateway uses the error register to display error messages. At the moment the following bits of the error registers are supported:

Bit	Meaning
0	*Generic*
1	-
2	-
3	-
4	*communication error (overrun, error state)*
5	-
6	-
7	-

The bits, which are not supported (-), are always restored as ***0***. In case of an error, the suitable error bit is on ***1***.

The following error message is supported:

11_h communication error

4.4.7.3 Pre-defined Error Field (1003_h)

Index	Name	Data type	R/W	Default-value
1003_h	*Pre-defined error field*	unsigned 32	ro	00_h

In the ***predefined error field*** a list with the last error is saved. The ***sub index 0*** contains the current number of errors saved in the list. In ***sub index 1*** the last occurred failure is stored. In case a new error, the previous failure is stored in ***sub index 2*** and the new error ***sub index 1*** and so on.
In this way a failure history list can be generated.
The error memory has a ring buffer structure. If the buffer is full, the oldest entry is deleted to clear space for the present entry.
This module supports maximum 10 error entries. If the 11th failure occurs, the oldest entry is deleted.
To delete the complete failure list, the ***sub index 0*** has to be set to ***0***. This is the only allowed writing access on that object.
With every new list entry, the module sends an emergency frame to advise the error.

Index	Sub-Index [Dec]	Description	Value range [Hex]	Default value [Dec]	Data type	R/W
1003_h	0	*no_of_errors_in_list*	0, 1…10	-	unsigned 8	rw
	1	*error-code n*	0…FFFFFFFF	-	unsigned 32	ro
	2	*error-code n-1*	0…FFFFFFFF	-	unsigned 32	ro
	:	:	:	:	:	ro
	16	*error-code n-16*	0…FFFFFFFF	-	unsigned 32	ro

Meaning of the variables

no_of_errors_in_list: - contains the current number of the error codes in the list,
n = number of failure which has occurred at last
- to delete the failure list, this variable has to be set on ***0***
- if there is no_of_errors_in_list …0, the error-register (Object 1001_h) is set

error-code x: The error code with 32-Bit length consists of the CANopen-Emergency-Error-Code (see table 21, DS 301) and of the failure codes which are defined by Lütze (Manufacturer-Specific Error Field).

Bit	31 … … 16	15 … … 0
Contents	*manufacturer-specific error field*	*emergency-error-code*

manufacturer-specific error field: always ***00***

emergency-error-code: The following error codes are supported:

8001_h – No SYNC received
8120_h - CAN in Error Passive Mode
8130_h - Lifeguard Error
8140_h - Recovered from “Bus Off”

Emergency Message The data of the emergency frame sent from the Gateway are configured as follows:

Byte	0	1	2	3	4	5	6	7
Contents	*emergency-error-code*		*error-register* 1001_h	*no_of_errors_in_list* $1003,00_h$				

4.4.7.4 COB-ID of SYNC-Message (1005_h)

Index	Name	Data type	R/W	Default value
1005_h	*COB-ID of SYNC-Message*	unsigned 32	rw	80_h

Structure of the parameter:

Bit-No.	Value	Meaning
31 (MSB)	-	do not care
30	0/1	0: no SYNC generation, 1: module generates SYNC
29	0	always 0, since 11-Bit-ID
28…11	0	always 0, since no 29-Bit-ID can be supported
10…0 (LSB)	x	Bit 0...10 of the SYNC-COB-ID

The identifier can accept values between 0...$7FF_h$.

4.4.7.5 Communication Cycle Period (1006_h)

Index	Name	Data type	R/W	Default value
1006_h	*Communication Cycle Period*	unsigned 32	rw	80_h

The ***Communication Cycle Period*** determines the time interval, in which the SYNC-Telegrams are sent (if in the object 1005_h the SYNC generation is switched on) or the SYNC telegrams are expected.

4.4.7.6 Device Name (1008_h)

Index	Name	Data type	R/W	Default value
1008_h	*Device Name*	visible string	ro	LOCC-Box-Net

A detailed description of the domain upload is available on the CiA DS-202-2 (CMS-Protocol Specification).

4.4.7.7 Hardware Version (1009_h)

Index	Name	Data type	R/W	Default value
1009_h	*Hardware Version*	visible string	ro	x.y

The reading of the Hardware version is similar to the reading of the manufacturer device name by means of the domain upload protocol. A detailed description of the upload is available on the CiA DS-202-2 (CMS-Protocol Specification).

4.4.7.8 Software Version ($100A_h$)

Index	Name	Data type	R/W	Default value
$100A_h$	*Software Version*	visible string	ro	x.y

The reading of the software version is similar to the reading of the manufacturer device name via the domain upload protocol. A detailed description of the upload is available on the CiA DS-202-2 (CMS-Protocol Specification).

4.4.7.9 Guard time ($100C_h$) und Life time factor ($100D_h$)

The LOCC-Box supports the Node-Guarding or alternative the Heartbeat function. Guard Time and Life Time Factor are evaluated together. The multiplication of both values results the Life Time. The Guard Time is indicated in milliseconds.

Index	Name	Data type	R/W	Default value
$100C_h$	*Guard time*	unsigned 16	rw	0000_h
$100D_h$	*Life time factor*	unsigned 8	rw	00_h

4.4.7.10 Node Guard COB-ID ($100E_h$)

The module supports only 11-Bit-Identifier.

Index	Name	Data type	R/W	Default value
$100E_h$	*Node guard COB-ID*	unsigned 32	rw	Node-ID + 700_h

Structure of the parameter ***Node Guarding Identifier**:*

Bit-No.	Meaning
31 (MSB) 30	reserved
29…11	always 0, since no 29-Bit-ID are supported
10…0 (LSB)	Bit 0...10 of the Node Guarding Identifier

The identifier can accept values between 1...$7FF_h$.

4.4.7.11 Store Parameters (1010_h)

Index	Name	Data type	R/W	Default value
1010_h	*Store Parameters*	unsigned 32	rw	1_h

The command parameters are saved in the EEPROM.

During the writing of the index, the following byte sequence must be sent. The reading of the index returns the information about the implemented save functions (for full particulars see CiA DS-301).

Index	Sub-Index	Description	Value range [Hex]	Data type	R/W
1010_h	0	*number_ of_entries*	4_h	unsigned 8	ro
	1	*save_all_parameters* (Objects 1000_h…$9FFF_h$)	no default, write: $65\ 76\ 61\ 73_h$ (=ASCII: ‘e‘ ‘v‘ ‘a‘ ‘s‘)	unsigned 32	rw

The following parameters can be saved or loaded:

- Communication Parameter of the objects 1005_h ... 1029_h
- Application Parameter of the objects 6000_h ... 6100_h
- Manufacturer Specific Parameter of the objects 2220_h ... 2310_h

4.4.7.12 Restore Default Parameters (1011_h)

Index	Name	Data type	R/W	Default value
1011_h	*Restore default parameters*	unsigned 32	rw	1_h

With this command it is possible to activate the default parameters again, which were active at the moment of the delivery. All individual adjustments, which were saved in the EEPROM will get lost. Only the instruction ***Restore all Parameters*** is supported. During the writing of the index, the sequence, described below, must be sent. The reading of the index returns the information about the implemented restore functions (for further information read CiA DS-301).

Index	Sub-Index	Description	Value range [Hex]	Data type	R/W
1011_h	0	*number_ of_entries*	4_h	unsigned 8	ro
	1	*load_all_default_parameters* (Objects 1000_h…$9FFF_h$)	no default, write: 64 61 6F $6C_h$ (=ASCII: ‘d‘ ‘a‘ ‘o‘ ‘l‘)	unsigned 32	rw

The following parameters can be saved or loaded:

- Communication Parameter of the objects 1005_h ... 1029_h
- Application Parameter of the objects 6000_h ... 6100_h
- Manufacturer Specific Parameter of the objects 2220_h ... 2310_h

4.4.7.13 COB_ID Emergency Message (1014_h)

Index	Name	Data type	R/W	Default value
1014_h	*COB_ID emergency message*	unsigned 32	rw	80_h + Node-ID

This object determines the COB-ID of the Emergency Message (EMCY).

The structure of the objects is described in the table below.

Bit-No.	Value	Meaning
31 (MSB)	0/1	0: EMCY exists / is valid, 1: no EMCY / EMCY is not valid
30	0	reserved (always 0)
29	0	always 0, since 11-Bit ID
28…11	0	always 0, since no 29-Bit-ID can be supported
10…0 (LSB)	x	Bit 0...10 des COB-ID

The identifier can accept values between 0...$7FF_h$.

4.4.7.14 Inhibit Time Emergency (1015_h)

Index	Name	Data type	R/W	Default value
1015_h	*Inhibit time emergency*	unsigned 16	rw	00_h

With this object it is possible to set the Inhibit Time for the emergency message.
The time is indicated as a multiple of 100µs.

4.4.7.15 Consumer Heartbeat Time (1016_h)

Index	Name	Data type	R/W	Default value
1016_h	*Consumer Heartbeat Time*	unsigned 32	rw	00_h

In order to get a reciprocal monitoring of the CANopen modules (in particular to recognize connection breakdown) it is possible to use the heartbeat function. In contrast to the Node Guarding/Life Guarding the Heartbeat function can work without RTR frames.
One module, called Heartbeat Producer, sends a heartbeat message via the CAN-Bus to the Node-Guarding-Identifier (see Object$100E_h$) in a cyclic way. One or more Heartbeat-Consumers receive the message. The message has to be received within the Heartbeat-Time, otherwise a heartbeat event is triggered on the heartbeat-consumer. On the Gateway the Heartbeat-Event triggers a Heartbeat-Error. Every module can be Heartbeat-Producer and Heartbeat-Consumer. The Gateway supports in a CAN-net one Heartbeat-Producer maximum.

Index	Sub-Index [Dec]	Description	Value range [Hex]	Default value [Dec]	Data type	R/W
1016_h	0	*number_ of_entries*	1	1	unsigned 8	ro
	1	*Consumer-heartbeat_time*	0 … 007FFFFF	0	unsigned 32	rw

Meaning of the variables:

consumer-heartbeat_time_x

	consumer-heartbeat_time_x		
Bit	31 … … 24	23 … … 16	15 … … 0
Assignment	reserved (always “0”)	Node-ID (unsigned 8)	Heartbeat_time (unsigned 16)

Node-ID Node-ID of the Heartbeat-Producer to be monitored

Heartbeat_time Time in which the monitored Heartbeat-Producer has to respond to the Node. Guarding-ID to avoid the triggering of a heartbeat event.
This Consumer-Heartbeat-Time must be always higher than the Producer-Heartbeat-Time.

4.4.7.16 Producer Heartbeat Time (1017_h)

Index	Name	Data type	R/W	Default value
1017_h	*Producer heartbeat time*	unsigned 16	rw	00_h

Enter the time in which the Gateway should send a heartbeat frame on the node guarding ID in a cyclic way. If a producer heartbeat time bigger than 0 is set, the heartbeat time will be active and the Node/Lifeguarding will stop.
If the Producer-Heartbeat time is set to ***0*** the heartbeat will stop.

Index	Sub-Index [Dec]	Description	Value range [Hex]	Default value [Dec]	Data type	R/W
1017_h	0	*Producer heartbeat time*	0 … FFFF	0 ms	unsigned 16	rw

producer-heartbeat_time Cyclic time of the Heartbeat-Producer to send the Heartbeat to the Node-Guarding-ID (see Object 100Eh). The Consumer-Heartbeat-Time of the module to be monitored must always be bigger than the Producer-Heartbeat-Time of the module which sends heartbeats.

4.4.7.17 Identity Object (1018_h)

Index	Name	Data type	R/W	Default value
1018_h	*Identity object*	unsigned 32	ro	-

This object includes some general information about the Gateway.

Index	Sub-Index [Dec]	Description	Value range [Hex]	Default value	Data type	R/W
1018_h	0	*no_of_entries*	4	4	unsigned 8	ro
	1	*Vendor_ID*	0…FFFFFFFF	0000 0064_h	unsigned 32	const
	2	*product_code*	0…FFFFFFFF	716459_d	unsigned 32	const
	3	*revision_number*	0…FFFFFFFF	x.y	unsigned 32	ro
	4	*serial_number*	0…FFFFFFFF	0	unsigned 32	ro

Meaning of the variables

Vendor_id This variable includes the Vendor-ID. The value is always 00000064_h .

product_code The part number of the product is stored. The Hex.-number corresponds to the decimal part number.

revision_number The Software version is stored. In the two head bytes numbers of the major changes are mentioned according to DS-301. In the two bytes below you can find the revision numbers of single modifications or changes (minor).

4.4.7.18 Verify Configuration (1020_h)

Index	Name	Data type	R/W	Default value
1020_h	*Verify configuration*	unsigned 32	rw	00_h

With this object it is possible to store data's and times of the last configuration in order to verify it a second time if the saved configuration matches the expected one.

Index	Sub-Index [Dec]	Description	Value range [Hex]	Default value [Dec]	Data type	R/W
1020_h	0	*no_of_entries*	2	2	unsigned 8	ro
	1	*configuration_date*	0…FFFFFFFF	0	unsigned 32	rw
	2	*configuration_time*	0…FFFFFFFF	0	unsigned 32	rw

Meaning of the variables

configuration_date Date of the last configuration of the modules, indicated in days since 01.01.1984.

configuration_time Time in ms starting from midnight of the day of the last configuration.

4.4.7.19 Error Behavior Object (1029_h)

Index	Name	Data type	R/W	Default value
1029_h	*Error behavior object*	unsigned 8	rw	00_h

In case an error event occurs (for example Heartbeat-Error), the module will be in the mode which is defined in the variable communication_error or output_error.

Index	Sub-Index [Dec]	Description	Value range [Hex]	Default value [Dec]	Data type	R/W
1020_h	0	*no_of_error_classes*	6	6	unsigned 8	ro
	1	*Communication_error*	0…2	0	unsigned 8	rw

Meaning of the variables

no_of_error_classes *Number of the error classes (here always '6')*

Communication_error Heartbeat/Lifeguard-error and Bus off

In case an error occurs, the module switches to the suitable indicated mode.

Value of the variables	Modes, where the module switches in case of error
0	pre-operational (only if the current status = operational)
1	no state change
2	stopped

4.4.7.20 Receive PDO Communication Parameter ($140x_h$)

With the objects ***Receive PDO Communication Parameter*** the properties of the Receive PDO (Rx-PDO) are defined.

Index	Name	Data type	R/W
1400_h	*1. Receive PDO Parameter*	PDO CommPar	rw
1401_h	*2. Receive PDO Parameter*	PDO CommPar	rw
1402_h	*3. Receive PDO Parameter*	PDO CommPar	rw
1403_h	*4. Receive PDO Parameter*	PDO CommPar	rw

Index	Sub-Index [Dec]	Description	Value range [Hex]	Default	Data type	R/W
1400_h	0	*no_of_entries*	2	2	unsigned 8	ro
	1	*COB-ID used by PDO1*	1…800007FF	200_h+Node-ID	unsigned 32	rw
	2	*transmission type*	0…FF	255_d	unsigned 8	rw
1401_h	0	*no_of_entries*	2	2	unsigned 8	ro
	1	*COB-ID used by PDO2*	1…800007FF	300_h+Node-ID	unsigned 32	rw
	2	*transmission type*	0…FF	255_d	unsigned 8	rw
1402_h	0	*no_of_entries*	2	2	unsigned 8	ro
	1	*COB-ID used by PDO3*	1…800007FF	400_h+Node-ID	unsigned 32	rw
	2	*transmission type*	0…FF	255_d	unsigned 8	rw
1403_h	0	*no_of_entries*	2	2	unsigned 8	ro
	1	*COB-ID used by PDO4*	1…800007FF	500_h+Node-ID	unsigned 32	rw
	2	*transmission type*	0…FF	255_d	unsigned 8	rw

All transmission types 0-240, 254 and 255 are supported.

4.4.7.21 Receive PDO Mapping Parameter ($160x_h$)

With this object ***Receive PDO Mapping Parameter*** the allocation of the receive data to the Rx-PDOs is defined.

Index	Name	Data type	R/W
1600_h	*1. Receive PDO Mapping*	PDO Mapping	ro
1601_h	*2. Receive PDO Mapping*	PDO Mapping	ro
1602_h	*3. Receive PDO Mapping*	PDO Mapping	ro
1603_h	*4. Receive PDO Mapping*	PDO Mapping	ro

The following table shows the definition of the Receive PDO Mapping Parameter for the default configuration:

Index	Sub-Index [Dec]	Description	Default [Hex]	Data type	R/W
1600_h	0	*no_of_entries*	1	unsigned 8	ro
	1	*1st_application_object*	62000108	unsigned 32	ro
	2	*2nd_application_object*	62000208	unsigned 32	ro
	3	*3rd_application_object*	62000308	unsigned 32	ro
	4	*4th_application_object*	62000408	unsigned 32	ro
	5	*5th_application_object*	62000508	unsigned 32	ro
	6	*6th_application_object*	62000608	unsigned 32	ro
	7	*7th_application_object*	62000708	unsigned 32	ro
	8	*8th_application_object*	62000808	unsigned 32	ro
1601_h	0	*no_of_entries*	1	unsigned 8	ro
	1	*1st_application_object*	62000908	unsigned 32	ro
	2	*2nd_application_object*	62000A08	unsigned 32	ro
	3	*3rd_application_object*	62000B08	unsigned 32	ro
	4	*4th_application_object*	62000C08	unsigned 32	ro
	5	*5th_application_object*	62000D08	unsigned 32	ro
	6	*6th_application_object*	62000E08	unsigned 32	ro
	7	*7th_application_object*	62000F08	unsigned 32	ro
	8	*8th_application_object*	62001008	unsigned 32	ro
1602_h	0	*no_of_entries*	1	unsigned 8	ro
	1	*1st_application_object*	62001108	unsigned 32	ro
	2	*2nd_application_object*	62001208	unsigned 32	ro
	3	*3rd_application_object*	62001308	unsigned 32	ro
	4	*4th_application_object*	62001408	unsigned 32	ro
	5	*5th_application_object*	62001508	unsigned 32	ro
	6	*6th_application_object*	62001608	unsigned 32	ro
	7	*7th_application_object*	62001708	unsigned 32	ro
	8	*8th_application_object*	62001808	unsigned 32	ro
1603_h	0	*no_of_entries*	1	unsigned 8	ro
	1	*1st_application_object*	62001908	unsigned 32	ro
	2	*2nd_application_object*	62001A08	unsigned 32	ro
	3	*3rd_application_object*	62001B08	unsigned 32	ro
	4	*4th_application_object*	62001C08	unsigned 32	ro
	5	*5th_application_object*	62001D08	unsigned 32	ro
	6	*6th_application_object*	62001E08	unsigned 32	ro
	7	*7th_application_object*	62001F08	unsigned 32	ro
	8	*8th_application_object*	62002008	unsigned 32	ro

4.4.7.22 Transmit PDO Communication Parameter (180x$_h$)

With this object, the properties of a transmit-PDO1 are defined.

Index	Name	Data type	R/W
1800$_h$	*1. Transmit PDO Parameter*	PDO CommPar	ro
1801$_h$	*2. Transmit PDO Parameter*	PDO CommPar	ro
1802$_h$	*3. Transmit PDO Parameter*	PDO CommPar	ro
1803$_h$	*4. Transmit PDO Parameter*	PDO CommPar	ro

Index	Sub-Index [Dec]	Description	Value range [Hex]	Default [Hex]	Data type	R/W
1800$_h$	0	*no_of_entries*	0…FF	5	unsigned 8	ro
	1	*COB-ID used by PDO1*	1…800007FF	180$_h$+Node-ID	unsigned 32	rw
	2	*transmission type*	0…FF	255$_d$	unsigned 8	rw
	3	*inhibit time*	0…FFFF	0	unsigned 16	rw
	4	*reserved*	0…FF	0	unsigned 8	const
	5	*event timer*	0…FFFF	0	unsigned 16	rw
1801$_h$	0	*no_of_entries*	0…FF	5	unsigned 8	ro
	1	*COB-ID used by PDO2*	1…800007FF	280$_h$+Node-ID	unsigned 32	rw
	2	*transmission type*	0…FF	255$_d$	unsigned 8	rw
	3	*inhibit time*	0…FFFF	0	unsigned 16	rw
	4	*reserved*	0…FF	0	unsigned 8	const
	5	*event timer*	0…FFFF	0	unsigned 16	rw
1802$_h$	0	*no_of_entries*	0…FF	5	unsigned 8	ro
	1	*COB-ID used by PDO3*	1…800007FF	380$_h$+Node-ID	unsigned 32	rw
	2	*transmission type*	0…FF	255$_d$	unsigned 8	rw
	3	*inhibit time*	0…FFFF	0	unsigned 16	rw
	4	*reserved*	0…FF	0	unsigned 8	const
	5	*event timer*	0…FFFF	0	unsigned 16	rw
1803$_h$	0	*no_of_entries*	0…FF	5	unsigned 8	ro
	1	*COB-ID used by PDO4*	1…800007FF	480$_h$+Node-ID	unsigned 32	rw
	2	*transmission type*	0…FF	255$_d$	unsigned 8	rw
	3	*inhibit time*	0…FFFF	0	unsigned 16	rw
	4	*reserved*	0…FF	0	unsigned 8	const
	5	*event timer*	0…FFFF	0	unsigned 16	rw

The transmission types 0, 1-240, 252, 253, 254 and 255 are supported.

4.4.7.23 Transmit PDO Mapping Parameter (1A0x$_h$)

With the object ***Transmit PDO Mapping Parameter*** the allocation of the transmit data to the Tx-PDOs is defined.

Index	Name	Data type	R/W
1A00$_h$	*1. Receive PDO Mapping*	PDO Mapping	ro
1A01$_h$	*2. Receive PDO Mapping*	PDO Mapping	ro
1A02$_h$	*3. Receive PDO Mapping*	PDO Mapping	ro
1A03$_h$	*4. Receive PDO Mapping*	PDO Mapping	ro

The following table shows the configuration of the Transmit PDO Mapping parameter:

Index	Sub-Index	Description	Default [Hex]	Data type	R/W
1A00$_h$	0	*no_of_entries*	4	unsigned 32	ro
	1	*object_to_be_mapped_1*	60000108	unsigned 32	ro
	2	*object_to_be_mapped_2*	60000208	unsigned 32	ro
	3	*object_to_be_mapped_3*	60000308	unsigned 32	ro
	4	*object_to_be_mapped_4*	60000408	unsigned 32	ro
	5	*object_to_be_mapped_5*	60000508	unsigned 32	ro
	6	*object_to_be_mapped_6*	60000608	unsigned 32	ro
	7	*object_to_be_mapped_7*	60000708	unsigned 32	ro
	8	*object_to_be_mapped_8*	60000808	unsigned 32	ro
1A01$_h$	0	*no_of_entries*	4	unsigned 32	ro
	1	*object_to_be_mapped_1*	60000908	unsigned 32	ro
	2	*object_to_be_mapped_2*	60000A08	unsigned 32	ro
	3	*object_to_be_mapped_3*	60000B08	unsigned 32	ro
	4	*object_to_be_mapped_4*	60000C08	unsigned 32	ro
	5	*object_to_be_mapped_5*	60000D08	unsigned 32	ro
	6	*object_to_be_mapped_6*	60000E08	unsigned 32	ro
	7	*object_to_be_mapped_7*	60000F08	unsigned 32	ro
	8	*object_to_be_mapped_8*	60001008	unsigned 32	ro
1A02$_h$	0	*no_of_entries*	4	unsigned 32	ro
	1	*object_to_be_mapped_1*	60001108	unsigned 32	ro
	2	*object_to_be_mapped_2*	60001208	unsigned 32	ro
	3	*object_to_be_mapped_3*	60001308	unsigned 32	ro
	4	*object_to_be_mapped_4*	60001408	unsigned 32	ro
	5	*object_to_be_mapped_5*	60001508	unsigned 32	ro
	6	*object_to_be_mapped_6*	60001608	unsigned 32	ro
	7	*object_to_be_mapped_7*	60001708	unsigned 32	ro
	8	*object_to_be_mapped_8*	60001808	unsigned 32	ro
1A03$_h$	0	*no_of_entries*	4	unsigned 32	ro
	1	*object_to_be_mapped_1*	60001908	unsigned 32	ro
	2	*object_to_be_mapped_2*	60001A08	unsigned 32	ro
	3	*object_to_be_mapped_3*	60001B08	unsigned 32	ro
	4	*object_to_be_mapped_4*	60001C08	unsigned 32	ro
	5	*object_to_be_mapped_5*	60001D08	unsigned 32	ro
	6	*object_to_be_mapped_6*	60001E08	unsigned 32	ro
	7	*object_to_be_mapped_7*	60001F08	unsigned 32	ro
	8	*object_to_be_mapped_8*	60002008	unsigned 32	ro

4.4.8 Device Profile Area (6000_h … $9FFF_h$)

Index	Sub-Index	Description	Data type	R/W	Example in Chapter
6000_h	32	Module status On / OFF, (Byte) Sub-Index 01h = Nodes 1-8, Sub-Index 02h = Nodes 9-16 … Sub-Index 20h = Nodes 249-254 Binary 0 = Off / Blown / Ext.-Off Binary 1 = On	uint 8	ro	4.4.8.1
6020_h	128	Module status On / Off, (Bit) Sub-Index 01h = Node 1, Sub-Index 02h = Node 2 … Sub-Index 80h = Node 128	Bool	ro	4.4.8.2
6021_h	126	Module status On / Off, (Bit) Sub-Index 01h = Node 129, Sub-Index 02h = Node 130 … Sub-Index 7Eh = Node 254	Bool	ro	4.4.8.3
6200_h	32	Modules switch On / Off, (Byte) Sub-Index 01h = Nodes 1-8, Sub-Index 02h = Nodes 9-16 … Sub-Index 20h = Nodes 249-254 Binary 0 = Off / Blown / Ext.-Off Binary 1 = On	unit 8	rw	4.4.8.4
6220_h	128	Modules switch On / Off, (Bit) Sub-Index 01h = Node 1, Sub-Index 02h = Node 2 … Sub-Index 80h = Node 128	Bool	rw	4.4.8.5
6221_h	126	Modules switch On / Off, (Bit) Sub-Index 01h = Node 129, Sub-Index 02h = Node 130 … Sub-Index 7Eh = Node 254	Bool	rw	4.4.8.6

ro = only read
rw = read, write

4.4.8.1 Module status On / Off (6000_h)

Index	Name	Data type	R/W
6000_h	*Module status On / Off - Byte*	uint 8	ro

The read value writes back, byte by byte the module status ***ON/OFF*** of the connected LOCC-Box-Net modules.

Read out of the module status

The CANopen master sends with the Identifier '605_h' (600_h + Node-ID) the read request to the Gateway No. 5 (Node-ID=5_h).

→ **Sub-Index:** The Sub-Index is valid ascending for each 8 Bit group (1 Byte).
LOCC-Box-Net node numbers 1 – 8 = 01_h
LOCC-Box-Net node numbers 9 – 16 = 02_h
LOCC-Box-Net node numbers 17 – 24 = 03_h
: :
LOCC-Box-Net node numbers 241 – 248 = $1F_h$
LOCC-Box-Net node numbers 249 – 254 = 20_h

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
605_h	0_h	8_h	40_h	00_h	60_h	01_h	00_h	00_h	00_h	00_h
			Read Request	Index=6000_h		Modules / node 1-8				

The Gateway module No. 5 answers via the read response ($4F_h$ = 1 Data byte) with the Identifier ***585_h*** (580_h + Node-ID) and the status of the modules (node numbers 1 – 8).

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
585_h	0_h	8_h	$4F_h$	00_h	60_h	01_h	03_h	00_h	00_h	00_h
			Read Response	Index=6000_h		Modules / node 1-8	read out value: 00 00 00 03_h			

The read out value is converted into a binary value.

Example 1:
Data byte 03_h = binary: 0 0 0 0 0 0 1 1
Sub-Index 01_h = node number: 8 7 6 5 4 3 2 1
→ Modules 1 and 2 are ON
Modules 3 - 8 are OFF

Example 2:
Data byte 20_h = binary: X X 1 0 0 0 0 0
Sub-Index 20_h = node number: 256 255 254 253 252 251 250 249
→ Module 254 is ON
Modules 249–253 are OFF

Remark: For each Gateway it is possible to manage maximum 254 LOCC-Box-Net modules.

4.4.8.2 Module status On / Off (6020_h)

Index	Name	Data type	R/W
6020_h	*Module status On / Off – Bit, nodes 1 - 128*	uint 8	ro

The value, which has been read, returns the *module status* ***ON/OFF*** of the selected module.

Read out of the module status

The CANopen master sends with the identifier ***605_h*** (600_h + Node-ID) the read request to the Gateway No.5 (Node-ID=5_h).

→ **Sub-Index:** The Sub-Index is valid ascending for the secondary modules 1 – 128.
LOCC-Box-Net node number 1 = 01_h
LOCC-Box-Net node number 2 = 02_h
LOCC-Box-Net node number 3 = 03_h
: :
LOCC-Box-Net node number 127 = $7F_h$
LOCC-Box-Net node number 128 = 80_h
In the example the module No.8 is requested.

→ **Data byte (LSB):** The data byte has to be read as follows:
01_h = Module is switched on
00_h = Module is switched off

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
605_h	0_h	8_h	40_h	20_h	60_h	08_h	00_h	00_h	00_h	00_h
			Read Request	Index=6020_h		Modules / nodes 8				

The Gateway module No. 5 sends an answers to the master via the read response ($4F_h$ = 1 Data byte) with the identifier ***585_h*** (580_h + Node-ID) and the status of the modules with the node number 8.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
585_h	0_h	8_h	$4F_h$	20_h	60_h	08_h	01_h	00_h	00_h	00_h
			Read Response	Index=6020_h		Modules / nodes 8	read out value: 00 00 00 01_h			

Example 1:
Sub-Index = 08_h / Data byte = 01_h → Module with node number 8 is on!

Example 2:
Sub-Index = 08_h / Data byte = 00_h → Module with node number 8 is off!

4.4.8.3 Module status On / Off (6021_h)

Index	Name	Data type	R/W
6021_h	*Module status On / Off – Bit, nodes 129 - 254*	uint 8	ro

The read value shows the status *ON/OFF* of the selected module.

Read out of the module status

The CANopen master sends the read request via the identifier ***605_h*** (600_h + Node-ID) to the Gateway No. 5 (Node-ID=5_h).

→ **Sub-Index:** The Sub-Index is valid ascending for the secondary modules 129–254.
LOCC-Box-Net node number 129 = 81_h
LOCC-Box-Net node number 130 = 82_h
LOCC-Box-Net node number 131 = 83_h
: :
LOCC-Box-Net node number 253 = FD_h
LOCC-Box-Net node number 254 = FE_h
In the example the module No. 130 is requested

→ **Data byte (LSB):** The data byte has to be read as follows:
01_h = module is switched on
00_h = module is switched off

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
605_h	0_h	8_h	40_h	20_h	60_h	82_h	00_h	00_h	00_h	00_h
			Read Request	Index=6020_h		Modules / nodes 130				

The Gateway module No. 5 sends an answers to the master via the read response ($4F_h$ = 1 Data byte) with the identifier ***585_h*** (580_h + Node-ID) and the status of the modules with the node number 8.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
585_h	0_h	8_h	$4F_h$	20_h	60_h	82_h	01_h	00_h	00_h	00_h
			Read Response	Index=6020_h		Modules / nodes 130	read out value: 00 00 00 01_h			

Example 1:
Sub-Index = 82_h / Data byte = 01_h → Module with the node number 130 is switched on!

Example 2:
Sub-Index = 82_h / Data byte = 00_h → Module with the node number 130 is switched off!

4.4.8.4 Switch ON and OFF modules - (6200_h)

Index	Name	Data type	R/W
6200_h	*Switch On / Off modules – Byte*	uint 8	rw

The value (the least order byte) sets byte-by-byte selected LOCC-Box-Net modules in the status ON or OFF.

Switch ON / OFF modules
The CANopen Master sends the Write Request via the identifier ***605_h*** (600_h + Node-ID) to the Gateway No.5 (Node-ID=5_h).

→ **Command code**: corresponds to $2F_h$ - write request for 1 data byte (*see chapter 4.4.3.1*)

→ **Sub-Index:** The sub index is valid ascending for each 8 bit group (1 byte).
LOCC-Box-Net node numbers 1 – 8 = 01_h
LOCC-Box-Net node numbers 9 – 16 = 02_h
LOCC-Box-Net node numbers 17 – 24 = 03_h
: :
LOCC-Box-Net node numbers 241 – 248 = $1F_h$
LOCC-Box-Net node numbers 249 – 254 = 20_h

→ **Data byte (LSB):** The data byte has to be written as follows:
1 = module is switched on
0 = module is switched off

Example 1:
Data byte 03_h corresp.to binary 0 0 0 0 0 0 1 1
Sub-Index 01_h = Module node: 8 7 6 5 4 3 2 1
→ Modules 1 and 2 = ON
Modules 3 - 8 = OFF

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
605_h	0_h	8_h	$2F_h$	00_h	62_h	01_h	03_h	00_h	00_h	00_h
			Write Request	Index=6200_h		Modules / nodes 1-8	Value to be written			

Example 2:
Data byte $0F_h$ = binary: X X 0 0 1 1 1 1
Sub-Index 20_h = Module nodes: 256 255 254 253 252 251 250 249
→ Module 249 - 252 = ON
Module 253 – 254 = OFF

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
605_h	0_h	8_h	$2F_h$	00_h	62_h	20_h	$0F_h$	00_h	00_h	00_h
			Write Request	Index=6200_h		Modules / nodes 249-254	Value to be written			

The Gateway module No. 5 sends an answers to the master via a write response (60_h) with the identifier ***585_h*** (580_h + Node-ID). The response contains no data.

4.4.8.5 Switch ON and OFF modules - (6220_h)

Index	Name	Data type	R/W
6220_h	*Module status ON / OFF – Bit, nodes 1 - 128*	uint 8	rw

The value sets the selected LOCC-Box-Net module to the status ON or OFF.

Switch ON / OFF modules

The CANopen master sends the write request via the identifier ***605_h*** (600_h + Node-ID) to the Gateway with No. 5 (Node-ID=5_h).

→ **Command code**: corresponds to $2F_h$ - write request for 1 data byte (*see section 4.4.3.1*)

→ **Sub-Index:** The sub index is valid ascending for the secondary modules 1 – 128

LOCC-Box-Net node number 1 = 01_h
LOCC-Box-Net node number 2 = 02_h
LOCC-Box-Net node number 3 = 03_h
: :
LOCC-Box-Net node number 127 = $7F_h$
LOCC-Box-Net node number 128 = 80_h

→ **Data byte (LSB):** The data byte has to be written as follows:

01_h = Module is switched on
00_h = Module is switched off

Example 1:
Sub index = 01_h / Data byte = 01_h → Module with node number 1 is switched on

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
605_h	0_h	8_h	$2F_h$	20_h	62_h	01_h	01_h	00_h	00_h	00_h
			Write Request	Index=6220_h		Module / node 1	Value to be written			

Example 2:
Sub-Index = 14_h / Data byte = 00_h → Module with the node number 20 is switched off

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
605_h	0_h	8_h	$2F_h$	20_h	62_h	14_h	00_h	00_h	00_h	00_h
			Write Request	Index=6220_h		Module / node 20	Value to be written			

The Gateway module No. 5 sends an answer to the master via a write response (60_h) with the identifier ***585_h*** (580_h + Node-ID). The response contains no data.
If an already switched module is switched on again or a module is already switched off and it is switched off again, an error response (80_h) occurs.

4.4.8.6 Switch ON and OFF modules - (6221_h)

Index	Name	Data type	R/W
6221_h	*Module status ON / OFF – Bit, nodes 129 - 254*	uint 8	rw

The value sets the selected LOCC-Box-Net module in the status ON or OFF.

Switch ON / OFF modules

The CANopen Master sends a Write Request via the identifier ***605_h*** (600_h + Node-ID) to the Gateway No.5 (Node-ID=5_h).

→ **Command code**: corresponds to $2F_h$ - write request for 1 data byte *(see chapter 4.4.3.1)*

→ **Sub-Index:** The sub index is valid ascending to the secondary modules 129 – 254_d
LOCC-Box-Net node number 129 = 01_h
LOCC-Box-Net node number 130 = 02_h
LOCC-Box-Net node number 131 = 03_h
: :
LOCC-Box-Net node number 253 = $7D_h$
LOCC-Box-Net node number 254 = $7E_h$

→ **Data byte (LSB):** The data byte has to be written as follows:
01_h = Module is switched on
00_h = Module is switched off

Example 1:
Sub-Index = 03_h / Data byte = 01_h → Module with the node number 131 is switched on

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
605_h	0_h	8_h	$2F_h$	21_h	62_h	03_h	01_h	00_h	00_h	00_h
			Write Request	Index=6221_h		Module / node 131	Value to be written			

Example 2:
Sub-Index = $7D_h$ / Data byte = 00_h → Module with the node number 253 is switched off

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
605_h	0_h	8_h	$2F_h$	21_h	62_h	$7D_h$	00_h	00_h	00_h	00_h
			Write Request	Index=6221_h		Module / node 253	Value to be written			

The Gateway module No. 5 sends an answers to the master via a write response (60_h) with the identifier ***585_h*** (580_h + Node-ID). The response contains no data.
If a module, already switched on, is switched on again or a module already switched off, is switched off again, you receive an error response (80_h).

4.4.9 Manufacturer Specific Profile Area (2000_h … $5FFF_h$)

Index	Sub-Index (max.)[Dec]	Description	Data type	R/W	Example in Chapter
2000_h	254	Module type of nodes 1 – 254	uint 8	ro	4.4.9.1
2010_h	254	Module status of nodes 1 – 254	uint 8	ro	4.4.9.2
2011_h	254	Module configuration (rotary switch)	uint 8	ro	4.4.9.3
2100-$210A_h$	254	Analogue value current / voltage	uint 16	ro	
2100_h	254	Output voltage			4.4.9.4
2101_h	254	Input voltage			4.4.9.5
2104_h	254	Current measurement			4.4.9.6
$210A_h$	254	Characteristic curves adjustment			4.4.9.7
2200-2206_h	254	Device data and counters	uint 32	ro	
2200_h	254	Software version			4.4.9.8
2201_h	254	Serial number			4.4.9.9
2202_h	254	LOCC-Box counter “operation voltage On“			4.4.9.10
2203_h	254	LOCC-Box counter “Operation hours (h)“			4.4.9.11
2204_h	254	LOCC-Box counter “Operating hours ON (h)“			4.4.9.12
2205_h	254	LOCC-Box counter “Blown“ (see page 23)			4.4.9.13
2206_h	254	LOCC-Box counter “Switch on“ (see page 24)			4.4.9.14

Sub-Index = node number of the connected LOCC-Box
ro = only read
rw = read and write

4.4.9.1 Module Type (2000_h)

Index	Name	Data type	R/W
2000_h	*Module type*	uint 8	ro

The value of *Module Type* interprets the module version: 1 = 716410

Read out of the module type

The CANopen master sends the identifier ***605_h*** (600_h + Node-ID) via the read request to the Gateway with No. 5 (Node-ID=5_h).

→ The secondary module with the node number 9 is read out.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
605_h	0_h	8_h	40_h	00_h	20_h	09_h	00_h	00_h	00_h	00_h
			Read Request	Index=2000_h		Module / node 9				

The Gateway module No. 5 sends an answers to the master via the read response ($4F_h$ = 1 Data byte) with the Identifier ***585_h*** (580_h + Node-ID) and the value of the module with node number 9.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
585_h	0_h	8_h	$4F_h$	00_h	20_h	09_h	01_h	00_h	00_h	00_h
			Read Response (1Byte)	Index=2000_h		Module / node 9	Read out value : 00 00 00 01_h			

The read out value is converted into a decimal value.

Example:

00 00 00 01_h = 1

Corresponds to the module type: 1 → 716410

4.4.9.2 Module status (2010_h)

Index	Name	Data type	R/W
2010_h	*Module status*	uint 8	ro

The read out value returns the module status. This can be:

Readout of the module status
The CANopen master sends the read request via the identifier ***605_h*** (600_h + Node-ID) to the Gateway No.5 (Node-ID=5_h).
→ The secondary module with the node number 9 is read out.

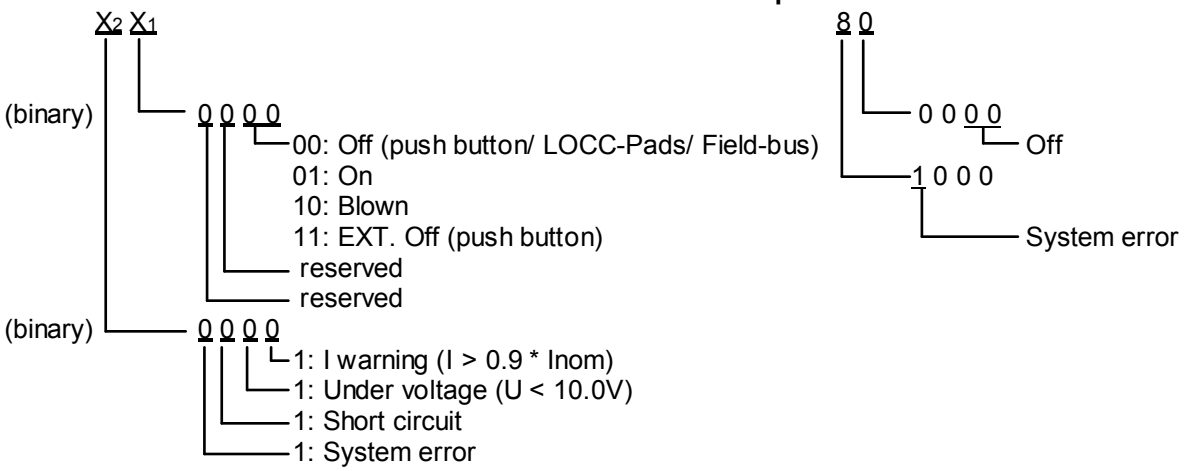
ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
605_h	0_h	8_h	40_h	10_h	20_h	09_h	00_h	00_h	00_h	00_h
			Read Request	Index=2010_h		Module / node 9				

The Gateway module No. 5 sends answers to the master via the read response ($4F_h$ = 1 Data byte) with the identifier ***585_h*** (580_h + Node-ID) and the value of the module with node number 9.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
585_h	0_h	8_h	$4F_h$	10_h	20_h	09_h	80_h	00_h	00_h	00_h
			Read Response (1Byte)	Index=2010_h		Module / node 9	Read out value : 00 00 00 80_h			

The read out value is converted into a binary value.

Example:



The value means: Module is switched off by the push LOCC Pads signalizes a system error.

4.4.9.3 Module configuration (2011_h)

Index	Name	Data type	R/W
2011_h	*Module configuration*	uint 8	ro

The value of the module configuration returns the adjustment of the current range or of the characteristic (rotary-switch).

Read out of the module configuration
The CANopen master sends the read request via the identifier ***605_h*** (600_h + Node-ID) to the Gateway No. 5 (Node-ID=5_h).
→ The secondary module with the node number 1 is read out.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
605_h	0_h	8_h	40_h	11_h	20_h	01_h	00_h	00_h	00_h	00_h
			Read Request	Index=2011_h		Module / node 1				

The Gateway module No. 5 sends an answers to the master via the read response ($4F_h$ = 1 Data byte) with the identifier ***585_h*** (580_h + Node-ID) and the value of the module with node number 1.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
585_h	0_h	8_h	$4F_h$	11_h	20_h	01_h	15_h	00_h	00_h	00_h
			Read Response (1Byte)	Index=2011_h		Module / node 1	read out value : 00 00 00 15_h			

The read out value is converted into a decimal value and added with 1.

Y_2 Y_1
└ = Decimal value 0-9 + 1 = current range 1-10A
└ = Decimal value 0-9 + 1 = characteristic 1-10

Example:

1 5
└ = decimal = 5 + 1 = current range 6A
└ = decimal = 1 + 1 = characteristic 2

4.4.9.4 Output Voltage (2100_h)

Index	Name	Data type	R/W
2100_h	*Output Voltage*	uint 16	ro

The value contains the amount of the adjacent output voltage.

Read out of the output voltage
The CANopen Master sends the read request via the identifier ***$60A_h$*** (600_h + Node-ID to the Gateway No. 10 (Node-ID=10_h).
→ The secondary module with the node number 1 is read out.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
$60A_h$	0_h	8_h	40_h	00_h	21_h	01_h	00_h	00_h	00_h	00_h
			Read Request	Index=2100_h		Module / node 1				

The Gateway module No. 10 sends an answers to the master via the read response ($4B_h$ = 2 Data bytes) with the identifier ***$58A_h$*** (580_h + Node-ID) and the value of the module with node number 1.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
$58A_h$	0_h	8_h	$4B_h$	00_h	21_h	01_h	$9C_h$	02_h	00_h	00_h
			Read Response (2Byte)	Index=2100_h		Module / node 1	read out value: 00 00 02 $9C_h$			

The read out value is converted into a decimal value. The max measuring value is 1024 and 39V are possible. The following equation results:

$$\text{Output voltage} = \frac{\textit{Decimal value} \times 39\text{V}}{1024}$$

Example:

00 00 02 $9C_h$ = decimal = 668

$$\text{Output voltage} = \frac{668 \times 39\text{V}}{1024} = \underline{25{,}44\text{V}}$$

4.4.9.5 Input Voltage (2101_h)

Index	Name	Data type	R/W
2101_h	*Input Voltage*	uint 16	ro

The value contains the amount of the adjacent input voltage.

Read out of the input voltage
The CANopen master sends the read request via the identifier '608_h' (600_h + Node-ID) to the Gateway No. 8 (Node-ID=8_h).
→ The secondary module with the node number 1 is read out.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
608_h	0_h	8_h	40_h	01_h	21_h	01_h	00_h	00_h	00_h	00_h
			Read Request	Index=2101_h		Module / node 1				

The Gateway module No. 8 sends an answers to the master via the read response ($4B_h$ = 2 Data bytes) with the identifier ***588_h*** (580_h + Node-ID) and the value of the module with the node number 1.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
588_h	0_h	8_h	$4B_h$	01_h	21_h	01_h	98_h	02_h	00_h	00_h
			Read Response (2Byte)	Index=2101_h		Module / node 1	read out value: 00 00 02 98_h			

The read out value is converted into a decimal value. The max measuring value is 1024 and 39V are possible. The following equation results:

$$\text{Input voltage} = \frac{\textit{Decimal value}\ 39\text{V}}{1024}$$

Example:

00 00 02 98_h = decimal = 664

$$\text{Input voltage} = \frac{664\ \text{x}\ 39\text{V}}{1024} = \underline{25{,}29\text{V}}$$

4.4.9.6 Current Measurement (2104_h)

Index	Name	Data type	R/W
2104_h	*Current measurement*	uint 16	ro

The value contains the amount of the flowing current.

Read out – Current measurement
The CANopen master sends the read request via the identifier ***$60A_h$*** (600_h + Node-ID) to the Gateway No. 10 (Node-ID=10_h).
→ The secondary module with the node number 1 is read out.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
$60A_h$	0_h	8_h	40_h	04_h	21_h	01_h	00_h	00_h	00_h	00_h
			Read Request	Index=2104_h		Module / node 1				

The Gateway module No. 10 sends an answers to the master via the read response ($4B_h$ = 2 Data bytes) with the identifier ***$58A_h$*** (580_h + Node-ID) and the value of the module with node number 1.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
$58A_h$	0_h	8_h	$4B_h$	04_h	21_h	01_h	$1F_h$	00_h	00_h	00_h
			Read Response (2Byte)	Index=2104_h		Module / node 1	read out value: 00 00 00 $1F_h$			

The read out value is converted into a decimal value. The max measuring value is 1024 and 32,75A are possible. The following equation results:

$$\text{Current} = \frac{\textit{Decimal value} \times 32{,}75\text{A}}{1024}$$

Example:

00 00 00 $1F_h$ = decimal = 31

$$\text{Current} = \frac{31 \times 32{,}75\text{A}}{1024} = \underline{0{,}99\text{A}}$$

4.4.9.7 Characteristic adjustment ($210A_h$)

Index	Name	Data type	R/W
$210A_h$	Characteristic adjustment	uint 16	ro

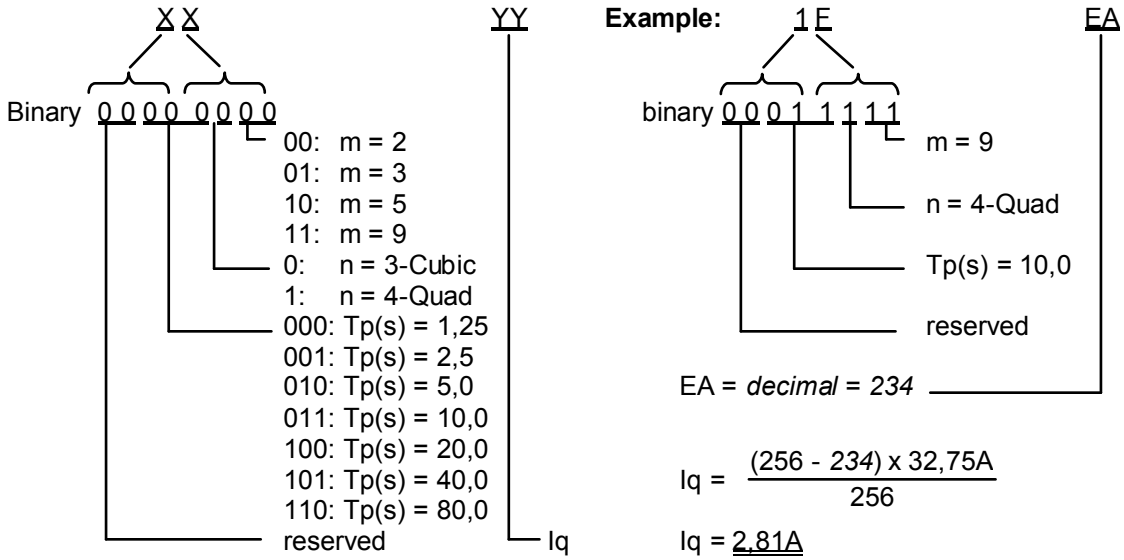
This object returns the current parameters of the characteristic adjusted.

Read out – Characteristic adjustment
The CANopen master sends the read request via the identifier ***605_h*** (600_h + Node-ID) to the Gateway No. 5 (Node-ID=5_h).
→ The secondary module with the node number 1 is read out.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
605_h	0_h	8_h	40_h	$0A_h$	21_h	01_h	00_h	00_h	00_h	00_h
			Read Request	Index=$210A_h$		Module / node 1				

The Gateway module No. 5 sends answers to the master via the read response ($4B_h$ = 2 data bytes) with the identifier ***585_h*** (580_h + Node-ID) and the value of the module with node number 1.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
585_h	0_h	8_h	$4B_h$	$0A_h$	21_h	01_h	EA_h	$1F_h$	00_h	00_h
			Read Response	Index=$210A_h$		Module / node 1	read out value: 00 00 1F EA_h (xx yy)			



Converting into *decimal value.* The max measuring value is 256 and corresponds to 32,75A.
The following rule of three results:

$$Iq = \frac{(256 - \text{Decimal value}) \times 32{,}75A}{256}$$

4.4.9.8 Software Version (2200_h)

Index	Name	Data type	R/W
2200_h	*Software Version*	uint 32	ro

This object returns the software version of the LOCC-Box.

Read out – Software version
The CANopen master sends the read request via the identifier ***608_h*** (600_h + Node-ID) to the Gateway No. 8 (Node-ID=8_h).
→ The secondary module with the node number 1 is read out.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
608_h	0_h	8_h	40_h	00_h	22_h	01_h	00_h	00_h	00_h	00_h
			Read Request	Index=2200_h		Module / node 1				

The Gateway module No. 8 sends an answers to the master via the read response (43_h = 4 data bytes) with the identifier ***588_h*** (580_h + Node-ID) and the value of the module with node number 1.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
588_h	0_h	8_h	43_h	00_h	22_h	01_h	12_h	00_h	00_h	00_h
			Read Response (4Byte)	Index=2200_h		Module / node 1	read out value: 00 00 00 12_h			

The read out value is converted into a decimal value.

Example:

00 00 00 12_h = 1.2

4.4.9.9 Serial Number (2201_h)

Index	Name	Data type	R/W
2201_h	*Serial Number*	uint 32	ro

This object returns the serial number of the LOCC-Box.

Read out - Serial number
The CANopen master sends the read request via under the identifier ***608_h*** (600_h + Node-ID) to the Gateway with No. 8 (Node-ID=8_h).
→ The secondary module with the node number 2 is read out.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
608_h	0_h	8_h	40_h	01_h	22_h	02_h	00_h	00_h	00_h	00_h
			Read Request	Index=2201_h		Module / node 2				

The Gateway module No. 8 sends an answer to the master via the read response (43_h = 4 data bytes) with the identifier ***588_h*** (580_h + Node-ID) and the value of the module with node number 2.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
588_h	0_h	8_h	43_h	01_h	22_h	02_h	EF_h	$E1_h$	01_h	00_h
			Read Response (4Byte)	Index=2201_h		Module / node 2	read out value: 00 01 E1 EF_h			

The read out value is converted into a decimal value.

Example:

00 01 E1 EF_h = decimal = 123375

4.4.9.10 LOCC-Box counter “Operating voltage ON“ (2202_h)

Index	Name	Data type	R/W
2202_h	LOCC-Box counter “Operating voltage On“	uint 32	ro

This object returns the count how many times the module has been connected to the supply voltage.

Read out – Operating voltage ON
The CANopen Master sends the read request via the identifier ***608_h*** (600_h + Node-ID) to the Gateway with No. 8 (Node-ID=8_h).
→ The secondary module with the node number 2 is read out.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
608_h	0_h	8_h	40_h	02_h	22_h	02_h	00_h	00_h	00_h	00_h
			Read Request	Index=2202_h		Module / node 2				

The Gateway module No. 8 sends an answer to the master via the read response (43_h = 4 data bytes) with the identifier ***588_h*** (580_h + Node-ID) and the value of the module with node number 2.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
588_h	0_h	8_h	43_h	02_h	22_h	02_h	$0C_h$	01_h	00_h	00_h
			Read Response (4Byte)	Index=2202_h		Module / node 2	read out value: 00 00 01 $0C_h$			

The read out value is converted into a decimal value.

Example:

00 00 01 $0C_h$ = decimal = 268

4.4.9.11 LOCC-Box Counter “Operating hours (h)” (2203_h)

Index	Name	Data type	R/W
2203_h	LOCC-Box Counter “Operating hours (h)”	uint 32	ro

This object returns the number of the operating hours in ½ hour cycle, this means how long the LOCC-Box is connected to the supply voltage.

Read out – Operating hours (h)
The CANopen master sends the read request via the identifier ***608_h*** (600_h + Node-ID) to the Gateway No.8 (Node-ID=8_h).
→ The secondary module with the node number 2 is read out.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
608_h	0_h	8_h	40_h	03_h	22_h	02_h	00_h	00_h	00_h	00_h
			Read Request	Index=2203_h		Module / node 2				

The Gateway module No. 8 sends an answer to the master via the read response (43_h = 4 data bytes) with the identifier ***588_h*** (580_h + Node-ID) and the value of the module with node number 2.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
588_h	0_h	8_h	43_h	03_h	22_h	02_h	60_h	01_h	00_h	00_h
			Read Response (4Byte)	Index=2203_h		Module / node 2	read out value: 00 00 01 60_h			

The read out value is converted into a decimal value.

Example:

00 00 01 60_h = decimal / 2 = 176h

4.4.9.12 LOCC-Box counter “Operating hours ON (h)“ (2204_h)

Index	Name	Data type	R/W
2204_h	LOCC-Box counter “Operating hours ON (h)“	uint 32	ro

This object returns the number of the operating hours ON in ½ hour cycle, this means how long the LOCC-Box has been switched on and how long it has supplied the load.

Read out – Operating hours ON (h)
The CANopen Master sends the read request via the Identifier ***608_h*** (600_h + Node-ID) to the Gateway No. 8 (Node-ID=8_h).
→ The secondary module with the node number 2 is read out.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
608_h	0_h	8_h	40_h	04_h	22_h	02_h	00_h	00_h	00_h	00_h
			Read Request	Index=2204_h		Module / node 2				

The Gateway module No. 8 sends an answer to the master via the read response (43_h = 4 data bytes) with the identifier ***588_h*** (580_h + Node-ID) and the value of the module with node number 2.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
588_h	0_h	8_h	43_h	04_h	22_h	02_h	FB_h	08_h	00_h	00_h
			Read Response (4Byte)	Index=2204_h		Module / node 2	read out value: 00 00 08 FB_h			

The read out value is converted into a decimal value.

Example:

00 00 08 FB_h = decimal / 2 = 1149.5h

4.4.9.13 LOCC-Box counter “Blown” (2205_h)

Index	Name	Data type	R/W
2205_h	LOCC-Box counter “Blown”	uint 32	ro

This object gives the information how many times the LOCC-Box has blown because of overload or short circuit.

Read out - blown
The CANopen Master sends the read request to the gateway via the identifier ***608_h*** (600_h + Node-ID) No. 8 (Node-ID=8_h).
→ The secondary module with the node number 2 is read out.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
608_h	0_h	8_h	40_h	05_h	22_h	02_h	00_h	00_h	00_h	00_h
			Read Request	Index=2205_h		Module / node 2				

The Gateway module No. 8 sends an answers to the master via the read response (43_h = 4 data bytes) with the identifier ***588_h*** (580_h + Node-ID) and the value of the module with node number 2.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
588_h	0_h	8_h	43_h	05_h	22_h	02_h	28_h	00_h	00_h	00_h
			Read Response (4Byte)	Index=2205_h		Module / node 2	read out value: 00 00 00 28_h			

The read out value is converted into a decimal value.

Example:

00 00 00 28_h = decimal = 40

4.4.9.14 LOCC-Box Counter “Switch on” (2206_h)

Index	Name	Data type	R/W
2206_h	LOCC-Box Counter “Switch on”	uint 32	ro

This object gives the information how many times the LOCC-Box has been switched on.

Read out – Switch on
The CANopen master sends the read request via the identifier ***608_h*** (600_h + Node-ID) to the gateway with the No. 8 (Node-ID=8_h).
→ The secondary module with the node number 3 is read out.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
608_h	0_h	8_h	40_h	06_h	22_h	03_h	00_h	00_h	00_h	00_h
			Read Request	Index=2206_h		Module / node 3				

The gateway module No. 8 sends an answer to the master via the read response (43_h = 4 data bytes) with the identifier ***588_h*** (580_h + Node-ID) and the value of the module with node number 3.

ID	RTR	LEN	Data							
			1	2	3	4	5	6	7	8
			Comm. code	Index low	Index high	Sub-Index	LSB	…	…	MSB
588_h	0_h	8_h	43_h	06_h	22_h	03_h	$2C_h$	00_h	00_h	00_h
			Read Response (4Byte)	Index=2206_h		Module / node 3	read out value: 00 00 00 $2C_h$			

The read out value is converted into a decimal value.

Example:

00 00 00 $2C_h$ = decimal = 44

4.5 Technical data

General data

Nominal voltage	DC 12/24V
Operating voltage range	DC 10 – 32V
Nominal current	max. 50mA
Polarization protection	yes
Termination	spring terminal
Housing material	PA 6.6 (UL 94 V0; NFF I2, F2)
Mounting	snap on TS 35 (according to EN 50022)
Protection level	IP 20
Mounting position	any
Installation technology	spring terminal 0.25mm^2 – 2.5mm^2 all types of wires up to 2.5mm² without end sleeve up to 1.5mm² with end sleeve
USB	USB 2.0 Full-Speed (12 Mbit/s)
UART (RS232)	Baud rate 600 - 115200 bit/s
CANopen	Baud rate 10 – 1000 kbit/s
Operation temperature	-20°C to +60°C
Store temperature	-40°C to +85°C
Dimensions (WxHxD)	6.2 x 90 x 115.5mm
Weight	0.06 kg
Approvals	CE
Standards	EN 60950-1; EN61131-1,2; EN 60898; EN 60947-4-1; EN 50081

LOCC-BUS

Access method	Single-Master - Multiple Slave
Bus technology	line
Physical level	1-wire
Subscriber	typical 40, max. 84
Bus length	typical 10 m, max. 40 m
Transfer rate	9600 Baud
Data rate	8 Bit + fixed parity
Transmission protocol	modified Multidrop

5 Gateway – Profinet, 716457

The LOCC-Box Gateway is an electronic part which distributes and transforms the data and the messages of the serial LOCC-Box-Net interface (LOCCbus) to 2 further communication interfaces USB or Profinet.

5.1 General Information

5.1.1 Explanation

The serial LOCC-Box-interface is a 1 wire communication interface. This is made according to the LIN specification. The protocol on this interface is leant on the Multidrop Protocol.

The Gateway supports the following interfaces:

- Full-Speed USB-interface with a max Bit rate of 12 MBit/s according to USB 2.0
- Profinet-IO Interface according to IEC 61158. Physical transmission layer is the Ethernet 100Base/T.

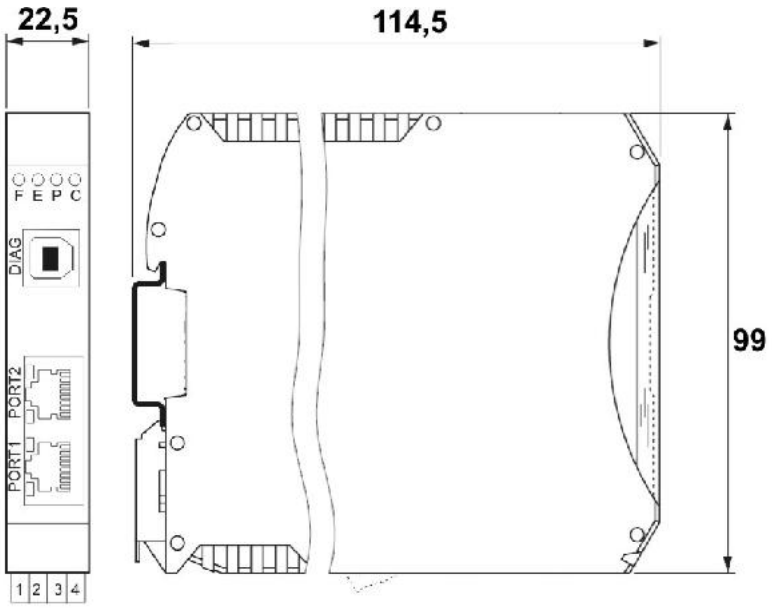
The USB-Interface is used for the connection to a common computer. The USB-interface is recognized under Windows XP [1)] or Windows Vista [1)] as serial COM-Interface. Together with the Software LOCC-Pads the interface is used for the initial operation and configuration of the LOCC-Box-Net.

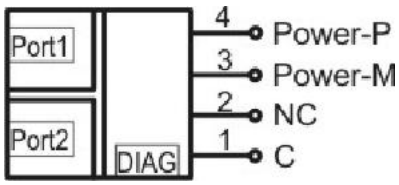
The Profinet interface with 2 ports is suitable for connecting example to a programmable logic controller (PLC) of different manufacturers.

A simultaneous operation mode of the USB- and Profinet-IO interface is not possible. In this case the communication through the USB interface has always priority.

The LIN-interface, the power supply for the LIN-interface and the power for the Gateway (P and M) is connected about 4 pluggable spring terminals. The USB-interface (form B) and the Profinet RJ-45 port are available in the front of housing.

5.1.2 Dimensions and Connections





1 C: 1 wire bus, LOCC-Box-Net
2: NC: - not connected
3: Power-M: 0V
4: Power-P: DC12/24V

DIAG: USB-Interface
Port1: Profinet RJ-45 terminal 1
Port2: Profinet RJ-45 terminal 2

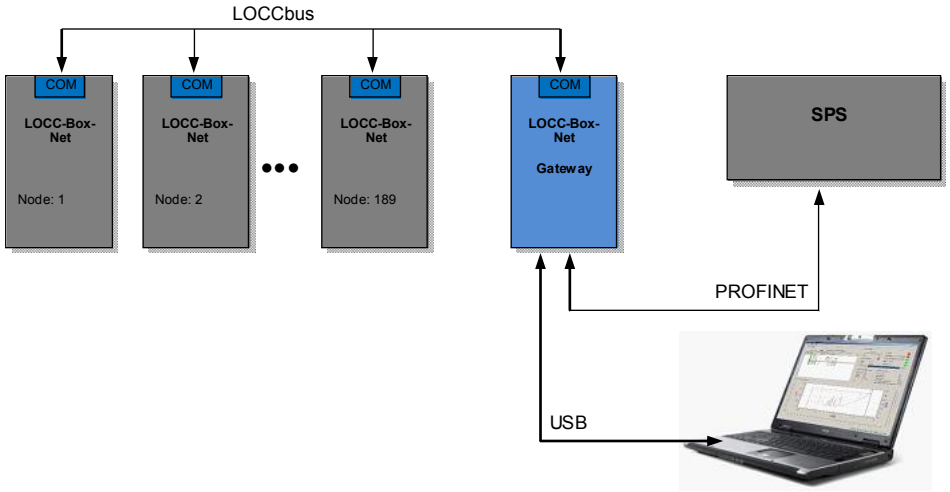
5.1.3 Function and Displays

Function	PIN	Description
C	1	Communication terminal, 1 wire bus, LOCCbus
NC	2	Not connected
Power-M	3	0V – terminal for the internal power of the gateway
Power-P	4	DC 12/24V – terminal for the internal power of the gateway

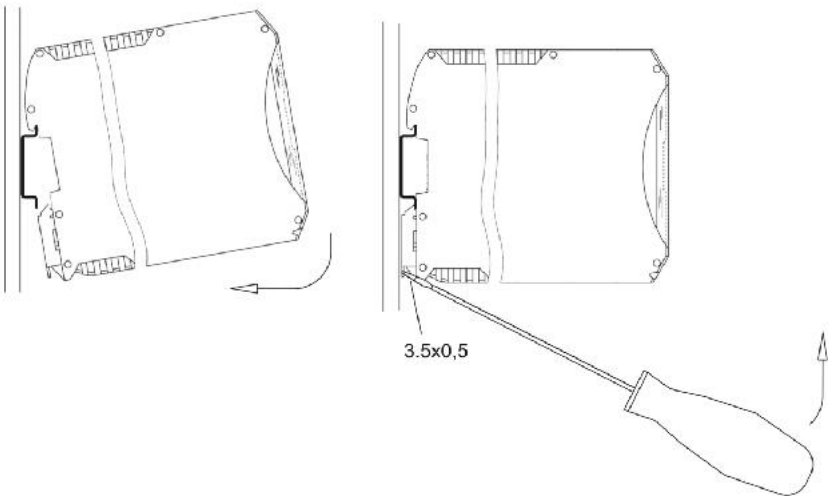
Connection: spring terminal, pluggable

Displays	Function	Description
LED F, yellow – blinking	PROFINET	Request for identification
LED E, red – lighting	PROFINET	No Profinet communication
LED P, green – lighting	Power	Power supply is connected
LED C, green - blinking	LOCCbus	Data traffic with LOCC-Box-Net modules
RJ-45 LED yellow	Link	100Base/T-communication
RJ-45 LED green	Activity	ON: valid communication, blanked: data traffic

5.1.4 Topology and Structure



5.1.5 Mounting



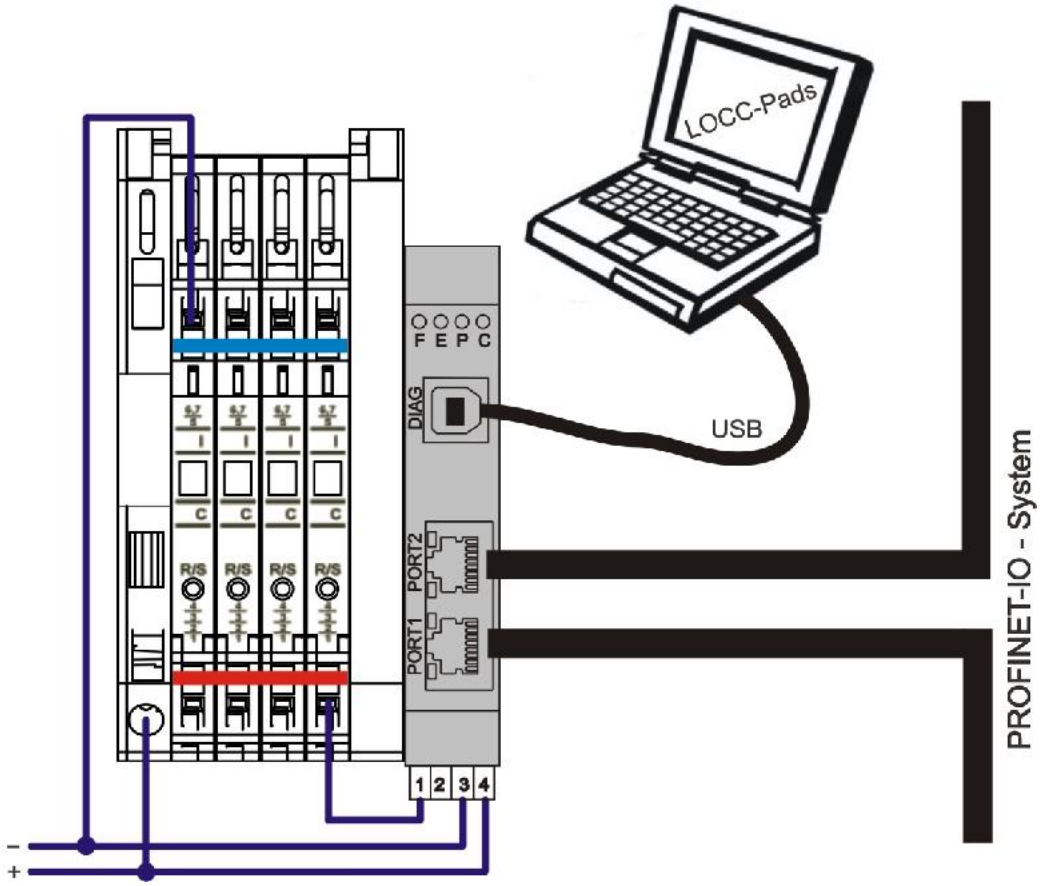
5.1.6 Installation

1. Supply the Gateway and all LOCC-Box-Net modules with an operating voltage of DC 12/24V.

2. Connect the ***COM*** connectors of the modules to the Gateway. For this purpose jumper combs can be used. *See chapter* **7** *Accessories.*

3. Connect the Gateway via the USB interface to the computer to use LOCC-Pads. For a Profinet communication connect the field bus cable to port 1 and port 2.

.



Connection to USB

Connect the Gateway to the computer by using the provided USB cable.
At the initial connection, the Gateway will find a new Hardware ***USB Serial Port*** and the ***Found new Hardware wizard*** will prompt.

Please choose ***Install the software automatically*** and confirm by clicking ***Next***. Follow the instructions of the wizard, which searches and installs the driver.

5.2 Communication via USB

See chapter **Fehler! Verweisquelle konnte nicht gefunden werden.** *LOCC-Pads*.

5.3 Communication via Profinet

Profinet is 100% conform to the Ethernet-Standard IEEE 802.3. It works full duplex and support Industrial Ethernet with a transfer speed of 100Mbit/s.

Used switches for NRT- and RT-Communication must support:

- Full duplex,
- Auto-Crossover (for using 1:1 cable) and
- Auto Negotiation (transfer from 10 to 100Mbit/s)

Following properties are recommended:

- Redundant power supply,
- Diagnostic LEDs and
- Status relay contact

5.3.1 Terms and Definitions

Alarm	Status signal of an event: The alarms are subdivided in process-/diagnosis-, pull-/plug-, return-, supervisor and redundant alarms.
Auto-Cross-Over	Recognizes automatically if the transmitting or receiving wire is crossed over or not
Auto-Negotiation	Automatic negotiation of the transmitting speed between two subscribers.
Broadcast	An unacknowledged transmission to multiple, unspecified recipients on a bus segment.
Cat	Category: The classification of cables. Used for ETHERNET. For PROFINET the Cat5 is mandatory.
CBA	Component Based Automation
CControl	Reports the start of the operation mode of an I/O module.
Consumer	Device that receives data from a provider.

Data Status	Collection of flags. The flags showing if the data are valid and can be used (they also showing the status of the diagnosis and the global status of the IO controller).
Device_ID	Device Identification: It is part of PROFINET-IO.
DL	Data Link: known as Layer 2. Profinet uses Ethernet (IEEE 802.3).
Ether type	Identification of an Ethernet frame by a 16 bit number assigned by IEEE.
GSDML	Generic Station Description Mark-up Language: Device Identification to generate a device root file (GSD). The file is XML based.
Hub	Active network component to connect single Ethernet subscriber: The hub forwards the frames to all subscribers, also to the non addressed ones.
IO-Device	A Profinet device which is for coupling onto an I/O Controller. I/O Devices are Input and Output modules.
IO-Controller	Device (typical: a controller) which initiates the I/O Data traffic.
IP	Internet Protocol: assures the data transfer in the internet from end node to end node.
IRT	Isochronous Real Time: Profinet communication channel which provides the synchronic, time controlled transfer of data.
MAC-Address	Media Access Control Address: Also known as Ethernet Address or physical Address. For Identifying a Ethernet node. The Mac address is 6 bytes long and is assigned by IEEE.
PCD	PROFINET Component Description: Device description of a PROFINET component.
Ping	Telegram which checks if the associated module is still available.
PROFINET	**PRO**cess **FI**eld **NET**work: Is the open industrial Ethernet standard for automation provided by PNO.
RT	Real Time: Identification of a real time protocol, which can be perform without a special support of a standard Ethernet communication.
Runtime	Describes the data exchange.
TCP	Transmission Control Protocol: is one of the main protocols in TCP/IP networks. TCP guarantees delivery of data and also guarantees that packets will be delivered in the same order in which they were sent.
UDP	User Datagram Protocol: Unsecured Multicast-/Broadcast telegram.

5.3.2 Operating System and Driver

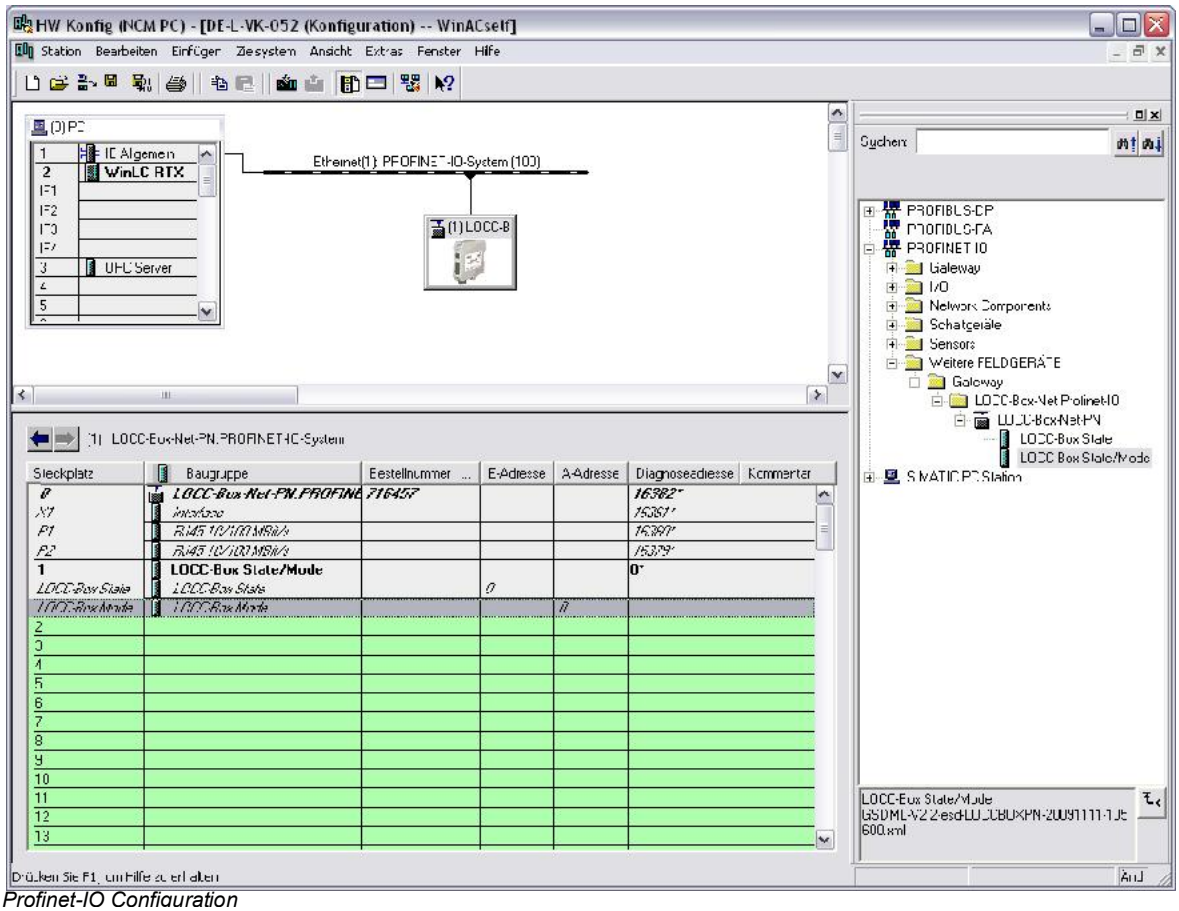
Operation system	eCos
Driver	Siemens-ComDec Profinet Stack V3.1.0

5.3.3 GSDML - Files

The GSDML file which can be used is in the ***LOCC-Pads_x.x.x.x.zip*** file included. The file can be downloaded free of charge on the Lütze website. According to the current control unit scheme V2.2 we offer the version ***GSDML-V2.2-esd-LOCCBOXPN-20100216-113200.xml***. For older control units (PLC) according to scheme V1.0 we provide the version ***GSDML-V1.0-esd-LOCCBOXPN-20100218-172400.xml***.

5.3.4 Profinet-IO Interface

In the Profinet-IO system the Gateway operates like modular equipment with 127 (254) slots. With these slots up to 127 (254) modules can be connected. (Module = LOCC-Box-Net)



Profinet-IO Configuration

5.3.5 Function Range

The Gateway supports (port to port) all Profinet–IO functions.

- ***Cyclic data exchange:*** IO data will transfer in an adjustable cycle between provider and consumer. The customer can select different properties for the transmit interval and the receive interval. (RT-channel)

- ***Alarm handling:*** All process and diagnostic events have to be transferred by a Profinet system alarm. The IO-device transfers the alarms as high priority RT-message. Alarms must be acknowledged. (RT-channel)

- ***Acyclic data exchange:*** Need-oriented data, for example writing and reading of parameters or reading of diagnostic information. Can be exchanged acyclic via the NRT channel.

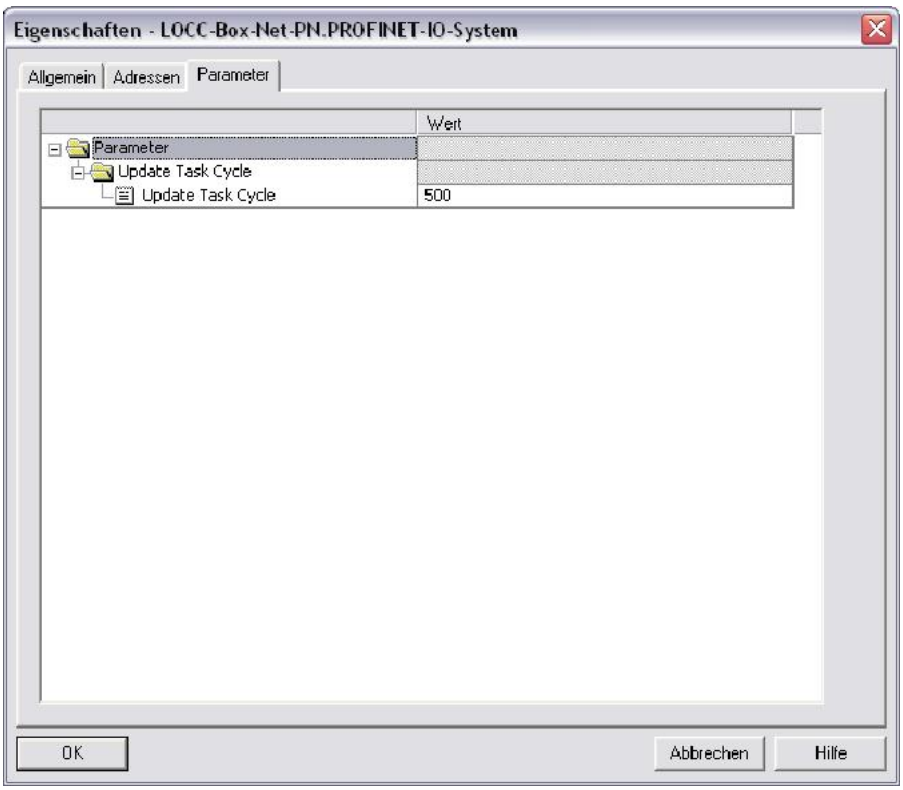
- ***Data cross-traffic:*** Process data can be exchanged between several field devices directly without superior SPS.

- ***Synchronous:*** The data exchange in standard Ethernet communication is subject to a certain dispersion. The planned bus cycle can deviate up to 100%. The synchronous ensures that the bus cycle is always kept exactly. Deviations from beginning of a Bus cycle from < 1µs are guaranteed. (IRT channel)

- ***Automatically address allocation:*** The address assignment is done automatically in a integrated standardized Profinet protocol.

5.3.6 Device Access Point (DAP)

The *Device Access Point* has following parameter: the cycle time for task time in millisecond.
Range: 20 ... 65535 ms



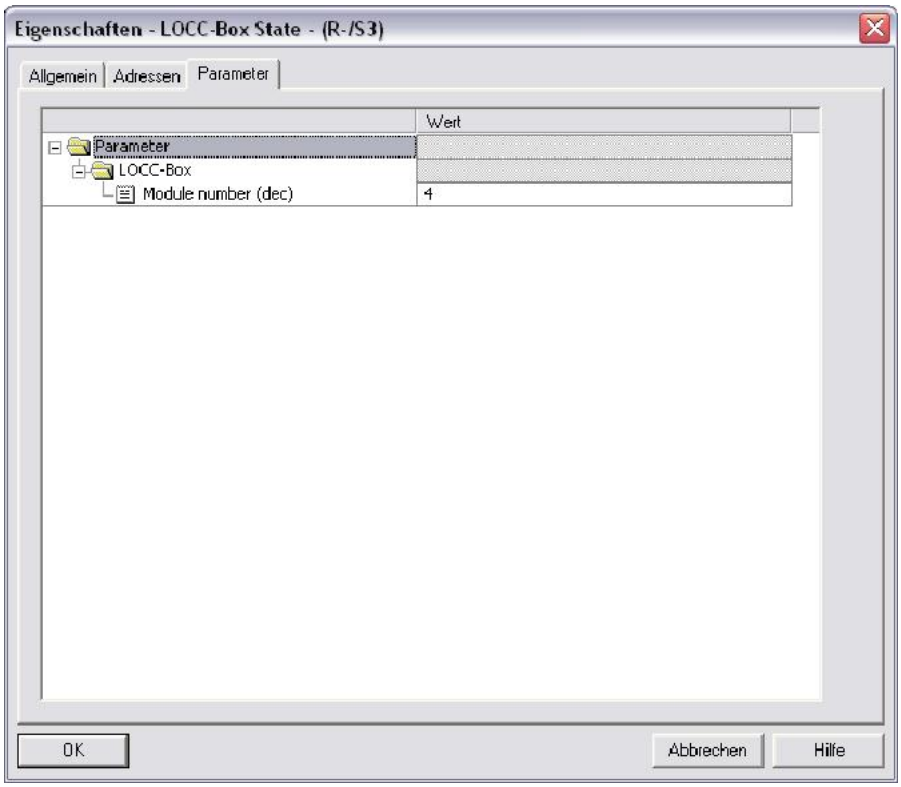
Parametrization LOCC-Box-PN

5.3.7 Process Image

Per attached LOCC box a I/O module is inserted.
2 types of modules are available::

- LOCC-Box state with 1 byte input
- LOCC-Box state/mode with 1 byte input and 1 byte output

The module state and module state/mode has a parameter: the address from the connected LOCC-Box on the LOCCbus (value range 1...254).



Configuration of LOCC-Box-Modules

The input byte includes the module state information of the connected LOCC-Box. The module state is according to the command 4 in the RS232-communication (*see chapter* **Fehler! Verweisquelle konnte nicht gefunden werden.**) or the index 0x2010 in the CANopen-communication (*see chapter 4.4.9.2*).

7	6	5	4	3	2	1	0
System error	Short-circuit	Undervoltage U<10V	Iwarning (I>0,9 * Inom)	New module on bus	Reserve	Status	

The output byte supports the 2 last signification bits and is used for switching on and off the LOCC-Box.

Bit 0: = 0: connected LOCC-Box will be switch off
= 1: connected LOCC-Box will be switch on
Bit 1: edge from 0 to 1: The status of bit 0 is transferred in the connected LOCC-Box.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	Rising edge = take over	New status

HINWEIS

All data's are transferred in Hex-format.
Switching ***On*** is made by sending „00 "and afterwards from „03 ".
Switching ***Off*** is made by sending „00 "and afterwards from „02 ".

5.3.8 Profinet-IO Read Request

All information about the modules are requested via the Profinet-I/O service ***Read Request***. The requested LOCC-Box is addressed via the slot number. The data are represented by an index like in the table below:

Index	Name	Data type	R/W	Example in section
2000_h	Module type	uint 8	ro	4.4.9.1
2010_h	Module status	unit 8	ro	4.4.9.2
2011_h	Module configuration	unit 8	ro	4.4.9.3
2100_h	Output voltage	unit 16	ro	4.4.9.4
2101_h	Input voltage	unit 16	ro	4.4.9.5
2104_h	Current measurement	unit 16	ro	4.4.9.6
$210A_h$	Characteristic adjustment	unit 16	ro	4.4.9.7
2200_h	Software version	unit 32	ro	4.4.9.8
2201_h	Serial number	unit 32	ro	4.4.9.9
2202_h	LOCC-Box counter “Operation voltage AN“	unit 32	ro	4.4.9.10
2203_h	LOCC-Box counter “Operation hour (h)“	unit 32	ro	4.4.9.11
2204_h	LOCC-Box counter “ Operation hour AN (h)“	unit 32	ro	4.4.9.12
2205_h	LOCC-Box counter “Blown“	unit 32	ro	4.4.9.13
2206_h	LOCC-Box counter “Switch on“	unit 32	ro	4.4.9.14

ro = only read

5.3.9 Siemens – Function Block SFB52

Siemens has a special function block which contains all parameters to request additional information. *See chapter 5.3.8.*

Example:
Current measurement of module with node number 3. The index 2104_h is to convert in a decimal value → 8452_d.

```
OB1 :   "Main Program Sweep (Cycle)"
Kommentar:

Netzwerk 1: Titel:
Kommentar:

      L     EB     1                  // Lade Modulzustand von LOCC-Box an Adresse 1: hier Knotennummer 1

      L     3                         // Lade Akku mit 3: Bit 0 = An, Bit 1 = steigende Flanke
      T     AB     3                  // speicher Akku in LOCC-Box an Adresse 3: hier Knotennummer 3
      CALL  "RDREC" , DB1
       REQ    :=M10.0                 // Einschalten der Funktion
       ID     :=DW#16#1               // Adresse der LOCC-Box: hier Knotennummer 1
       INDEX  :=8452                  // = 2104hex -> Ausgangsstrom
       MLEN   :=4                     // es werden 4 Byte erwartet
       VALID  :=M10.1                 // 1: Daten sind gültig
       BUSY   :=M10.2                 // 1 = Funktion ist noch nicht fertig
       ERROR  :=M10.3                 // 1 = Es ist ein Fehler aufgetreten
       STATUS:=MD12                   // hier wird der Fehlercode zurückgegeben
       LEN    :=MW16                  // die Anzahl der wirklich gelesenen Bytes
```

The saved data (4 Byte) have to be evaluated like in chapter **Fehler! Verweisquelle konnte nicht gefunden werden.** described.

5.3.10 Optional I&M – Services

The optional I&M (Identification & Maintenance) services are integrated in the Gateway.
In the simatic manager the services appearing automatically under properties. The properties can be changed and saved permanently in the module to simplify the identification.

If recognizing a new LOCC-Box-Net a Profinet diagnosis alarm is send in slot 0 with error number 0x13 (manufacturer specific). If the LOCC-Box-Net is isolated or configured via LOCC-Pads the diagnosis alarm stops.

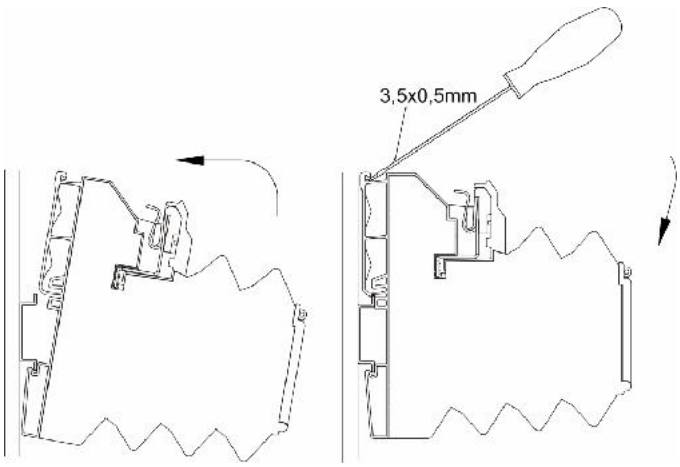
5.4 Exchanging the LOCC-Box-Net without LOCC-Pads

Exchanging the LOCC-Box with an existing configuration is possible without LOCC-Pads.

Requirement:

- It is only possible to change one module at a time.
- The new module has to be in the default setting. It has to have the node number 0.
- Profinet communication must be existing.

1. Start the communication.
2. Remove the jumper combs. Slide back the contact at connection 7.
3. Remove the module as shown in the picture.



4. Set the current value (I) and the characteristic (C) with the rotary switches on the new module.
5. Snap on the module – see picture
6. Close the sliding contact and reinstall the jumper combs.
7. The new LOCC-Box is blinking. Press the on/off switch within one minute; otherwise the module does not get a node number. If missing the time, remove the module and reinstall it again.

   NOTICE During that time, no communication is possible.

8. Switch the LOCC-Box Off and On again, otherwise the current and characteristic settings are not active

5.5 Technical Data

General Data

Rated voltage	DC 12/24V
Operation voltage	DC 10 – 32V
Rated current	max. 120mA
Polarity protection	yes
Housing material	PA 6.6 (UL 94 V0)
Mounting	snap on TS 35 (according to EN 50022)
Protection level	IP 20
Mounting position	any
Termination	spring terminal 0,25mm^2 – 2,5mm^2 all types of wire up to 2,5mm² without end sleeve up to 1,5mm² with end sleeve
USB	USB 2.0 Full-Speed (12 Mbit/s)
Profinet	100 Mbit/s
Operation temperature	-20°C to +60°C
Store temperature	-40°C to +85°C
Relative humidity	max. 90%, without condensation
Dimension (WxHxD)	22,5 x 99 x 114,5mm
Weight	0,130 kg
Approval	CE
Standards	EN 60950-1; EN61131-1,2; EN 60947-4-1; EN 50081

LOCC-BUS

Access method	Single-Master - Multiple Slave
Bus technology	line
Physical level	1-wire
Subscriber	typical 40, max. 84
Bus length	typical 10m, max. 40m
Transfer rate	9600 Baud
Data rate	8 Bit + fixed parity
Transmission-protocol	Modifiziertes Multidrop

6 Gateway EtherCAT – 716456

The LOCC-Box-Net Interface (Gateway) is an electronic assembly to allocate and convert the data and messages of the serial LOCC-Box-Net interface (LOCC Bus) to two additional communication interfaces like USB or EtherCAT.

6.1 General

6.1.1 Explanation

The serial LOCC-Box Interface is a 1-wire communication interface. The physical electrical interface is manufactured according to the LIN specification. The protocol of the interface is related to the Multidrop Protocol.
The gateway supports following interfaces:

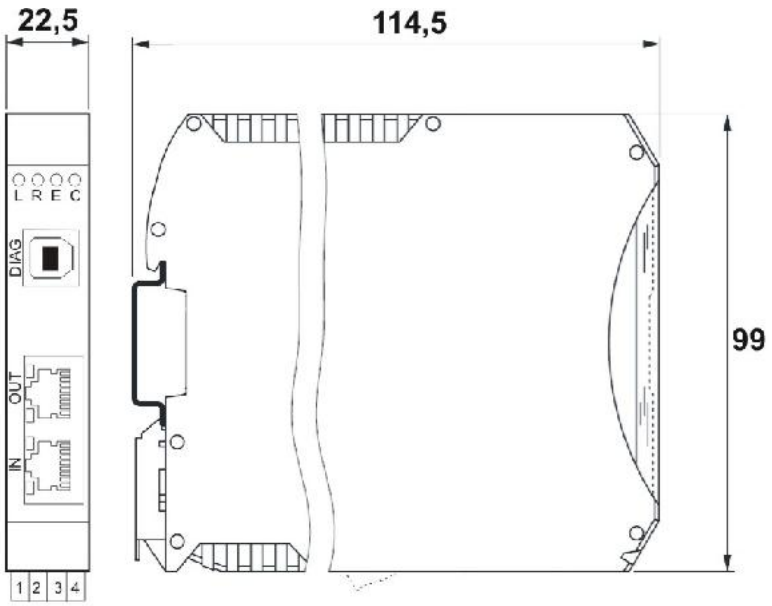
- Full-Speed USB interface with a maximum bit rate of 12 MBit/s according to USB 2.0
- EtherCAT-Interface ***IN*** and ***OUT***. Ethernet 100Base/T is used as the transmitting layer.

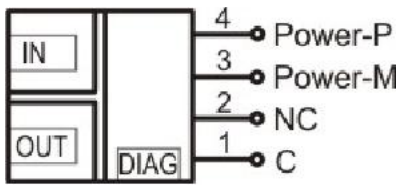
The USB interface is for connecting a computer. The interface is used as a serial COM interface by the operating system. With the software the interface is used for the initial operation and the configuration of the LOCC-Box-Net assembly.

The EtherCAT interface (***IN*** and ***OUT***) is for connecting Programmable Logic Controller (PLCs) of different manufactures. The communication of the 2 gateway interfaces, USB and EtherCAT are exclusive on the LOCC-Box-Net LIN interface. A crosswise communication (USB-EtherCAT) among each other is not possible.

The LIN interface including the power supply and the gateway (P and M) are routed via the 4 clamps of the housing. The USB interface can be found on the front side of the module, like the EtherCAT – RJ45 sockets.

6.1.2 Dimensions and Connections





1 C: 1 wire bus, LOCC-Box-Net
2: NC: not connected
3: Power-M: 0V
4: Power-P: DC12/24V

DIAG: USB-Interface
IN: RJ-45 terminal “IN”
OUT: RJ-45 terminal “OUT“

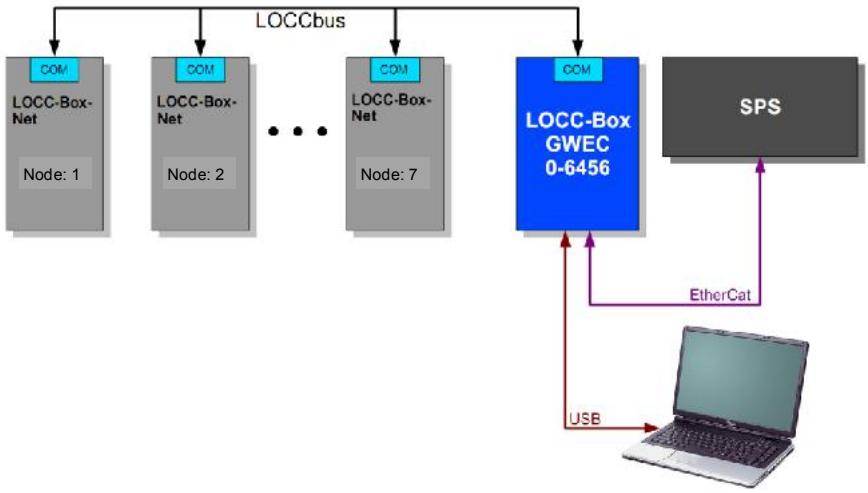
6.1.3 Function and Displays

Function	PIN	Description
C	1	Communication terminal, 1 wire bus, LOCCbus
NC	2	Not connected
Power-M	3	0V – terminal for the internal power of the gateway
Power-P	4	DC 12/24V – terminal for the internal power of the gateway

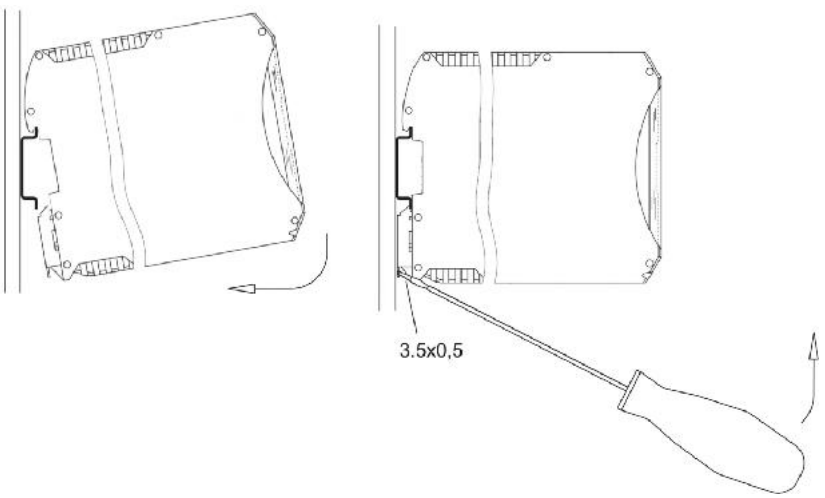
Connection: spring terminal, pluggable

Display	Function	Description
LED L, red – lighting	Error	EEPROM Error, EEPROM not loaded
LED R, green – lighting	EtherCAT	ECT Run
LED E, green – lighting	EtherCAT	ECT Error
LED C, green – blinking	LOCCbus	Data traffic with LOCC-Box-Net modules
RJ-45 LED green	Link/Activity	100Base/T-connection, blinking at EtherCAT-traffic
RJ-45 LED green	Connect	Speed-LED, 100Base/T-connection

6.1.4 Topology and Structure



6.1.5 Mounting

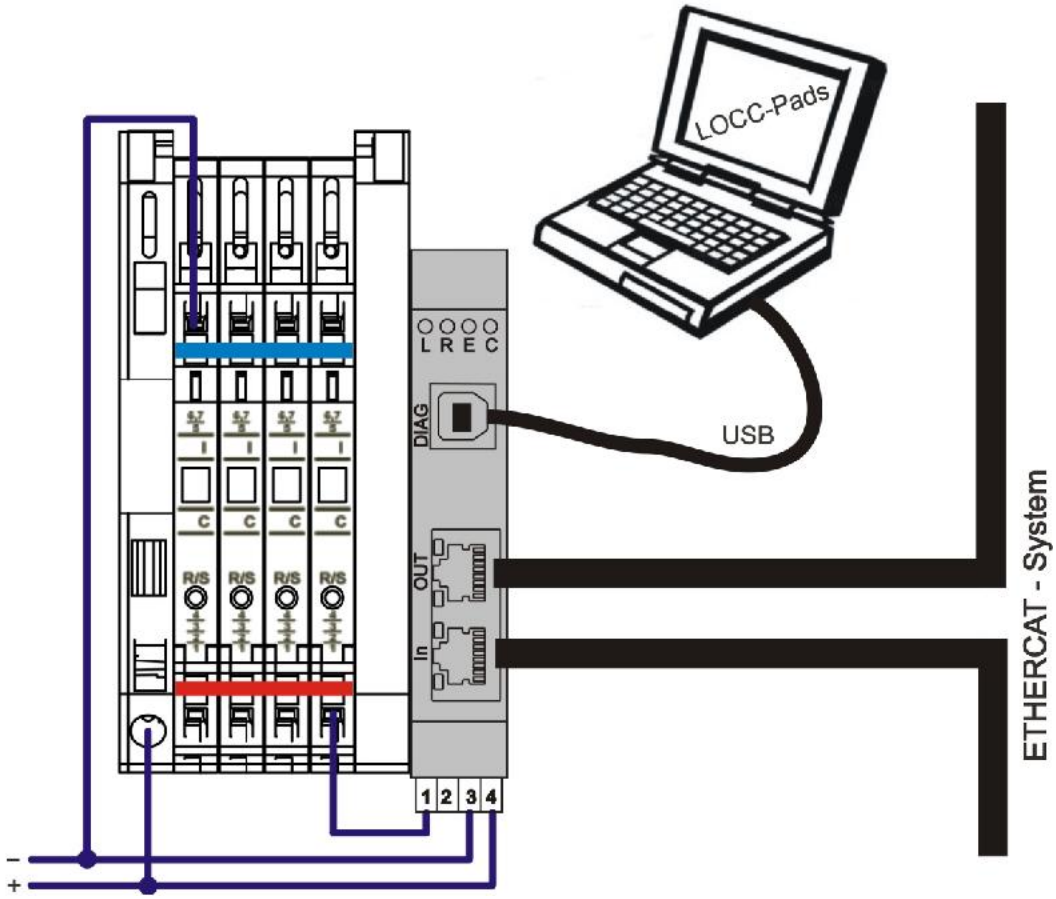


6.1.6 Installation

1. Supply the Gateway and all LOCC-Box-Net modules with an operating voltage of DC 12/24V.

2. Connect the ***COM*** connectors of the modules to the Gateway. For this purpose jumper combs can be used. *See chapter* **7** *Accessories.*

3. Connect the Gateway via the USB interface to the computer to use LOCC-Pads. For a EtherCAT communication connect the field bus cable to port “IN”.



Connection via USB

Connect the gateway to the computer via the provided USB cable.

At the first start the gateway is recognized as new hardware ***LOCC-Box-GW-EX 716456*** by the computer. The ***Found new Hardware*** wizard will prompt.

Choose ***Install the software automatically*** and confirm by clicking ***Next***. Follow the instructions of the wizard, which searches and installs a driver.

6.2 Communication via USB

See chapter 3 LOCC-Pads

6.3 Communication via EtherCAT

EtherCAT is 100% Ethernet-Standard (according to standard IEEE 802.3). It works full duplex and supports the industrial Ethernet with a transmitting rate of 100 Mbit/s (Fast Ethernet).
The device meets the requirements of the EtherCAT standard ETG.1000x S (R) V1.0.2 with x=2..6, ETG.1300 S (R) V1.0.3 and ETG.2000 S (R)V1.0.2

EtherCAT is an IEC, ISO and SEMI standard: (IEC 61158, IEC 61784, ISO 15745, SEMI E54.20)

6.3.1 Technical Terms

CoE	CAN application protocol over EtherCAT	(Former: “CANopen over EtherCAT”)
DC	Distributed Clock	
EEPROM	Electrically Erasable Programmable Read Only Memory	
ENI	EtherCAT Network Information	Contains configuration information for an EtherCAT master
EoE	Ethernet over EtherCAT	
ESC	EtherCAT Slave Controller	
ESI	EtherCAT Slave Information	Contains information about EtherCAT slave devices
ETG	EtherCAT Technology Group	Homepage: www.ethercat.org
FMMU	Field bus Memory Management Unit	
Hex Bin	Hexadecimal Binary	Used to display data in hexadecimal notation, e.g. decimal 43707 = 0xAABB = Hex Bin: “BB AA” (little-endian)

Init	EtherCAT device state “Init”	
IP	“Init” → “PreOp”	State transition from “Init” to “PreOp”
LCID	Locale Identifier	A number that describes a Language / culture setting
MAC	Media Access Control	
MBox	Mailbox	EtherCAT slave device Mailbox
NIC	Network Interface Card	Also used synonymic to “Network Interface”
NOP	No Operation	EtherCAT command that is ignored by the slaves
Op	EtherCAT device state “Operational”	
PDI	Process Data Interface	
PDO	Process Data Object	
PDU	Protocol Data Unit	
PreOp	EtherCAT device state “Pre-Operational”	
SafeOp	EtherCAT device state “Safe-Operational”	
SM	Sync Man, Synchronization Manager	
WKC	Working Counter	Data field within an EtherCAT PDU used for error detection, see ETG.1000 documents for details

6.3.2 Operation System, Interface

Operation System	Free-RTOS
Driver	ESD EtherCAT Slave Stack
Update-Mechanism	via USB with Configuration Program LOCC Pads
Type	EtherCAT Slave
Controller	Beckhoff ET1100
Galvanic Isolation	Transmitter
Description	IN and OUT, RJ45 connector with LED status indication

6.3.3 Description File

The ESI file (EtherCAT Slave Information) can be found in the ***LOCC-Pads_x.x.x.x.zip*** file. The file can be downloaded free of charge on the Lütze website.

6.3.4 EtherCAT Interface

In the EtherCAT System the gateway functions as a slave with a maximum of 64 modules.
Module = LOCC-Box-Net.

6.3.5 Communication Profile Area

Index	Sub-Index	Name	Data type	R/W	Default	Example in Chapter
1000_h	-	Device Type	UDINT	ro	00000191_h	4.4.7.1
1008_h	-	Device Name	String	ro	LOCC-Box-GWEC	4.4.7.6
1009_h	-	Hardware Version	String	ro	1.0	X,Y
$100A_h$	-	Software Version	String	ro	2.0	X,Y
1018_h	4	Identity Object	Record	ro	-	4.4.7.17
	1	Vendor-ID			$63C_h$	
	2	Product code, hexadecimal			716456	
	3	Revision number			X.Y	
	4	Serial number			0_h	
1600_h	64	RxPDO1	Record	ro		4.4.7.21
$1A00_h$	64	TxPDO1	Record	ro		4.4.7.23
$1C00_h$	4	SM Type	Array	ro		
$1C12_h$	1	PDO Assignment	Array	ro		
$1C13_h$	1	PDO Assignment	Array	ro		

6.3.6 Manufacturer Specific Area

Index	Sub-Index	Name	Data type	R/W	Example in Chapter
2000_h	64	Module type of node 1 - 64	uint 8	ro	4.4.9.1
2010_h	64	Module status of node 1 - 64	unit 8	ro	4.4.9.2
2011_h	64	Module configuration of the rotary switches	unit 8	ro	4.4.9.3
2100_h	64	Output voltage	unit 16	ro	4.4.9.4
2101_h	64	Input voltage	unit 16	ro	4.4.9.5
2104_h	64	Current measurement	unit 16	ro	4.4.9.6
$210A_h$	64	Characteristic Curve Setting	unit 16	ro	4.4.9.7

2200$_h$	64	Software version	unit 32	ro	4.4.9.8
2201$_h$	64	Serial number	unit 32	ro	4.4.9.9
2202$_h$	64	LOCC-Box counter “Operation voltage ON“	unit 32	ro	4.4.9.10
2203$_h$	64	LOCC-Box counter “Operation hours (h)“	unit 32	ro	4.4.9.11
2204$_h$	64	LOCC-Box counter “Operation hours ON (h)“	unit 32	ro	4.4.9.12
2205$_h$	64	LOCC-Box counter “Blown“	unit 32	ro	4.4.9.13
2206$_h$	64	LOCC-Box counter “Switching ON“	unit 32	ro	4.4.9.14
2400$_h$	64	Cycle Time of the query in ms	unit 32	r/w	-

Sub-Index = Node number of the connected LOCC-Box

6.3.7 Standardized Profile Area

Index	Name	Data type	R/W	Example in Chapter
6000$_h$	Module status ON / OFF - Byte wise	uint 8	ro	4.4.8.1

6.3.8 Alarm Message, Emergency

If a connected LOCC-Box breaks down during operation, the LOCC-Box-GWEC sends an emergency request according to ETG.1000.6.S (R) V1.0.2. In this case the error code = 0xFF00, error register = 0x80, Data [0]=node number of the failed LOCC-Box, Data [1] = Data [2]= Data [3]= Data [4]=0.

If the LOCC-Box is operating again, another emergency request is send. Error Code = 0x0000, Error Register = 0x00, Data [0] = Node number of the retriggered LOCC-Box, Data [1] = Data [2]= Data [3]= Data [4]=0.

If one or more LOCC-Boxes broke down and operating again, another emergency request is send. Error Code=0x0000, Error Register = 0x00, Data [0]= Data [1] = Data [2]= Data [3]= Data [4]=0.
If a node number between 1 and 64 is added during operation a emergency request is send. Error Code=0x0000, Error Register = 0x00, Data [0]= Node number of the added LOCC-Box, Data [1] = Data [2] = Data [3] = Data [4]=0.

If a new LOCC-Box with a node number between 1 and 64 is added to the system an emergency request is send. Error Code=0x0000, Error Register=0x00, Data[0]=Node Number of the added LOCC-Box, Data[1]=Data[2]=Data[3]=Data[4]=0.

6.3.9 Process Image

For every connected LOCC-Box one I/O Module is used. There are two different options:

- LOCC-Box State with 1 Byte Input
- LOCC-Box Mode with 1 Byte Output

In the Input-Byte (State) the module status of the connected LOCC-Box is displayed. The module status corresponds to command 4 at the RS232 communication (*See chapter 4.3.1*) or Index 0x2010 at the CANopen Communication (*See Chapter 4.4.9.2*).

7	6	5	4	3	2	1	0
System Error	Short Circuit	Under voltage U<10V	I-Warning (I>0,9 * Inom)	New Module on the bus	Reserved	Status	

The output-byte (mode) is for switching the module on and off. The low-order bits 0 and 1 are used.
Bit 0 = 0: switching off the connected LOCC-Box
= 1: switching on the connected LOCC-Box

Bit 1: Changing slope from 0 to 1: Status of 0 Bit accepted from the connected LOCC-Box.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	Rising edge = take over	New status

NOTICE

The data are transmitted in the hex-format.
Via sending ***00*** and connecting ***03*** the device is switched on.
Via sending ***00*** and connecting ***02*** the device is switched off.

6.4 Technical Data

General Data

Nominal voltage	DC 12/24V
Voltage range	DC 10 – 32V
Nominal current	Max. 55mA
Reverse voltage protection	Yes
Terminal	Spring terminal
Housing material	PA 6.6 (UL 94 V0)
Mounting	Snap on TS 35 (according to EN 50022)
Protection level	IP 20
Mounting position	any
Installation technology	Spring terminal 0,25mm^2 – 2,5mm^2 all types of wire until 2,5mm² without end sleeve, until 1,5mm² with end sleeve
USB	USB 2.0 Full-Speed (12 Mbit/s)
EtherCAT	100 Mbit/s
Operation temperature	-20°C to +60°C
Store temperature	-40°C to +85°C
Relative humidity	Max. 90%, non condensing
Dimension (WxHxD)	22,5 x 99 x 114,5mm
Weight	0,130 kg
Approvals	CE
Standards	EN 60950; EN 61131; EN 61000, EN 55016

LOCC-BUS

Access method	Single-Master - Multiple Slave
Bus technology	Line
Physical layer	1-wire
Subscriber	Typical 40, max. 64
Bus length	Typical 10m, max. 40m
Baud rate	9600 Baud
Date rate	8 Bit + fixed parity
Transmission protocol	Modifiziertes multidrop

7 Accessories

For the LOCC-Box-Net we offer a wide range of accessories:

Accessories	Part.-no.	Type	PU
Module			
Supply terminal with cut out of the copper bar for current increase	716421	LOCC-Box-EKL 7-6421	2
Distance terminal without contacts	716422	LOCC-Box-DKL 7-6422	2
LOCC-Box housing without terminals	716424	LOCC-Box-DY 7-6424	2
Supply set (supply- and end terminal)	716425	LOCC-Box-ES 7-6425	1
Gateway (USB, EtherCAT)	716456	LOCC-Box-GW-EC 0-6456	1
Gateway (USB, Profinet)	716457	LOCC-Box-GW-PN 0-6459	1
Gateway (USB, CANopen, RS232)	716459	LOCC-Box-GW 7-6459	1
Jumper combs			
Jumper comb 8pole, 6A, white	716428	LOCC-Box-BKW 7-6428	5
Jumper comb 8pole, 6A, red	716429	LOCC-Box-BKR 7-6429	5
Jumper comb 8pole, 6A, blue	716430	LOCC-Box-BKB 7-6430	5
Jumper comb 16pole, 6A, white	716438	LOCC-Box-BKW 7-6438	5
Jumper comb 16pole, 6A, red	716439	LOCC-Box-BKW 7-6439	5
Jumper comb 16pole, 6A, blue	716440	LOCC-Box-BKW 7-6440	5
Description plates			
Description plates 5x5mm , 200 pieces, white	716431	LOCC-Box-BZW 7-6431	1
Description plates 5x5mm , 200 pieces, red	716432	LOCC-Box-BZR 7-6432	1
Description plates 5x5mm , 200 pieces, blue	716433	LOCC-Box-BZB 7-6433	1
Description plates 5x5mm , 200 pieces, yellow	716434	LOCC-Box-BZG 7-6434	1
Description plates 12x6mm ,160 pieces, white	716441	LOCC-Box-BZW 7-6441	1
Description plates 39,3x8mm, white	716443	LOCC-Box-BZT 7-6443	20
Cover for 716443, transparent	716444	LOCC-Box-BAD 7-6444	20
A4 description sheet for 716443	716445	LOCC-Box-LEB 7-6445	240
Miscellaneous			
Copper bar 1m	716426	LOCC-Box-CU 7-6426	1
Cover for copper bar 1m	716427	LOCC-Box-AD 7-6427	1

8 Trouble shooting

8.1 “Error message during the start phase “

The Software LOCC-Pads requires the ***Microsoft. NET Framework3.0***. In case this version is not installed on the computer, following error message will prompt during the start phase of LOCC-Pads:

Please you would drive through in this case with its PC a Windows update. Start Internet Explorer and would drive you out under the rider extra “Windows Update “. Follow the instructions. Start the PC again after update.

8.2 “No Gateway“ by CANopen

Click **Extra > COM Config** to adjust the USB-Comport. If the message ***No Gateway*** prompts, it can have following reasons:

1. **USB-connection incorrectly**

   - Check and delete the USB connection
   - Shut down LOCC-Pads
   - Reset the supply voltage at the gateway
   - Establish the USB connection
   - Restart LOCC-Pads and repeat the adjustment

2. **The Device Driver is not installed or not correct**

   The device driver has to be installed during the initial operation (see chapter 4.1.6). Check if the driver has been installed.
   Open ***Start>Settings>Control panel.*** Open the folder ***System***. Click ***Hardware*** and open the ***Device Manager.***
   Under ***Other devices*** all connected devices are listed. Devices, which are not installed or which are installed wrong, can be identified by a question mark.

   In this case it is necessary to update the driver. Proceed as follows:

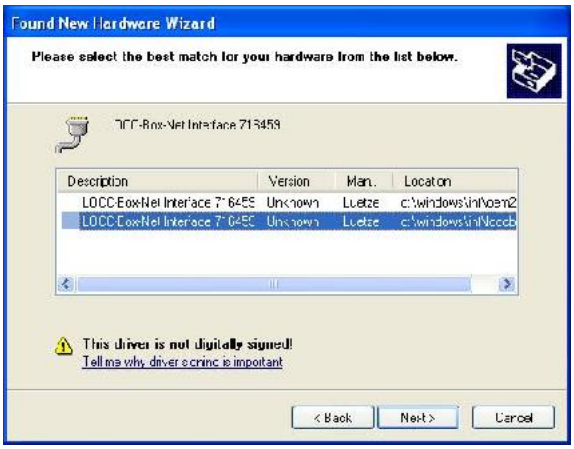
   - Click on the button ***Updating Driver***.

   - The ***Hardware Update Assistant*** prompts.

- Select ***Software automatic installation (recommended)*** and confirm by clicking ***Next***.

- In case more drivers are considered, these are displayed in an additional window. Please select the driver ***loccbn.inf*** and confirm by clicking ***Next***.

- Confirm the appearing safety message by clicking ***Continue Anyway***.



8.3 “No Gateway“ by Profinet

For the setting of the USB-comport click ***Extra > COM Config***. The message ***No Gateway*** can have different causes:

1. **USB-connection incorrectly**

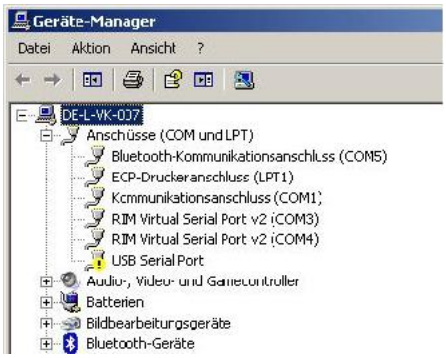
   - Check and delete the USB connection
   - Shut down LOCC-Pads
   - Reset the supply voltage at the gateway
   - Establish the USB connection
   - Restart LOCC-Pads and repeat the adjustment

2. **The Device Driver is not installed or not correct**

Open **Start / Control Panel / Hardware and Sound.**

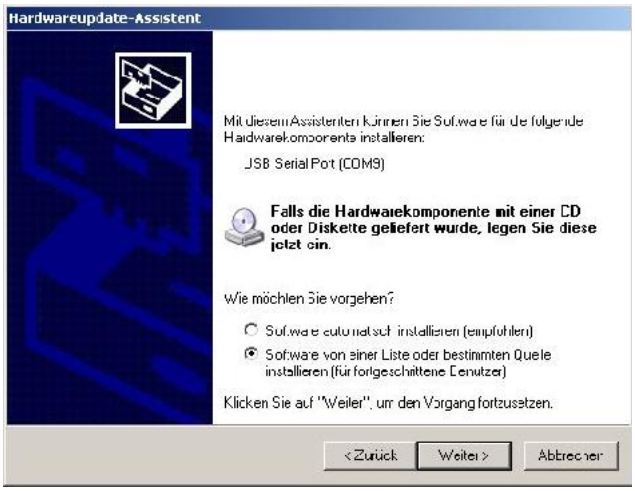
Click under ***Devices and Printer*** on the ***Device Manager***.

If not knowing which connector is the right one, connect and reconnect the USB cable. Mark the connection by clicking the right mouse button. Choose ***Properties.***
Choose the slide ***Driver*** in the property window. Click the Button ***Refresh***. The Hardware Update Assistant opens.



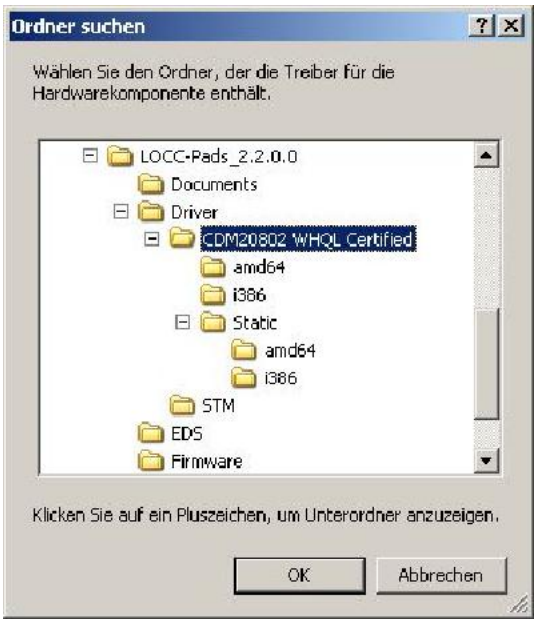
- Choose ***No, not this time.***
  Click ***next.***

- Choose ***Installing software from a list or source*** (for intermediate users).
  Click ***Next***.



- Choose the option search and installation as shown in the picture.
  Click Durchsuchen to define the driver source.

  ➔ The driver can be found in the extracted LOCC-Pads.zip file



8.4 “Lost LOCC-Box-Gateway Connection“

This message appears if there is no communication or if a short interruption of the communication between the Gateway and the computer has occurred. This could have following reasons:

- Under voltage of the DC 12/24V supply
- Malfunctioning connection of the supply voltage (wiring, loose contact)
- Malfunctioning USB connection (loose contact, defective USB cable)

After acknowledging, it is possible to save all data which have been received until that moment. For a restart the comport has to be adjusted again (*see chapter 3.4.1*) or the search has to be restarted (*see chapter 3.5.1*).

Scope of the document

The gateways may include software licensed by 3rd parties. The following third party intellectual property (IP) notices are provided to comply with the terms of such licenses.

Lütze Gateway 716459 Firmware

The firmware of the Gateway uses the FreeRTOSTM operating system which is developed under the terms of the GPL. As a special exception to the GPL, the copyright holder of FreeRTOS gives the permission to link FreeRTOS with independent modules that communicate with FreeRTOS solely through the FreeRTOS API interface, regardless of the license terms of these independent modules, and to copy and distribute the resulting combined work without being obliged to provide the source code of these proprietary modules. See the licensing section of h ttp://www.freeRTOS.org for full details.

1) Windows, Windows 2000, Windows XP und Windows Vista are registered trademarks of the Microsoft Corporation.

2) Program is based in part on the work of the Qwt project (http://qwt.sf.net).