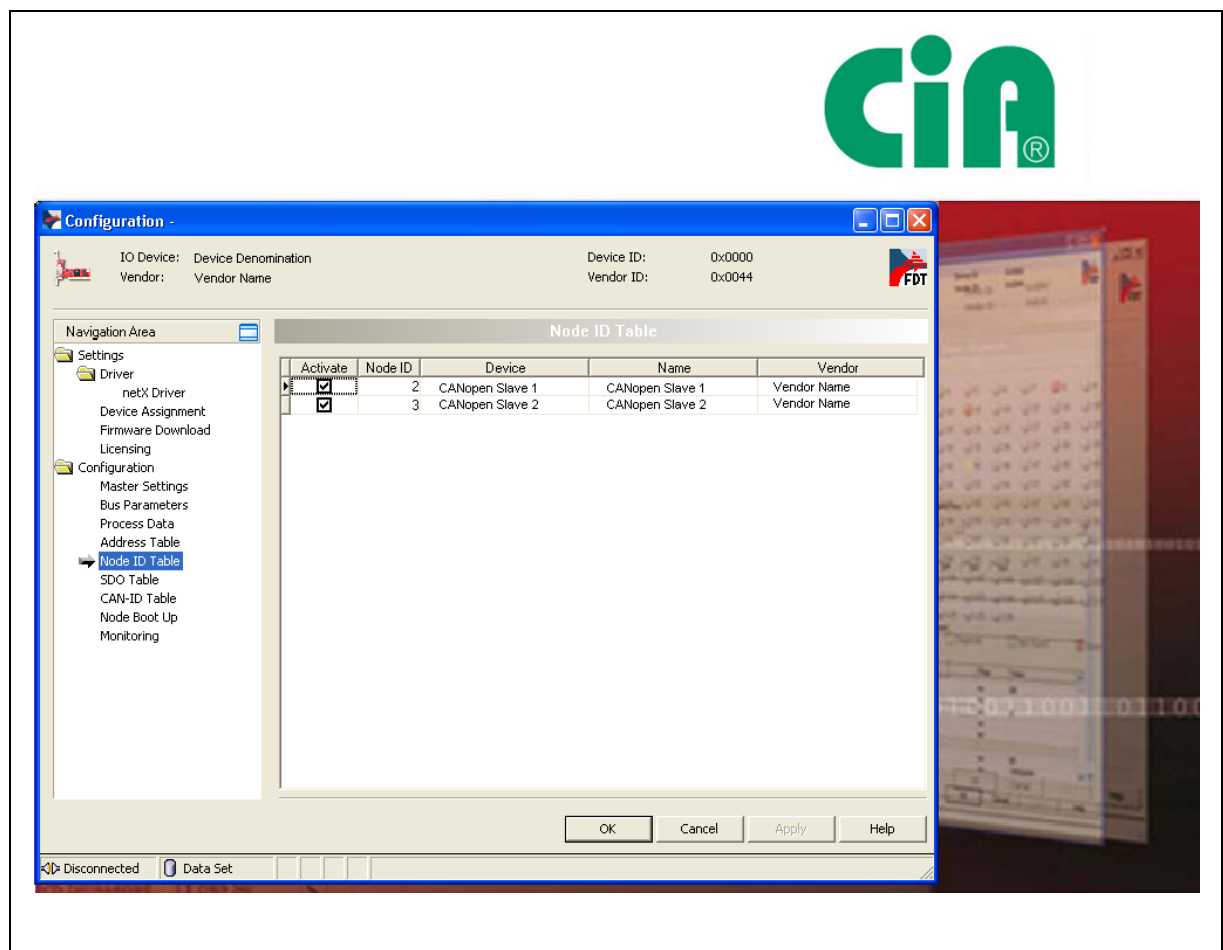


Operating Instruction Manual

DTM for Hilscher CANopen Master Devices

Configuration of Hilscher Master Devices

V1.1000



Hilscher Gesellschaft für Systemautomation mbH

www.hilscher.com

DOC070402OI17EN | Revision 17 | English | 2018-03 | Released | Public

Table of Contents

1	INTRODUCTION.....	6
1.1	About this Manual	6
1.1.1	Descriptions of the Dialog Panes	6
1.1.2	Online Help	7
1.1.3	List of Revisions	7
1.1.4	Conventions in this Manual	8
1.2	Legal Notes.....	9
1.3	Registered Trademarks.....	12
1.4	About CANopen Master DTM.....	12
1.4.1	Requirements	13
1.5	Dialog Structure CANopen Master DTM.....	14
1.5.1	General Device Information.....	15
1.5.2	Navigation Area	15
1.5.3	Dialog Panes	16
1.5.4	OK, Cancel, Apply and Help.....	17
1.5.5	Table Lines	17
1.5.6	Status Bar	18
2	SAFETY	19
2.1	General Note.....	19
2.2	Intended Use.....	19
2.3	Personnel Qualification	19
2.4	Safety Instructions.....	20
2.4.1	Communication Stop during Firmware or Configuration Download	20
2.4.2	Mismatching System Configuration	20
2.5	Property Damage	21
2.5.1	Power Disconnect while downloading Firmware or Configuration	21
2.5.2	Invalid Firmware	21
2.6	Labeling of Safety Messages	22
2.7	References Safety.....	22
3	GETTING STARTED.....	23
3.1	Configuration Steps.....	23
3.2	Safety Messages on Firmware or Configuration Download	27
4	SETTINGS	28
4.1	Overview Settings	28
4.2	Settings for Driver and Device Assignment	29
4.3	Driver	31
4.3.1	Verify or adapt Driver Settings	31

4.3.2	cifX Device Driver	33
4.3.3	netX Driver	33
4.3.4	Configuring netX Driver	34
4.3.5	netX Driver - USB/RS232 Connection	35
4.3.6	netX Driver - TCP/IP Connection	37
4.4	Device Assignment	40
4.4.1	Scanning for Devices	40
4.4.2	Selecting the Device (with or without firmware)	43
4.4.3	Selecting the Device once more (with Firmware)	44
4.5	Firmware Download	46
4.6	Licensing	53
5	CONFIGURATION	54
5.1	Overview Configuration	54
5.2	Configuring Device Parameters	55
5.3	Master Settings	57
5.3.1	Application Monitoring	58
5.3.2	Process Image Storage Format	59
5.3.3	Module Alignment	59
5.3.4	Process Data Handshake	60
5.3.5	Advanced (For future Application)	60
5.3.6	Device Status Offset	61
5.4	Bus Parameters	62
5.4.1	Device Description	63
5.4.2	Node Settings	63
5.4.3	Sync Master Settings	64
5.4.4	29 Bit COB-ID	64
5.5	Process Data	66
5.6	Address Table	67
5.6.1	Description of the Parameters	68
5.6.2	Display Mode, CSV Export	68
5.7	Node ID Table	69
5.8	SDO Table	70
5.8.1	Mapping Method for PDO and PDO-Mapping	71
5.9	CAN-ID Table	72
5.10	Node BootUp Mapping Sequence	73
5.11	Monitoring	75
5.11.1	Node Guarding Protocol	76
5.11.2	Heartbeat Protocol	77
6	ONLINE FUNCTIONS	78
6.1	Connecting/Disconnecting Device	78
6.2	Debug Mode	80
6.2.1	Requirements	81
6.2.2	Starting Debug Mode	82

6.2.3	Debug Mode Busline Colors and Debug Icons	83
6.2.4	Reset of the Diagnosis Information and of the Station Status	84
6.2.5	Stopping Debug Mode.....	84
6.3	'Network Scan'	85
6.3.1	Starting 'Network Scan'	86
6.3.2	Settings in the Scan Response Dialog of the Master DTM.....	88
6.3.3	Description on the Scan Response Dialog of the Master DTM.....	89
6.3.4	Creating Devices	91
6.3.5	Download to the CANopen Master Device.....	91
6.4	Download Configuration	92
6.5	Start /Stop Communication	94
6.6	Licensing.....	95
6.6.1	Open License Dialog	95
6.6.2	License Dialog	96
6.6.3	Which Licenses are present in the Device?	97
6.6.4	How to order a License.....	99
6.6.5	Selecting License(s)	99
6.6.6	Ordering Data	100
6.6.7	Ordering the License	102
6.6.8	How to get the License and transfer it to the Device.....	107
7	DIAGNOSIS	108
7.1	Overview Diagnosis	108
7.2	General Diagnosis.....	109
7.3	Master Diagnosis	111
7.4	Bus Diagnosis	112
7.5	Station Diagnosis	113
7.6	Firmware Diagnosis	114
8	EXTENDED DIAGNOSIS	115
8.1	Overview Extended Diagnosis.....	115
8.2	Task Information	116
8.3	IniBatch Status.....	117
8.4	CAN_DL.....	118
8.4.1	Application Commands.....	118
8.4.2	CAN Driver Status	119
8.5	CANOPEN_MASTER.....	120
8.5.1	Common diagnostic.....	120
8.5.2	CMS domain services.....	121
8.5.3	Node Initializations	122
8.5.4	Node state	122
8.5.5	Commands	123
8.6	CANOPEN_APM.....	125
8.6.1	Master Configuration	125
8.6.2	Commands	126

9	TOOLS	127
9.1	Overview Tools	127
9.2	Packet Monitor	128
9.2.1	Sending Packet	129
9.2.2	Receiving Packet.....	130
9.2.3	Example – Reading Data via SDO Upload.....	131
9.2.4	Example – Writing Data via SDO Download	138
9.3	IO Monitor	145
9.4	Process Image Monitor	146
10	ERROR CODES.....	148
10.1	Error Code Definition.....	148
10.2	Overview Error Codes	149
10.3	General Hardware Error Codes	150
10.3.1	RCX General Task Errors.....	150
10.3.2	RCX Common Status & Errors Codes	151
10.3.3	RCX Status & Error Codes	152
10.4	ODM Error Codes	153
10.4.1	General ODM Error Codes	153
10.4.2	General ODM Driver Error Codes	154
10.4.3	cifX Driver Specific ODM Error Codes	155
10.5	Error Codes cifX Device Driver and netX Driver	158
10.5.1	Generic Error Codes.....	158
10.5.2	Generic Driver Error Codes	159
10.5.3	Generic Device Error Codes	160
10.6	Error Codes netX Driver	161
10.6.1	CIFX API Transport Error Codes.....	161
10.6.2	CIFX API Transport Header State Error Codes	161
10.7	ODM Error Codes DBM V4	162
11	APPENDIX	166
11.1	User Rights	166
11.1.1	Settings.....	166
11.1.2	Configuration	167
11.2	References.....	167
11.3	List of Figures	168
11.4	List of Tables.....	170
11.5	Glossary.....	172
11.6	Contacts.....	174

1 Introduction

1.1 About this Manual

This manual provides information on how to set and configure the device parameters of a netX based CANopen Master device using the CANopen Master DTM, and what can be read from the diagnosis panes.

1.1.1 Descriptions of the Dialog Panes

The table below gives an overview for the individual dialog panes descriptions:

Section	Subsection	Page
<i>Settings</i>	<i>Overview Settings</i>	28
	<i>Driver</i>	31
	<i>Device Assignment</i>	40
	<i>Firmware Download</i>	46
	<i>Licensing</i>	95
<i>Configuration</i>	<i>Overview Configuration</i>	54
	<i>Master Settings</i>	57
	<i>Bus Parameters</i>	62
	<i>Process Data</i>	66
	<i>Address Table</i>	67
	<i>Node ID Table</i>	69
	<i>SDO Table</i>	70
	<i>CAN-ID Table</i>	72
	<i>Node BootUp Mapping Sequence</i>	73
	<i>Monitoring</i>	75
<i>Diagnosis</i>	<i>Overview Diagnosis</i>	108
	<i>General Diagnosis</i>	109
	<i>Master Diagnosis</i>	111
	<i>Bus Diagnosis</i>	112
	<i>Station Diagnosis</i>	113
	<i>Firmware Diagnosis</i>	114
<i>Extended Diagnosis</i>	<i>Overview Extended Diagnosis</i>	115
<i>Tools</i>	<i>Overview Tools</i>	127
	<i>Packet Monitor</i>	128
	<i>IO Monitor</i>	145
	<i>Process Image Monitor</i>	146

Table 1: Descriptions Dialog Panes

1.1.2 Online Help

The CANopen Master DTM contains an integrated online help facility.

- To open the online help, click on **Help** or press **F1**.

1.1.3 List of Revisions

Index	Date	Version	Chapter	Revision
16	17-02-23	1.1000	1.4.1	Section <i>Requirements</i> Internet access added, Windows 8.1 and Windwos 10 added.
17	18-02-14	1.1000	2.5.1, 3.2, 4.5, 6.4	Versioning information revised: Title page and this section Safety Message or warning added: Sections <i>Power Disconnect while downloading Firmware or Configuration</i> , <i>Safety Messages on Firmware or Configuration Download</i> , <i>Firmware Download</i> and <i>Download Configuration</i> . Safety communication in the document revised in general.

Table 2: List of Revisions

1.1.4 Conventions in this Manual

Notes, operation instructions and results of operation steps are marked as follows:

Notes



Important: <important note you must follow to avoid malfunction>



Note: <general note>



<note, where to find further information>

Operation Instructions

1. <instruction>

2. <instruction>

or

➤ <instruction>

Results

↪ <result>

Positions in Figures

The *Positions* ①, ②, ③ ... or a, b, c ... or A, B, C ... refer to the figure used in that section. If the numbers reference to a section outside the current section then a cross reference to that section and figure is indicated.

1.2 Legal Notes

Copyright

© Hilscher Gesellschaft für Systemautomation mbH

All rights reserved.

The images, photographs and texts in the accompanying materials (in the form of a user's manual, operator's manual, Statement of Work document and all other document types, support texts, documentation, etc.) are protected by German and international copyright and by international trade and protective provisions. Without the prior written consent, you do not have permission to duplicate them either in full or in part using technical or mechanical methods (print, photocopy or any other method), to edit them using electronic systems or to transfer them. You are not permitted to make changes to copyright notices, markings, trademarks or ownership declarations. Illustrations are provided without taking the patent situation into account. Any company names and product designations provided in this document may be brands or trademarks by the corresponding owner and may be protected under trademark, brand or patent law. Any form of further use shall require the express consent from the relevant owner of the rights.

Important notes

Utmost care was/is given in the preparation of the documentation at hand consisting of a user's manual, operating manual and any other document type and accompanying texts. However, errors cannot be ruled out. Therefore, we cannot assume any guarantee or legal responsibility for erroneous information or liability of any kind. You are hereby made aware that descriptions found in the user's manual, the accompanying texts and the documentation neither represent a guarantee nor any indication on proper use as stipulated in the agreement or a promised attribute. It cannot be ruled out that the user's manual, the accompanying texts and the documentation do not completely match the described attributes, standards or any other data for the delivered product. A warranty or guarantee with respect to the correctness or accuracy of the information is not assumed.

We reserve the right to modify our products and the specifications for such as well as the corresponding documentation in the form of a user's manual, operating manual and/or any other document types and accompanying texts at any time and without notice without being required to notify of said modification. Changes shall be taken into account in future manuals and do not represent an obligation of any kind, in particular there shall be no right to have delivered documents revised. The manual delivered with the product shall apply.

Under no circumstances shall Hilscher Gesellschaft für Systemautomation mbH be liable for direct, indirect, ancillary or subsequent damage, or for any loss of income, which may arise after use of the information contained herein.

Liability disclaimer

The hardware and/or software was created and tested by Hilscher Gesellschaft für Systemautomation mbH with utmost care and is made available as is. No warranty can be assumed for the performance or flawlessness of the hardware and/or software under all application

conditions and scenarios and the work results achieved by the user when using the hardware and/or software. Liability for any damage that may have occurred as a result of using the hardware and/or software or the corresponding documents shall be limited to an event involving willful intent or a grossly negligent violation of a fundamental contractual obligation. However, the right to assert damages due to a violation of a fundamental contractual obligation shall be limited to contract-typical foreseeable damage.

It is hereby expressly agreed upon in particular that any use or utilization of the hardware and/or software in connection with

- Flight control systems in aviation and aerospace;
- Nuclear fusion processes in nuclear power plants;
- Medical devices used for life support and
- Vehicle control systems used in passenger transport

shall be excluded. Use of the hardware and/or software in any of the following areas is strictly prohibited:

- For military purposes or in weaponry;
- For designing, engineering, maintaining or operating nuclear systems;
- In flight safety systems, aviation and flight telecommunications systems;
- In life-support systems;
- In systems in which any malfunction in the hardware and/or software may result in physical injuries or fatalities.

You are hereby made aware that the hardware and/or software was not created for use in hazardous environments, which require fail-safe control mechanisms. Use of the hardware and/or software in this kind of environment shall be at your own risk; any liability for damage or loss due to impermissible use shall be excluded.

Warranty

Hilscher Gesellschaft für Systemautomation mbH hereby guarantees that the software shall run without errors in accordance with the requirements listed in the specifications and that there were no defects on the date of acceptance. The warranty period shall be 12 months commencing as of the date of acceptance or purchase (with express declaration or implied, by customer's conclusive behavior, e.g. putting into operation permanently).

The warranty obligation for equipment (hardware) we produce is 36 months, calculated as of the date of delivery ex works. The aforementioned provisions shall not apply if longer warranty periods are mandatory by law pursuant to Section 438 (1.2) BGB, Section 479 (1) BGB and Section 634a (1) BGB [Bürgerliches Gesetzbuch; German Civil Code] If, despite of all due care taken, the delivered product should have a defect, which already existed at the time of the transfer of risk, it shall be at our discretion to either repair the product or to deliver a replacement product, subject to timely notification of defect.

The warranty obligation shall not apply if the notification of defect is not asserted promptly, if the purchaser or third party has tampered with the products, if the defect is the result of natural wear, was caused by unfavorable operating conditions or is due to violations against our operating regulations or against rules of good electrical engineering

practice, or if our request to return the defective object is not promptly complied with.

Costs of support, maintenance, customization and product care

Please be advised that any subsequent improvement shall only be free of charge if a defect is found. Any form of technical support, maintenance and customization is not a warranty service, but instead shall be charged extra.

Additional guarantees

Although the hardware and software was developed and tested in-depth with greatest care, Hilscher Gesellschaft für Systemautomation mbH shall not assume any guarantee for the suitability thereof for any purpose that was not confirmed in writing. No guarantee can be granted whereby the hardware and software satisfies your requirements, or the use of the hardware and/or software is uninterrupted or the hardware and/or software is fault-free.

It cannot be guaranteed that patents and/or ownership privileges have not been infringed upon or violated or that the products are free from third-party influence. No additional guarantees or promises shall be made as to whether the product is market current, free from deficiency in title, or can be integrated or is usable for specific purposes, unless such guarantees or promises are required under existing law and cannot be restricted.

Confidentiality

The customer hereby expressly acknowledges that this document contains trade secrets, information protected by copyright and other patent and ownership privileges as well as any related rights of Hilscher Gesellschaft für Systemautomation mbH. The customer agrees to treat as confidential all of the information made available to customer by Hilscher Gesellschaft für Systemautomation mbH and rights, which were disclosed by Hilscher Gesellschaft für Systemautomation mbH and that were made accessible as well as the terms and conditions of this agreement itself.

The parties hereby agree to one another that the information that each party receives from the other party respectively is and shall remain the intellectual property of said other party, unless provided for otherwise in a contractual agreement.

The customer must not allow any third party to become knowledgeable of this expertise and shall only provide knowledge thereof to authorized users as appropriate and necessary. Companies associated with the customer shall not be deemed third parties. The customer must obligate authorized users to confidentiality. The customer should only use the confidential information in connection with the performances specified in this agreement.

The customer must not use this confidential information to his own advantage or for his own purposes or rather to the advantage or for the purpose of a third party, nor must it be used for commercial purposes and this confidential information must only be used to the extent provided for in this agreement or otherwise to the extent as expressly authorized by the disclosing party in written form. The customer has the right, subject to the obligation to confidentiality, to disclose the terms and conditions of this agreement directly to his legal and financial consultants as would be required for the customer's normal business operation.

Export provisions

The delivered product (including technical data) is subject to the legal export and/or import laws as well as any associated regulations of various countries, especially such laws applicable in Germany and in the United States. The products / hardware / software must not be exported into such countries for which export is prohibited under US American export control laws and its supplementary provisions. You hereby agree to strictly follow the regulations and to yourself be responsible for observing them. You are hereby made aware that you may be required to obtain governmental approval to export, reexport or import the product.

1.3 Registered Trademarks

Windows® XP, Windows® Vista, Windows® 7, Windows® 8, Windows® 8.1 and Windows® 10 are registered trademarks of Microsoft Corporation.

CANopen® is a registered trademark of CAN in AUTOMATION - International Users and Manufacturers Group e.V (CiA), Nürnberg.

CODESYS® is a registered trademark of 3S-Smart Software Solutions GmbH, Deutschland.

All other mentioned trademarks are property of their respective legal owners.

1.4 About CANopen Master DTM

You can use the CANopen Master DTM to configure the CANopen Master device within a FDT Framework.



Important: For a 2-channel device consecutively channel 1 or channel 2 must be assigned to the DTM and each must be configured individually.

1.4.1 Requirements

System Requirements

- PC with 1 GHz processor or higher
- Windows® XP SP3,
Windows® Vista (32-Bit) SP2,
Windows® 7 (32-Bit and 64-Bit) SP1,
Windows® 8 (32-Bit and 64-Bit),
Windows® 8.1 (32-Bit and 64-Bit),
Windows® 10 (32-Bit and 64-Bit)
- Administrator privilege required for installation
- Internet Explorer 5.5 or higher
- RAM: min. 512 MByte, recommended 1024 MByte
- Graphic resolution: min. 1024 x 768 pixel
- Keyboard and Mouse
- Restriction: Touch screen is not supported.



Note: If the project file is used on a further PC,

- this PC must also comply with the above system requirements,
- the device description files of the devices used in the project must be imported into the configuration software SYCON.net on the new PC,
- and the DTMs of the devices used in the project must also be installed on that further PC.

Requirements CANopen Master DTM

To configure the CANopen Master device with the CANopen Master DTM the following requirements have to be accomplished:

- Completed hardware installation of a netX based DTM-compatible CANopen Master device, inclusive loaded firmware, license and loaded cifX configuration file
- Installed FDT/DTM V 1.2 compliant frame application
- Loaded DTM in the Device Catalog of the FTD Framework



Note: If the CANopen Master DTM and the CANopen Master device are installed on the same PC, the **cifX Device Driver** must be installed on that PC, as you can connect the DTM to the device.



For more information to the hardware installation, please refer to the corresponding **User Manual** of your device.

To get information on how to order and to download the license to the device, please refer to section *Licensing* on page 95.

1.5 Dialog Structure CANopen Master DTM

The graphical user interface of the DTM is composed of different areas and elements listed hereafter:

1. A header area containing the **General Device Information**,
2. The **Navigation Area** (area on the left side),
3. The **Dialog Pane** (main area on the right side),
4. **OK, Cancel, Apply, Help**,
5. The **Status Line** containing information e. g. the online-state of the DTM.

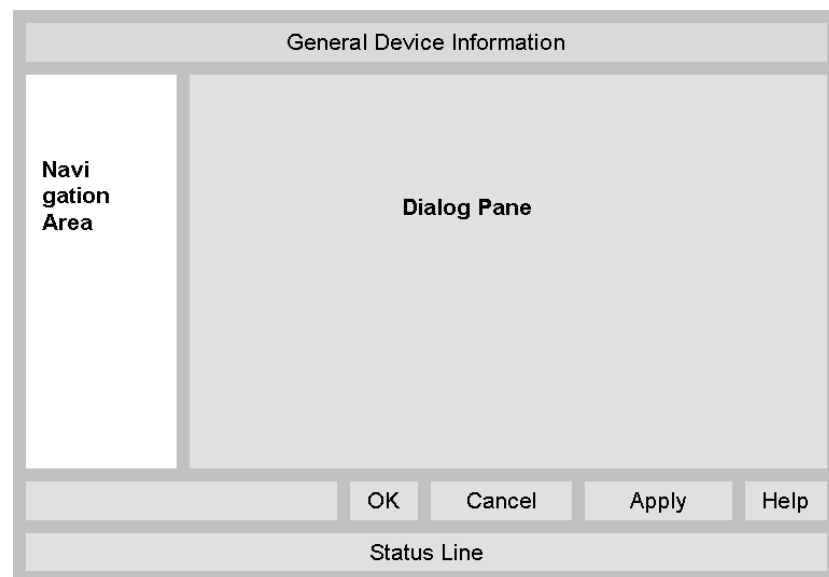


Figure 1: Dialog Structure of the CANopen Master DTM

1.5.1 General Device Information

Parameter	Meaning
IO Device	Name of the device
Vendor	Vendor name of the device
Device ID	Identification number of the device
Vendor ID	Identification number of the vendor

Table 3: General Device Information

1.5.2 Navigation Area

The **Navigation Area** contains folders and subfolders to open the dialog panes of the DTM.

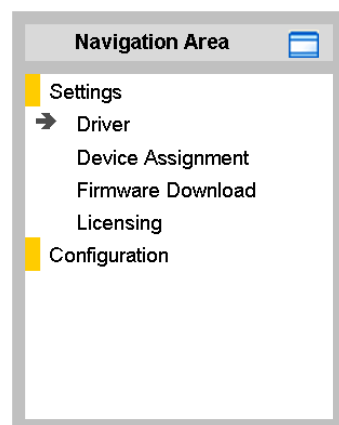


Figure 2: Navigation Area

- Select the required folder and subfolder.
- The corresponding Dialog pane is displayed.

Hide / display Navigation

	Hiding the navigation area (above right side).
Show navigation area	Opening the navigation area (below left side).

1.5.3 Dialog Panes

At the dialog pane the **Settings**, **Configuration**, **Diagnosis/Extended Diagnosis** or the **Tools** panes are opened via the corresponding folder in the navigation area.

Settings	
Driver	To establish a connection from the CANopen Master DTM to the CANopen Master device, on the pane Driver you can verify if the default driver is checked and respectively check another driver or multiple drivers. For further information, refer to section <i>Driver</i> on page 31.
Device Assignment	On the pane Device Assignment you select the device and assign the device to the driver. For further information, refer to section <i>Device Assignment</i> on page 40.
Firmware Download	The dialog on the pane Firmware Download is used to load a new firmware into the device. A detailed description can be found in section <i>Firmware Download</i> on page 46.
Licensing	Using the license dialog, you can order licenses for Master protocols or Utilities and download them to your device. A detailed description can be found in section <i>Licensing</i> on page 95.
Configuration	
Master Settings	At the Master Settings pane device related settings can be made. For further information, refer to section <i>Master Settings</i> on page 57.
Bus Parameter	The Bus Parameters are the basis of an operating data exchange. For further information see section <i>Bus Parameters</i> on page 62.
Process Data	The Process Data pane serves for the CANopen Master DTM as an external process data interface. For further information, refer to section <i>Process Data</i> on page 66.
Address Table	The Address Table shows a list of all addresses used in the process data image. For further information, refer to section <i>Address Table</i> on page 67.
Node ID Table	The Node ID Table shows a list of all configured Nodes of the Master. For further information, refer to section <i>Node ID Table</i> on page 69.
SDO Table	The SDO Table shows an overview of the transmitted objects during the Node boot up phase for each Node. For further information, refer to section <i>SDO Table</i> on page 70.
CAN-ID Table	The CAN-ID Table displays a list for each Node which message numbers in the CAN network are occupied by the respective Node. For further information, refer to section <i>CAN-ID Table</i> on page 72.
Node BootUp	The Node BootUp defines the start up behavior of the Master with regard to each individual node and is described in section. For further information, refer to section <i>Node BootUp Mapping Sequence</i> on page 73.
Monitoring	The function Monitoring is used to configure the monitoring between different Nodes. For further information, refer to section <i>Monitoring</i> on page 75.
Diagnosis	
Diagnosis/ Extended Diagnosis	At the Diagnosis panes information can be read for troubleshooting. For further information, refer to section <i>Overview Diagnosis</i> on page 108 or section <i>Overview Extended Diagnosis</i> on page 115.
Tools	
Packet Monitor/ IO Monitor/ Process Image Monitor	Under Tools the Packet Monitor and the IO Monitor are provided for test and diagnosis purposes. For further information, refer to section <i>Packet Monitor</i> on page 128, section <i>IO Monitor</i> on page 145 or section <i>Process Image Monitor</i> on page 146.

Table 4: Overview Dialog Panes



Note: Accessing the **Diagnosis** panes of the CANopen Master DTM requires an online connection from the CANopen Master DTM to the CANopen Master device.



For further information, refer to section *Connecting/Disconnecting Device* on page 78.

1.5.4 OK, Cancel, Apply and Help

OK, Cancel, Apply and Help you can use as described hereafter.

	Meaning
OK	To confirm your latest settings, click OK . All changed values will be applied on the frame application database. <i>The dialog then closes.</i>
Cancel	To cancel your latest changes, click Cancel . Answer to the safety query Configuration data has been changed. Do you want to save the data? by Yes , No or Cancel . Yes: The changes are saved or the changed values are applied on the frame application database. <i>The dialog then closes.</i> No: The changes are <u>not</u> saved or the changed values are not applied on the frame application database. <i>The dialog then closes.</i> Cancel: <i>Back to the DTM.</i>
Apply	To confirm your latest settings, click Apply . All changed values will be applied on the frame application database. <i>The dialog remains opened.</i>
Help	To open the DTM online help, click Help .

Table 5: OK, Cancel, Apply and Help

1.5.5 Table Lines

In the DTM dialog pane table lines can be selected, inserted or deleted.

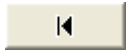
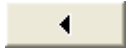

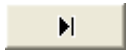


	Meaning
	To select the first line of a table use First Line .
	To select the previous line of a table use Previous Line .
	To select the next line of a table use Next Line .
	To select the last line of a table use Last Line .
	Create a new Line inserts new lines into the table.
	Delete selected Line deletes the selected line from the table.

Table 6: Selecting, inserting, deleting Table Line

1.5.6 Status Bar

The **Status Bar** displays information about the current state of the DTM. The current activity, e.g. download, is signaled graphically via icons in the status bar.

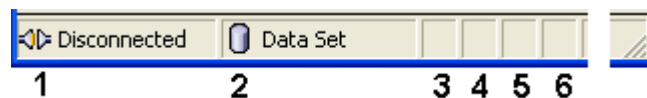
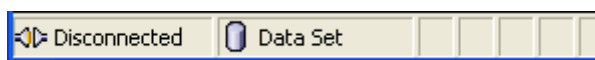


Figure 3: Status Bar – Status Fields 1 to 6

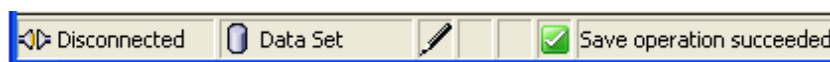
Status Field	Icon / Meaning	
1	DTM Connection States	
		Connected: Icon closed = Device is online
		Disconnected: Icon opened = Device is offline
2	Data Source States	
		Data set: The displayed data are read out from the instance data set (database).
		Device: The displayed data are read out from the device.
3	States of the instance Date Set	
		Valid Modified: Parameter is changed (not equal to data source).
4	Changes directly made on the Device	
		Load/configure diagnosis parameters: Diagnosis is activated.
6	Device Diagnosis Status	
		Save operation succeeded: The save operation has been successful. Further messages due to successful handling of device data.
		Firmware Download: Firmware Download is running
		Save operation failed: The save operation has failed. Further fail operation messages due to incorrect communication due to malfunction in the field device or its peripherals.

Table 7: Status Bar Icons [1]

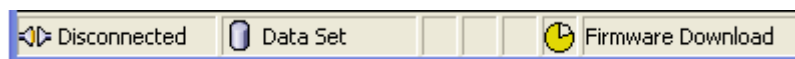
Offline State



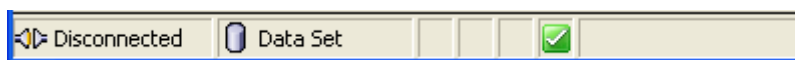
Save operation succeeded



Firmware Download



Firmware Download successful



Online State and Diagnosis

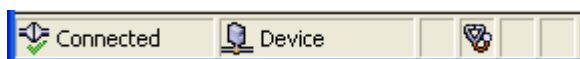


Figure 4: Status Bar Display Examples

2 Safety

2.1 General Note

The documentation in the form of a user manual, an operating instruction manual or other manual types, as well as the accompanying texts have been created for the use of the products by educated personnel. When using the products, all Safety Messages, Integrated Safety Messages, Property Damage Messages and all valid legal regulations must be obeyed. Technical knowledge is presumed. The user has to assure that all legal regulations are obeyed.

2.2 Intended Use

The CANopen Master DTM serves for configuration and diagnosis of CANopen Master devices.

2.3 Personnel Qualification

Personnel responsible for the application of the network system shall be aware of the system behavior and shall be trained in using the system.

2.4 Safety Instructions

To ensure your own personal safety and to avoid personal injury, you necessarily must read, understand, and comply with the safety instructions and safety messages in this manual before you install and operate your system.

For cases if both, personal injury as well as property damage (damage of equipment or device) may occur together, you find the safety instructions in this section.

2.4.1 Communication Stop during Firmware or Configuration Download

If you want to perform either a firmware update (as a download) or a configuration download, both via the CANopen Master DTM, be aware of the following:

- Together with the firmware download, an automated reset to the device is performed that will interrupt all network communication and all established connections will drop.
- If you download the configuration during bus operation, the communication between master and slaves is stopped.

Possible faulty System Operation

- An unpredictable and unexpected behavior of machines and plant components may cause personal injury and property damage.
 - Stop the application program, before starting the firmware update or before downloading the configuration.
 - Make sure that your equipment operates under conditions that prevent personal injury or property damage. All network devices should be placed in a fail-safe mode, before starting the firmware update or before downloading a configuration.

Loss of Device Parameters, Overwriting of Firmware

- Both the firmware download and the configuration download erase the configuration data base. The firmware download overwrites the existing firmware in the network device.
 - To complete the firmware update and to make the device operable again, re-load the configuration after the firmware update has been finished.

2.4.2 Mismatching System Configuration

Mismatching system configuration loaded into the device could result in faulty data mapping in the application program and thus unexpected equipment operation may cause personal injury or damage of equipment.

- In the device use only a configuration suitable for the system.

2.5 Property Damage

To avoid property damage and damage to your system or to your equipment, you necessarily must read, understand, and comply with the safety instructions and safety messages in this manual before you configure your system.

2.5.1 Power Disconnect while downloading Firmware or Configuration

If during the process of downloading a firmware or configuration

- the power supply to a PC with the software application is interrupted,
- or the power supply to the CANopen Master device is interrupted,
- or a reset to the device is performed,

this may lead to the following consequences:

Loss of Device Parameters, Firmware Corruption

- The firmware download or the configuration download will be interrupted and remains incomplete.
- The firmware or the configuration database will be corrupted and device parameters will be lost.
- Device damage may occur as the device cannot be rebooted.

Whether these consequences occur depends on when the power disconnect occurs during the download.

- During configuration download process, do not interrupt the power supply to the PC, or to the device and do not perform a reset!

Otherwise you might be forced to return your device for repair.

Power Drop during Write and Delete Accesses in the File System

The FAT file system in the netX firmware is subject to certain limitations in its operation. Write and delete accesses in the file system (firmware update, configuration download etc.) can destroy the FAT (File Allocation Table) if the accesses cannot be completed if the power drops. Without a proper FAT, a firmware may not be found and cannot be started.

Make sure that the power supply to the device is not interrupted during write and delete accesses in the file system (firmware update, configuration download, etc.).

2.5.2 Invalid Firmware

Loading invalid firmware files could render your device unusable.

- Only download firmware files to the device that are valid for this device.

Otherwise you might be forced to return your device for repair.

2.6 Labeling of Safety Messages

- The **Section Safety Messages** at the beginning of a chapter are pinpointed particularly and highlighted by a signal word according to the degree of endangerment. The type of danger is specified by the safety message text
- The **Integrated Safety Messages** within an instruction description are highlighted with a signal word according to the degree of endangerment. The kind of danger is specified exactly by the safety message text.





Signal Word	Meaning (International)	Meaning (USA)
 DANGER	Indicates a direct hazard with high risk, which will have a consequence of death or grievous bodily harm if it is not avoided.	Indicates a hazardous situation which if not avoided, will result in death or serious injury.
 WARNING	Indicates a possible hazard with medium risk, which will have a consequence of death or (grievous) bodily harm if it is not avoided.	Indicates a hazardous situation which if not avoided, could result in death or serious injury.
 CAUTION	Indicates a minor hazard with medium risk, which could have a consequence of minor or moderate bodily harm if it is not avoided.	Indicates a hazardous situation which if not avoided, may result in minor or moderate injury.
 NOTICE	Indicates a property damage message.	Indicates a property damage message.

Table 8: Signal Words



Note: The ANSI Z535.6 standard specifies in section 4.8: "Messages about hazards that could result in both, physical injury and property damage are considered safety messages, not property damage messages." Thus depending of the type of danger and its consequences, warning messages marked by a signal word DANGER, WARNING or CAUTION may include both, messages on physical injury and property damage.

In this document all Safety Instructions and Safety Messages are designed according both to the international used safety conventions as well as to the ANSI Z535.6 standard, refer to safety reference [S1].

In this document the signal words 'WARNING', 'CAUTION' and 'NOTICE' are used according to ANSI Z535.6 standard. The meaning given in ISO/IEC 26514 [S4] section '11.11 Contents of warnings and cautions' is not relevant in this manual.

2.7 References Safety

- [S1] ANSI Z535.6-2011 American National Standard for Product Safety Information in Product Manuals, Instructions, and Other Collateral Materials
- [S4] 26514-2010 - IEEE Standard for Adoption of ISO/IEC 26514:2008 Systems and Software Engineering--Requirements for Designers and Developers of User Documentation

3 Getting started


3.1 Configuration Steps




The following overview provides to you the step sequence on how to configure a netX based CANopen Master device with CANopen Master DTM as it is typical for many cases. At this time it is presupposed that the hardware installation was done.

The overview lists all the steps in a compressed form. For detailed descriptions of each step refer to the sections noted in the column *For detailed information see section*.



Important: For a 2-channel device consecutively channel 1 or channel 2 must be assigned to the DTM and each must be configured individually.

#	Step	Short Description	For detailed information see section	Page
1	Add Slave in the Device Catalog	Add the Slave in the Device Catalog by importing the device description file to the Device Catalog. Depending of the FDT Container. For netDevice: - Network > Import Device Descriptions.	(See Operating Instruction Manual netDevice and netProject)	-
2	Load device catalog	Depending of the FDT Container: For netDevice: - select Network > Device Catalog , - select Reload Catalog .	(See Operating Instruction Manual netDevice and netProject)	-
3	Create new project / Open existing project	Depending of the frame application. For the configuration software: - select File > New or File > Open .	(See Operating Instruction Manual of the Frame Application)	-
4	Insert Master or Slave into configuration	Depending of the FDT Container: For netDevice:  Important! In order to select the desired device in the device catalog, note the details about the DTM and the device at the bottom of the window. When sorting by <i>Fieldbus</i> multiple devices with identical names by different vendors can be displayed. - in the Device Catalog click to the Master, - and insert the device via drag and drop to the line in the network view, - in the Device Catalog click to the Slave,* - and insert the device via drag and drop to the Master bus line in the network view.* (*This step won't be necessary if the network structure is scanned automatically. See step 17.)	(See Operating Instruction Manual netDevice and netProject)	-
5	Open the Master DTM configuration dialog	Open the Master DTM configuration dialog. - Double click to the device icon of the Master. - The Master DTM configuration dialog is displayed.	-	-

#	Step	Short Description	For detailed information see section	Page
6	Verify or adapt Driver Settings	In the Master DTM configuration dialog: - select Settings > Driver .	<i>Settings for Driver and Device Assignment and Driver</i>	29
		 Note! For PC cards cifX the cifX Device Driver is preset as a default driver. For all the other Hilscher devices the netX Driver is preset as a default driver. <ul style="list-style-type: none">• Use the cifX Device Driver if the CANopen Master DTM is installed on the same PC as the CANopen Master device.• Use the netX Driver to establish a USB, Serial (RS232) or TCP/IP connection from the CANopen Master DTM to the CANopen Master device.• The 3SGateway Driver for netX (V3.x) is used only in relationship with CODESYS. To search for devices you can check one or multiple drivers simultaneously. - Verify that the default driver is checked. - If necessary, check another driver or multiple drivers.		31
7	Configure Driver	If you use the netX Driver , you respectively must configure it. For netX Driver and communication via TCP/IP set the IP address of the device: - Select Settings > Driver > netX Driver > TCP Connection . - Via  add an IP range. - Under IP Address enter the IP Address of the device or an IP range. - Click Save . Adjust the driver parameters netX Driver USB/RS232 only if they differ from the default settings.  Note! <ul style="list-style-type: none">• The cifX Device Driver requires no configuration.• The configuration of the 3SGateway Driver for netX (V3.x) is carried out via the CODESYS surface.	<i>Configuring netX Driver</i>	34
8	Assign Master device (with or without firmware)	Assign the device to this driver. In the Master DTM configuration dialog: - select Settings > Device Assignment , - select a Master device (with or without firmware), - therefore check the appropriate checkbox, - select Apply .	<i>Selecting the Device (with or without firmware)</i>	43

#	Step	Short Description	For detailed information see section	Page
9	Select and download firmware	<p>If not yet a firmware was loaded to the device:</p> <ul style="list-style-type: none"> - Adhere to the necessary safety precautions to prevent personnel injury and property damage. <p>In the Master DTM configuration dialog:</p> <ul style="list-style-type: none"> - select Settings > Firmware Download, - select Browse.., - select a firmware file, - select Open, - select Download and Yes. 	<p><i>Safety Messages on Firmware or Configuration Download</i></p> <p><i>Firmware Download</i></p>	<p>27</p> <p>46</p>
10	Assign Master device once more (with firmware and system channel) <i>For repeated download this step is omitted.</i>	<p>In the Master DTM configuration dialog:</p> <ul style="list-style-type: none"> - select Settings > Device Assignment, - select Scan, - select the Master device (with loaded and defined system channel), - therefore check the appropriate checkbox, - select Apply, - close the Master DTM configuration dialog via OK. 	<i>Selecting the Device once more (with Firmware)</i>	44
11	Configure Slave device* (*This step won't be necessary if the network structure is scanned automatically. See step 17.)	<p>Configure the Slave device.</p> <ul style="list-style-type: none"> - Double click to the device icon of the Slave. - The Slave DTM configuration dialog is displayed. <p>In the Slave DTM configuration dialog:</p> <ul style="list-style-type: none"> - select Configuration > Object Dictionary, - define the object filters, - select Configuration > Process Data Objects > PDO Properties, - configure the PDO to be used for the communication, - select Configuration > Process Data Objects > PDO Mapping, - configure the list of the mapable or the list of the mapped objects each, - close the Slave DTM configuration dialog via OK. 	<i>(See Operating Instruction Manual Generic Slave DTM for CANopen Slave Devices)</i>	-
12	Configure Master device	<p>Configure the Master device.</p> <ul style="list-style-type: none"> - Double click to the device icon of the Master. - The Master DTM configuration dialog is displayed. <p>In the Master DTM configuration dialog</p> <ul style="list-style-type: none"> - select Configuration > Master Settings, - set the Master Settings. - select Configuration > Bus Parameters, - set the bus parameters, - select Configuration > Process data, - set symbolic names for the configured modules or signals. - select Configuration > Node ID Table, - configure the nodes for the data exchange, - select Configuration > Node BootUp, - configure the bootup of the nodes, - select Configuration > Monitoring, - configure the device monitoring of the nodes, - close the Master DTM configuration dialog via OK. 	<p><i>Configuring Device Parameters</i></p> <p><i>Master Settings</i></p> <p><i>Bus Parameters</i></p> <p><i>Process Data</i></p> <p><i>Node ID Table</i></p> <p><i>Node BootUp Mapping Sequence</i></p> <p><i>Monitoring</i></p>	<p>55</p> <p>57</p> <p>62</p> <p>66</p> <p>69</p> <p>73</p> <p>75</p>
13	Save project	<p>Depending of the frame application.</p> <p>For the configuration software:</p> <ul style="list-style-type: none"> - select File > Save. 	<i>(See Operating Instruction Manual of the Frame Application)</i>	-
14	Connect Master device	<p>Depending of the FDT Container.</p> <p>For netDevice:</p> <ul style="list-style-type: none"> - right click to the device icon of the Master, - select Connect. 	<i>Connecting/Disconnecting Device</i>	78

#	Step	Short Description	For detailed information see section	Page
15	Licensing	How to order licenses later and how to transfer them to the device.	<i>Licensing</i>	95
16	Download Configuration	<p>- Adhere to the necessary safety precautions to prevent personnel injury and property damage.</p> <p>Depending of the FDT Container.</p> <p>For netDevice:</p> <ul style="list-style-type: none"> - right click to the device icon of the Master, - select Download. 	<p><i>Safety Messages on Firmware or Configuration Download</i></p> <p><i>Download Configuration</i></p>	<p>27</p> <p>92</p>
17	Network Scan	<p>As an alternative to manually configure the Slave device, you can automatically scan the network structure by using the context menu Network Scan. Then confirm the query whether the module configuration of the Slave device shall be generated and download the configuration to the Master device.</p> <p>Therefore proceed the following steps:</p> <ol style="list-style-type: none"> 1. Start the Network Scan function. 2. Make the settings in the Scan Response dialog of the Master DTM. 3. Click Create devices. 4. Download of the Slave device configurations to the Master device (Download). 	<i>'Network Scan'</i>	85
18	Diagnosis	<p>Depending of the FDT Container.</p> <p>For netDevice:</p> <ul style="list-style-type: none"> - right click to the device icon of the Master, - select Diagnosis. - The Master DTM diagnosis dialog is displayed. <p>(1) Check whether the communication is OK: Diagnosis > General Diagnosis > Device status "Communication" must be green!</p> <p>(2) "Communication" is green: Open the IO Monitor and test the input or output data.</p> <p>(3) "Communication" is not green: Use Diagnosis and Extended diagnosis for troubleshooting.</p> <ul style="list-style-type: none"> - close the Master DTM diagnosis dialog via OK. 	<i>Overview Diagnosis</i>	108
19	IO Monitor	<p>Depending of the FDT Container:</p> <p>For netDevice:</p> <ul style="list-style-type: none"> - right click to the device icon of the Master, - select Diagnosis, - select Tools > IO Monitor. - Check the input or output data, - close the IO Monitor dialog via OK. 	<i>IO Monitor</i>	145
20	Disconnect	<p>Depending of the FDT Container.</p> <p>For netDevice:</p> <ul style="list-style-type: none"> - right click to the device icon of the Master, - select Disconnect. 	<i>Connecting/Disconnecting Device</i>	78

Table 9: Getting started - Configuration Steps

3.2 Safety Messages on Firmware or Configuration Download

If you perform a firmware download or a configuration download via the CANopen Master DTM be aware of the following:

⚠ WARNING**Communication Stop caused by Firmware or Configuration Download**

Initiating a firmware or configuration download process during bus operation will stop the communication and a subsequent plant stop may cause unpredictable and unexpected behavior of machines and plant components, possibly resulting in personal injury and damage to your equipment.

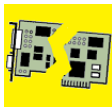
The firmware download overwrites the existing firmware. The communication stop may cause loss of device parameters and possible device damage may occur.

- Stop the application program, before you start the firmware or configuration download.
 - Make sure that all network devices are placed in a fail-safe condition.
-

⚠ WARNING**Mismatching System Configuration**

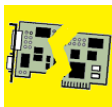
Mismatching system configuration loaded into the device could result in faulty data mapping in the application program and thus unexpected equipment operation may cause personal injury or damage of equipment.

- In the device use only a configuration suitable for the system.
-

NOTICE**Power Disconnect while downloading Firmware or Configuration**

If the power supply to the PC or device is interrupted while the firmware or configuration is being downloaded, the download will be aborted, the firmware may be corrupted, the device parameters may be lost, and the device may be damaged.

- During firmware or configuration download process do not interrupt the power supply to the PC, or to the device and do not perform a reset to the device!
-

NOTICE**Invalid Firmware**

Loading invalid firmware files could render your device unusable.

- Only proceed with a firmware version valid for your device.
-

4 Settings

4.1 Overview Settings

Settings Dialog Panes

The table below gives an overview for the individual **Settings** dialog panes descriptions:

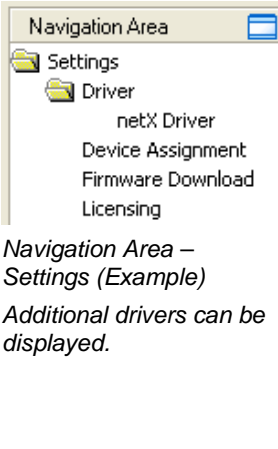
CANopen Master DTM	Folder Name / Section	Subsection	Manual Page
	Driver		31
		Verify or adapt Driver Settings	31
		cifX Device Driver	33
		netX Driver	33
		Configuring netX Driver	34
	Device Assignment		40
		Scanning for Devices	40
		Scanning for all Devices or for suitable only	42
		Selecting the Device (with or without firmware)	43
		Selecting the Device once more (with Firmware)	44
	Firmware Download		46
	Licensing		95

Table 10: Descriptions of the Dialog Panes Settings



Note: To edit the **Settings** dialog panes you need *User Rights* for “Maintenance”.



Notice the descriptions in the section *Settings for Driver and Device Assignment* on page 29.

To access to the online help with the descriptions of the drivers:

- Select **Settings > Driver > [Name of the assigned driver]**.
- Press the **F1** key.

4.2 Settings for Driver and Device Assignment



Important: For a 2-channel device consecutively channel 1 or channel 2 must be assigned to the DTM.

The following steps are needed to establish a connection from the CANopen Master DTM to the CANopen Master device:

Verifying or adapting Driver Settings

Verify the Driver Settings and adapt them if necessary.

1. Open the DTM configuration dialog.
 - In the FDT container **netDevice** double click to the CANopen Master device icon.
2. Verify that the default driver is checked and respectively check another or multiple drivers.
 - Select **Settings > Driver**.



Note! For PC cards cifX the **cifX Device Driver** is preset as a default driver. For all the other Hilscher devices the **netX Driver** is preset as a default driver.

- Use the **cifX Device Driver** if the CANopen Master DTM is installed on the same PC as the CANopen Master device.
- Use the **netX Driver** to establish an USB, Serial (RS232) or TCP/IP connection from the CANopen Master DTM to the CANopen Master device.
- The **3SGateway Driver for netX (V3.x)** is used only in relationship with CODESYS.

To search for devices on the network you can check one or multiple drivers simultaneously.

- Verify that the default driver for your device is checked.
- If necessary, check another driver or multiple drivers.

Configuring Driver



Note!

- The **cifX Device Driver** requires no configuration.
- The configuration of the **3SGateway Driver for netX (V3.x)** is carried out via the CODESYS surface.

If you use the **netX Driver**, you respectively must configure it.

3. Configure the **netX Driver** if necessary.

For the driver **netXDriver** an individual driver dialog window can be opened where you can configure the driver.

- Select **Settings > Driver > netX Driver**.
- For netX Driver and communication via TCP/IP set the IP address of the device.

Adjust the driver parameters **netX Driver USB/RS232** only if they differ from the default settings.

Assigning the Master device to the DTM

4. Scan for and select the devices (with or without firmware).
 - Select **Settings > Device Assignment**.
 - Under **Device selection** select *suitable only* or *all* and then **Scan**.
 - In the table check the required devices.
 - Select **Apply**.

Selecting and downloading the Firmware

5. If not yet a firmware was loaded to the device, select and download the firmware.
 - Select **Settings > Firmware Download**.
 - Select and download the firmware via **Download**.
 - Select **Apply**.
6. Scan for and select the devices (with firmware and defined system channel) once more.

For repeated download this step is omitted.

 - Select **Settings > Device Assignment**.
 - Select **Scan**.
 - In the table check the required device.
7. Close the DTM configuration dialog via **OK**.

Connecting the Device

8. In **netDevice** put a right-click on the CANopen Master device icon.
9. Select the **Connect** command from the context menu.
- 🔗 In the network view the device description at the device icon of the Master is displayed with a green colored background. The CANopen Master device now is connected to the CANopen Master DTM via an online connection.

Further Information



For descriptions about these steps refer to the sections following hereafter.

4.3 Driver

The **Driver** dialog pane displays the drivers to be used for a CANopen Master DTM to establish a device communication connection.



Note! A **default driver** is set in the configuration software.

Driver			
	Driver	Version	ID
<input checked="" type="checkbox"/>	CIFX Device Driver	1.101.1.9801	{368BEC5B-0E92-4C0E-B4A9-64F62AE7AAFA}
<input type="checkbox"/>	3SGateway Driver for netX (V3.x)	0.9.1.2	{787CD3A9-4CF6-4259-8E4D-109B6A6BEA91}
<input type="checkbox"/>	netX Driver	1.103.2.5183	{B54C8CC7-F333-4135-8405-6E12FC88EE62}

Figure 5: Default Driver 'CIFX Device Driver' for PC cards cifX

Parameter	Meaning
Driver	Name of the driver (for more details see descriptions hereafter)
Version	ODMV3 Version of the respective driver
ID	ID of the driver (driver identification)

Table 11: Driver Selection List Parameters

To establish a connection from the CANopen Master DTM to the CANopen Master device, verify if the default driver is checked and respectively check another driver or multiple drivers.

4.3.1 Verify or adapt Driver Settings

Proceed as follows:

1. Select **Settings > Driver** in the navigation area.
- The **Driver** dialog pane is displayed with the available drivers and the setting for the default driver.

Driver			
	Driver	Version	ID
<input checked="" type="checkbox"/>	CIFX Device Driver	1.101.1.9801	{368BEC5B-0E92-4C0E-B4A9-64F62AE7AAFA}
<input type="checkbox"/>	3SGateway Driver for netX (V3.x)	0.9.1.2	{787CD3A9-4CF6-4259-8E4D-109B6A6BEA91}
<input type="checkbox"/>	netX Driver	1.103.2.5183	{B54C8CC7-F333-4135-8405-6E12FC88EE62}

Figure 6: Default Driver 'CIFX Device Driver' for PC cards cifX (example)

Driver			
	Driver	Version	ID
<input type="checkbox"/>	CIFX Device Driver	1.101.1.9801	{368BEC5B-0E92-4C0E-B4A9-64F62AE7AAFA}
<input type="checkbox"/>	3SGateway Driver for netX (V3.x)	0.9.1.2	{787CD3A9-4CF6-4259-8E4D-109B6A6BEA91}
<input checked="" type="checkbox"/>	netX Driver	1.103.2.5183	{B54C8CC7-F333-4135-8405-6E12FC88EE62}

Figure 7: Default Driver 'netX Driver' for Hilscher devices except for PC cards cifX (example)

2. Verify that the default driver is checked.
- Verify that the default driver for your device is checked.

Default Driver (Pre-settings in the Configuration Software): For PC cards cifX the **cifX Device Driver** is preset as a default driver. For all the other Hilscher devices the **netX Driver** is preset as a default driver.

3. Respectively check another driver.



Note! The driver used for the connection from the CANopen Master DTM to the CANopen Master device must be supported by the device and must be available for the device.

- Use the **cifX Device Driver** if the CANopen Master DTM is installed on the same PC as the CANopen Master device.
 - Use the **netX Driver** to establish a USB, Serial (RS232) or TCP/IP connection from the CANopen Master DTM to the CANopen Master device.
 - The **3SGateway Driver for netX (V3.x)** is used only in relationship with CODESYS. The version V3.x refers to the driver version defined by 3S-Smart Software Solutions GmbH.
- Check the checkbox for the driver in the selection list.

4. Respectively check multiple drivers.

To search for devices on the network you can check multiple drivers simultaneously.

Driver			
	Driver	Version	ID
<input checked="" type="checkbox"/>	CIFX Device Driver	1.101.1.9801	{368BEC5B-0E92-4C0E-B4A9-64F62AE7AAFA}
<input type="checkbox"/>	3SGateway Driver for netX (V3.x)	0.9.1.2	{787CD3A9-4CF6-4259-8E4D-109B6A6BEA91}
<input checked="" type="checkbox"/>	netX Driver	1.103.2.5183	{B54C8CC7-F333-4135-8405-6E12FC88EE62}

Figure 8: Manual Selection of multiple drivers (Example)

4.3.2 cifX Device Driver

In the CANopen Master DTM for the **cifX Device Driver** no driver dialog pane is available, since for the **cifX Device Driver** no driver settings are required.

The **cifX Device Driver** will be used if the CANopen Master DTM is installed in the same PC as the CANopen Master device.



Note: To establish a connection from a DTM to a Master device via the **cifX Device Driver**, the **cifX Device Driver** must be installed and the driver must have access to the Master device.

4.3.3 netX Driver

The **netX Driver** is used to connect the DTM to the device via different connection types. The DTM communicates with the device via an USB connection, a serial (RS232) connection or a TCP/IP connection. The **netX Driver** establishes

- via the USB interface of the device and the USB port of the PC an USB connection to the device,
- via the RS232 interface of the device and the COM port of the PC a serial connection (RS232) to the device
- and via Ethernet a TCP/IP connection to the device.

To connect the DTM to the physical layer of the device the **netX Driver** software works in combination with the software components:

- “USB/COM connector” for the USB connection and for the serial connection (RS232) and
- “TCP/IP connector” for the Ethernet connection.

4.3.4 Configuring netX Driver

The following steps are required to configure the netX Driver:

USB/RS232 Connection

To set the driver parameters for an USB/RS232 connection note:




Note: Adjust the driver parameters netX Driver USB/RS232 only if they differ from the default settings. After saving the changed driver parameters, these parameters are used for the device assignment when scanning devices.

For setting the driver parameters for an USB connection or a serial connection:

1. Select **Settings > Driver > netX Driver > USB/RS232 Connection**.
 - Set the driver netX Driver USB/RS232 parameters.

TCP/IP Connection

For setting the driver parameters for a TCP/IP connection:

1. Select **Settings > Driver > netX Driver > TCP Connection**.
2. Set IP Address of the device:
 - Add an IP Range via **Select IP Range** .
3. Under **IP Range Configuration > IP Address** enter the IP Address of the device (**Use IP Range** is unchecked).

Or

4. Set IP Range:
 - Check **Use IP Range**.
 - Under **IP Range Configuration > IP Address** enter the start address (left side) and the ending address of the IP scanning range (right side).
 5. Click **Save**, to save the IP address or the IP range.
- After saving the changed driver parameters, these parameters are used for the device assignment when scanning devices.

4.3.5 netX Driver - USB/RS232 Connection

The communication from the DTM to the device via an **USB/RS232 Connection** is used when the DTM is installed on a PC and between the PC and the device

- an USB connection
- or a serial connection (RS232) exists.

The DTM accesses the device via the USB interface or via the RS232 interface. This requires either to connect an USB port of the PC to the USB interface of the device using an USB cable or to connect a physical COM port of the PC to the RS232 interface of the device via a serial cable.

The **netX Driver / USB/RS232 Connection** supports all physical and virtual COM ports available on the PC.

Via the RS232 interface or USB interface, the device is configured or diagnosis is performed.

4.3.5.1 Driver Parameters for netX Driver - USB/RS232 Connection

The settings of the driver parameters for the USB/RS232 connection are made via the **netX Driver / USB/RS232 Connection** configuration dialog.

- Open the **USB/RS232 Connection** dialog via navigation area **Settings > Driver > netX Driver**.

- The **USB/RS232 Connection** dialog is displayed:

The screenshot shows the 'netX Driver' configuration window with the 'USB/RS232 Connection' tab selected. The 'TCP Connection' tab is also visible. The 'Enable USB/RS232 Connector (Restart of ODM required)' checkbox is checked. The 'Select Port:' dropdown is set to 'COM1'. The 'Port Configuration' section contains a 'Disable Port' checkbox (unchecked) and several settings: 'Baud Rate' (115.2 kBit/s), 'Byte Size' (8 Byte), 'Stop Bits' (1 Stopbit), 'Parity' (No Parity), 'Send Timeout' (1000 ms), 'Keep Alive Timeout' (2000 ms), and 'Reset Timeout' (10000 ms). At the bottom, there are three buttons: 'Restore', 'Save', and 'Save All'.

Figure 9: netX Driver > USB/RS232 Connection


Parameter	Meaning	Range of Value / Default Value
Enable USB/RS232 Connector (Restart of ODM required)	<p>checked: The netX Driver can communicate via the USB/RS232 interface.</p> <p>unchecked: The netX Driver can <u>not</u> communicate via the USB/RS232 interface.</p> <p>If the check mark for Enable USB/RS232 Connector is set or removed, then the ODM server must be restarted¹, to make the new setting valid.</p> <p>¹ Restart the ODM server via the ODMV3 Tray Application:</p> <p>- In the foot line click on  using the right mouse key.</p> <p>- In the context menu select Service > Start.</p>	checked, unchecked; Default: unchecked
Select Port	Depending on the COM ports (interfaces) available on the PC, they will be listed under Select Port .	COM 1 to COM N
Port Configuration		
Disable Port	<p>checked: No connection.</p> <p>unchecked: The netX Driver tries to establish a connection using the configured USB/RS232 interface.</p>	checked, unchecked (Default)
Baud rate	<p>Transfer rate: number of bits per second.</p> <p>The device must support the baud rate.</p>	9.6, 19.2, 38.4, 57.6 or 115.2 [kBit/s]; Default (RS232): 115.2 [kBit/s]
Stop bits	Number of stop bits sent after the transfer of the send data for synchronization purposes to the receiver.	Stop bit: 1, 1.5, 2; Default (RS232): 1
Send Timeout	Maximum time before the transfer of the transmission data is canceled, when the send process fails, for example, because of the transfer buffer is full.	100 ... 60.000 [ms]; Default (RS232 and USB): 1000 ms
Reset Timeout	Maximum time for a device reset, including the re-initialization of the physical interface used for the communication.	100 ... 60.000 [ms]; Default (RS232 and USB): 5000 ms
Byte size	Number of bits per byte by byte specification	7 Bit, 8 Bit; Default (RS232): 8 Bit
Parity	<p>In the error detection in data transmission using parity bits, "parity" describes the number of bits occupied with 1 in the transmitted information word.</p> <p>No Parity: no parity bit</p> <p>Odd Parity: The parity is "odd" if the number of bits occupied with 1 in the transmitted information word will be odd.</p> <p>Even parity: The parity is "even" if the number of bits occupied with 1 in the transmitted information word will be even.</p> <p>Mark Parity: if the parity bit is always 1, this will be named mark-parity (the bit does not contain any information).</p> <p>Space Parity: if the parity bit always 0, this will be named space-parity (the bit represents an empty space).</p>	No Parity, Odd Parity, Even Parity, Mark Parity, Space Parity; Default (RS232): No Parity
Keep Alive Timeout	The "Keep Alive" mechanism is used to monitor whether the connection to the device is active. Connection errors are detected using a periodic heartbeat mechanism. The heartbeat mechanism will be initiated after the set time has elapsed if the communication has failed.	100 ... 60.000 [ms]; Default (RS232 and USB): 2000 ms
Restore	Resets all settings in the configuration dialog to the default values.	
Save	Saving all settings made in the configuration dialog netX Driver > Save USB/RS232 Connection , i. e. only for the selected connection type.	
Save All	Saving all settings made in the configuration dialog netX Driver , i. e. for all connection types.	

Table 12: Parameters netX Driver > USB/RS232 Connection

4.3.6 netX Driver - TCP/IP Connection

The communication from the DTM to the device via a **TCP/IP Connection** is used in the following two typical applications:

Application 1: The device has its own Ethernet interface. The DTM is installed on a PC and the TCP/IP connection is established from this PC to the stand-alone device. The IP address of the device is used.

Application 2: The device is installed in a remote PC. The DTM is installed on an additional PC and the TCP/IP connection is established from this PC to the remote PC. The IP address of the remote PC is used. For the TCP/IP connection is made, on the remote PC the cifX TCP/IP server must be started. The cifX TCP/IP server allows the remote access to the device via a TCP/IP connection.



Note: An exe file for the cifXTCP/IP server is provided on the product CD in the *Tools* directory.

Via the TCP/IP interface of the device or of the remote PC, the device is configured or diagnosis is performed.

4.3.6.1 Driver Parameters for netX Driver - TCP/IP Connection

The settings of the driver parameters for the TCP/IP connection are made via the **netX Driver / TCP Connection** configuration dialog.

- Open the **TCP Connection** dialog via navigation area **Settings > Driver > netX Driver**.
- The dialog **netX Driver** is displayed:
- Select **TCP Connection**.

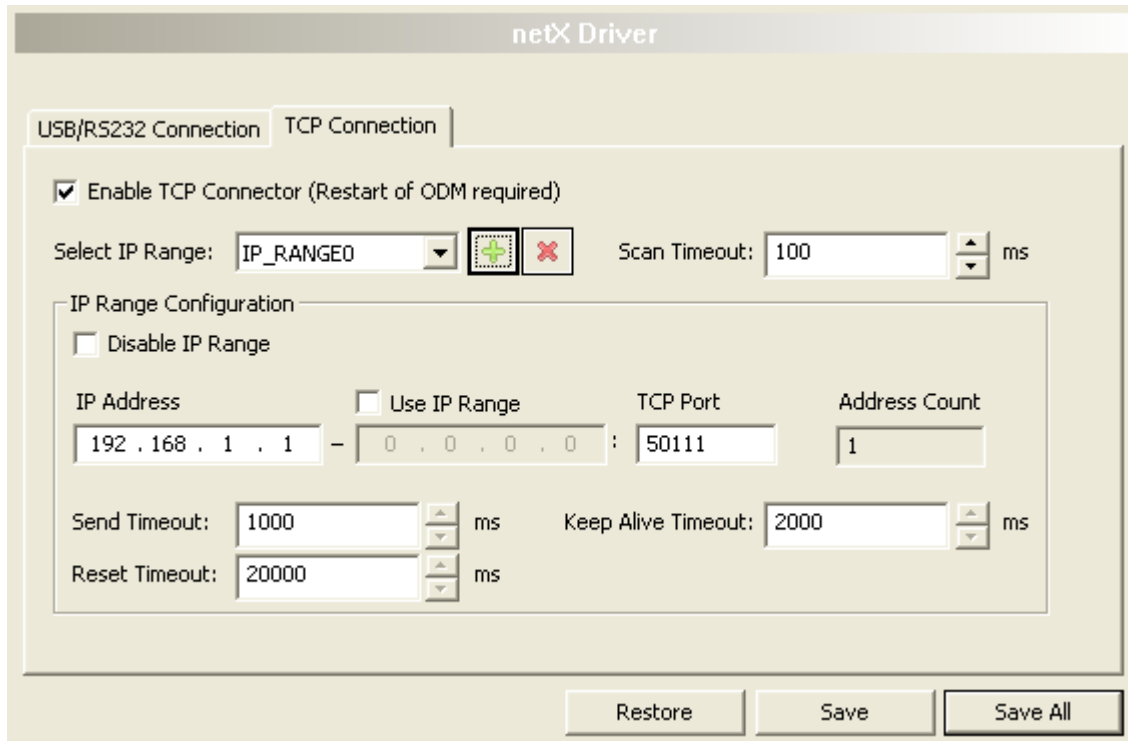





Figure 10: netX Driver > TCP Connection

Parameter	Meaning	Range of Value / Default Value
Enable TCP Connector (Restart of ODM required)	<p>checked: The netX Driver can communicate via the TCP/IP interface.</p> <p>unchecked: The netX Driver can <u>not</u> communicate via the TCP/IP interface.</p> <p>If the check mark for Enable TCP Connector is set or removed, then the ODM server must be restarted¹, to make the new setting valid.</p> <p>¹ Restart the ODM server via the ODMV3 Tray Application:</p> <ul style="list-style-type: none"> - In the foot line click on  using the right mouse key. - In the context menu select Service > Start. 	checked, unchecked; Default: unchecked
Select IP Range	<p>Via Select IP Range already created IP ranges can be selected.</p> <p>Via  an additional IP range can be added.</p> <p>Via  an IP range can be deleted.</p>	

Parameter	Meaning	Range of Value / Default Value
Scan Timeout [ms]	With Scan Timeout can be set, how long to wait for a response while a connection is established.	10 ... 10000 [ms]; Default: 100 ms
IP Range Configuration		
Disable IP Range	checked: No connection. unchecked: The netX Driver tries to establish a connection using the configured TCP/IP interface.	checked, unchecked (Default)
IP Address (left)	Enter the IP address of the device, (if Use IP Range is not checked). Enter the start address of the IP scanning range, (if Use IP Range is checked).	valid IP address; Default: 192.168.1.1
Use IP Range	checked: An IP address range is used. unchecked: Only one IP address is used.	checked, unchecked; Default: unchecked
IP Address (right)	Enter the ending address of the IP scanning range, (only if Use IP Range is checked).	valid IP address; Default: 0.0.0.0
Address Count	Displays the scanning range address count, depending on the selected IP-start or IP-end address. (For this read the note given below.)	recommended: 10
TCP Port	Identifies the endpoint of a logical connection or addresses a specific endpoint on the device or PC.	0 - 65535; Default Hilscher device: 50111
Send Timeout [ms]	Maximum time before the transfer of the transmission data is canceled, when the send process fails, for example, because of the transfer buffer is full.	100 ... 60.000 [ms]; Default (TCP/IP): 1000 ms
Reset Timeout [ms]	Maximum time for a device reset, including the re-initialization of the physical interface used for the communication.	100 ... 60.000 [ms]; Default (TCP/IP): 2000 ms
Keep Alive Timeout [ms]	The "Keep Alive" mechanism is used to monitor whether the connection to the device is active. Connection errors are detected using a periodic heartbeat mechanism. The heartbeat mechanism will be initiated after the set time has elapsed if the communication has failed.	100 ... 60.000 [ms]; Default (TCP/IP): 2000 ms
Restore	Resets all settings in the configuration dialog to the default values.	
Save	Saving all settings made in the configuration dialog netX Driver > Save TCP/IP Connection , i. e. only for the selected connection type.	
Save All	Saving all settings made in the configuration dialog netX Driver , i. e. for all connection types.	

Table 13: Parameters netX Driver > TCP Connection



Note: Do not use large IP ranges in combination with a low scan timeout. Microsoft introduced in Windows® XP SP2 a limit of concurrent half-open outbound TCP/IP connections (connection attempts) to slow the spread of virus and malware from system to system. This limit makes it impossible to have more than 10 concurrent half-open outbound connections. Every further connection attempt is put in a queue and forced to wait. Due to this limitation a large IP range used in combination with a low scan timeout could prevent the connection establishment to a device.

4.4 Device Assignment



Note: In the **Device Assignment** dialog pane you first must assign the CANopen Master device to the CANopen Master DTM by checking the check box. This is essential to establish an online connection from the CANopen Master DTM to the CANopen Master device later, as described in section *Connecting/Disconnecting Device* on page 78.

Therefore in the **Device Assignment** dialog pane you scan for the CANopen Master device and select it.

If the device did not get a firmware or shall get a new firmware:

1. first you scan for the device (with or without firmware) and select the device,
2. then you download a firmware to the device and
3. subsequently you scan for the device (with firmware) once more and select the device again.



Important: For a 2-channel device consecutively channel 1 or channel 2 must be assigned to the DTM.

4.4.1 Scanning for Devices

1. Select **Settings > Device Assignment** in the navigation area.

➤ The dialog pane **Device Assignment** is displayed.

Device	Hardware Port 0/1/2/3	Slot number	Serial number	Driver	Channel Protocol	Access path
<input type="checkbox"/> Device Cl*	-/-/PROFIBUS/-	1	20148	CIFX Device Driver	Undefined Undefined	...\cifX3_SYS

Figure 11: Device Assignment - detected Devices (* The name of the device class is displayed.) – Example for a device without firmware

2. Under **Device Selection** select *suitable only*.
3. Select **Scan**, to start the scanning process.

➤ In the table all devices are displayed, which can be connected to the CANopen Master DTM via the preselected driver.



Note: For devices, which have been found via the **cifX Device Driver** in the column **Access path** the indication **...\cifX[0toM]_SYS** is displayed. This is correct, as long as a device did not get a firmware. After the firmware download has been completed, in the column **Access path** the indication **...\cifX[0toM]_Ch[0to3]** is displayed.

Parameter	Meaning	Range of Value / Default Value
Device selection	Selecting suitable only or all devices.	suitable only, all
Device	Device class of the CANopen Master Devices.	
Hardware Port 0/1/2/3	Shows, which hardware is assigned to which communication interface.	
Slot number	Shows the Slot Number (Card ID) preset at the PC card cifX via the Rotary Switch Slot Number (Card ID) . The indication n/a means, that no Slot-Nummer (Card ID) exists. This will occur if the PC card cifX is not equipped with a Rotary Switch Slot Number (Card ID) or for PC cards cifX equipped with a Rotary Switch Slot Number (Card ID) if the rotary switch is set to the value 0 (zero).	1 to 9, n/a
Serial number	Serial number of the device	
Driver	Name of the driver	
Channel Protocol	Shows, which firmware is loaded to which device channel. The data for the used channel consists of the protocol class and the communication class. a.) For devices without firmware: Undefined Undefined, b.) For devices with firmware: Protocol name corresponding to the used Firmware	
Access path (last column on the right)	Depending on the used driver in the column Access path different data to the device are displayed. For the cifX Device Driver the following data are displayed: a.) For devices without firmware: ...cifX[0toN]_SYS, b.) For devices with firmware: ...cifX[0toN]_Ch[0to3]. cifX[0toN] = Board number 0 to N Ch[0to3] = Channel number 0 to 3	Depending on the device and on the driver: board or channel number, IP address or COM interface
Access path (at the lower side of the dialog pane)	If in the table a device is checked, under Access path (at the lower side of the dialog pane) the driver identification or depending on the used driver additional data to the device will be displayed. For the cifX Device Driver the following data are displayed: a.) For devices without firmware: ...cifX[0toN]_SYS, b.) For devices with firmware: ...cifX[0toN]_Ch[0to3]. cifX[0toN] = Board number 0 to N Ch[0to3] = Channel number 0 to 3	driver identification (ID) depending on the device and on the driver: board or channel number, IP address or COM interface

Table 14: Parameters of the Device Assignment

4.4.1.1 Scanning for all Devices or for suitable only

all

1. Under **Device Selection** select *all*.
2. Select **Scan**.

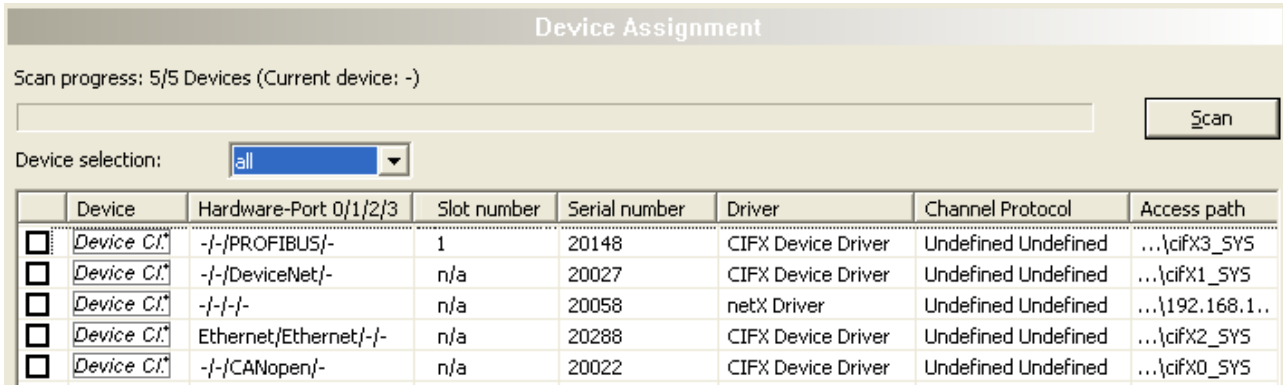


Figure 12: Device Assignment - detected Devices (* The name of the device class is displayed.) Example for Devices without Firmware

- ⇒ In the table all devices are displayed, which are attainable in the network and which can be connected to a single DTM each via the preselected drivers.



Note: During a subsequent firmware download in the selection window **Select Firmware File** all files from the selected folder are displayed, under **Files of Type** „All Files (*.*)“ is displayed and the check box **Validate the selected firmware file.** is unchecked.

suitable only

1. Under **Device Selection** select *suitable only*.
2. Select **Scan**.

- ⇒ In the table all devices are displayed, which can be connected to the CANopen Master DTM via the preselected drivers.



Note: During a subsequent firmware download in the selection window **Select Firmware File** only firmware files from the selected folder are displayed, under **Files of Type** „Firmware Files (*.nxm)“ or „Firmware Files (*.nxf)“ is displayed and the check box **Validate the selected firmware file.** is checked.

4.4.2 Selecting the Device (with or without firmware)



Note: A connection with the CANopen Master DTM can only be established with one CANopen Master device.

To select the physical CANopen Master device (with or without firmware):

1. Check the appropriate device.

Device Assignment

Scan progress: 5/5 Devices (Current device: -)

Device selection:

	Device	Hardware Port 0/1/2/3	Slot number	Serial number	Driver	Channel Protocol	Access path
<input checked="" type="checkbox"/>	Device Cl*	-/-/PROFIBUS/-	1	20148	CIFX Device Driver	PROFIBUS Master	...\cifX3_SYS

Access path: {368BEC5B-0E92-4C0E-B4A9-64F62AE7AAFA}\cifX3_SYS

Figure 13: Device Assignment - Selecting the Device (* The name of the device class is displayed.) – Example for a device without firmware / one Device is selected

- Under **Access path** (below in the dialog pane) the access path to the device, e. g. the driver identification, or depending on the used driver additional access data of the device are displayed.

2. Select **Apply**, to apply the selection.



Note: Before an online connection from the CANopen Master DTM to the CANopen Master device can be established, a firmware must be loaded to the device and the device must be selected once more.



For further information refer to section *Firmware Download* on page 46 or to section *Selecting the Device once more (with Firmware)* on page 44.

4.4.3 Selecting the Device once more (with Firmware)



Note: For repeated download this step is omitted.

To select the CANopen Master device (with firmware and defined system channel) once more, proceed as described hereafter:

all

1. Under **Device Selection** select *all*.
2. Select **Scan**.
 - In the table all devices are displayed, which are attainable in the network and which can be connected to a DTM via the preselected drivers.
3. Check the appropriate device.

Device Assignment

Scan progress: 5/5 Devices (Current device: -)

Device selection: all Scan

	Device	Hardware-Port 0/1/2/3	Slot number	Serial number	Driver	Channel Protocol	Access path
<input checked="" type="checkbox"/>	Device Cl.*	-/-/PROFIBUS/-	1	20148	CIFX Device Driver	PROFIBUS-DP Master	...\cifX3_Ch0
<input type="checkbox"/>	Device Cl.*	-/-/DeviceNet/-	n/a	20027	CIFX Device Driver	DeviceNet Master	...\cifX1_Ch0
<input type="checkbox"/>	Device Cl.*	-/-/-/-	n/a	20058	netX Driver	Undefined Undefined	...\192.168....
<input type="checkbox"/>	Device Cl.*	Ethernet/Ethernet/-/-	n/a	20288	CIFX Device Driver	PROFINET IO Device	...\cifX2_Ch0
<input type="checkbox"/>	Device Cl.*	-/-/CANopen/-	n/a	20022	CIFX Device Driver	Undefined Undefined	...\cifX0_SYS

Access path: {368BEC5B-0E92-4C0E-B4A9-64F62AE7AAFA}\cifX3_Ch0

Figure 14: Device Assignment - Selecting the Device (* The name of the device class is displayed.) – Example for Devices with and without Firmware / one Device is selected



Note: After the firmware download has been completed, for the devices which have been detected via the **cifX Device Driver** the following data are displayed:

- In the column **Channel Protocol**: the data for the firmware for the used channel
- In the column **Access path** or under **Access path** (below in the dialog pane): the data: ...\\cifX[0toN]_Ch[0to3].
 cifX[0toN] = board number 0 to N
 Ch[0to3] = channel number 0 to 3

4. Select **Apply**, to apply the selection.
5. Or select **OK**, to apply the selection and to close the DTM interface dialog.
6. Connect the DTM to the device using the context menu (right mouse click).

Or:

suitable only

1. Under **Device Selection** select *suitable only*.
2. Select **Scan**.
- In the table all devices are displayed, which can be connected to the CANopen Master DTM via the preselected drivers.
3. Check the appropriate device.

Device Assignment

Scan progress: 5/5 Devices (Current device: -)

Device selection: suitable only Scan

	Device	Hardware Port 0/1/2/3	Slot number	Serial number	Driver	Channel Protocol	Access path
<input checked="" type="checkbox"/>	Device Cl*	-/-/PROFIBUS/-	1	20148	CIFX Device Driver	PROFIBUS-DP Master	...\cifX3_Ch0

Access path: {368BEC5B-0E92-4C0E-B4A9-64F62AE7AAFA}\cifX3_Ch0

Figure 15: Device Assignment - Selecting the Device (* The name of the device class is displayed.) – Example for a device with firmware / one Device is selected



Note: After the firmware download has been completed, for the devices which have been detected via the **cifX Device Driver** the following data are displayed:

- In the column **Channel Protocol**: the data for the firmware for the used channel
- In the column **Access path** or under **Access path** (below in the dialog pane): the data: ...\\cifX[0toN]_Ch[0to3].
cifX[0toN] = board number 0 to N
Ch[0to3] = channel number 0 to 3

4. Select **Apply**, to apply the selection.
5. Or select **OK**, to apply the selection and to close the DTM interface dialog.
6. Connect the DTM to the device using the context menu (right mouse click).



For further information how to establish an online connection from the CANopen Master DTM to the CANopen Master device, refer to section *Connecting/Disconnecting Device* on page 78.

4.5 Firmware Download

Using the **Firmware Download** dialog a firmware can be transferred to the device.



Note: Prior to the firmware download, you must select the driver and the Master device (with or without firmware) and the device must be assigned to the hardware.



For further information refer to section **Overview Settings** on page 28.

To load the firmware to the device:

1. In the navigation area select **Settings > Firmware Download**.
 ➤ The dialog **Firmware-Download** pane is displayed.

Figure 16: Firmware Download

Element	Meaning
Name	The path and name of the firmware file selected are displayed.
Version	The version and build version of the firmware file selected are displayed.
Browse...	Via 'Browse...' you can select the firmware file for the download.
Download	Via 'Download' you can download the firmware to the device.

Table 15: Parameter Firmware Download

2. Select the firmware file.

➤ Select **Browse**.

Device is not assigned to the Hardware

If the device is not assigned to the Hardware, the error message 'The device is not assigned to the hardware!' is displayed:

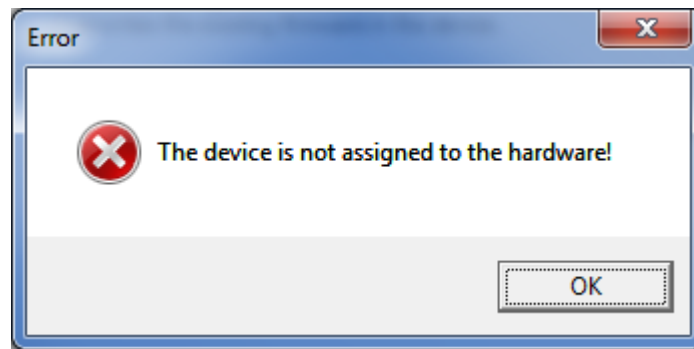


Figure 17: Error Message 'The device is not assigned to the hardware!'

- Click **OK** and select and assign the Master device as described in section *Device Assignment*.

Device is assigned to the Hardware

- The selection window **Select Firmware File** is displayed.
- Enlarge the selection window to view the columns **Hardware** and **Version**.

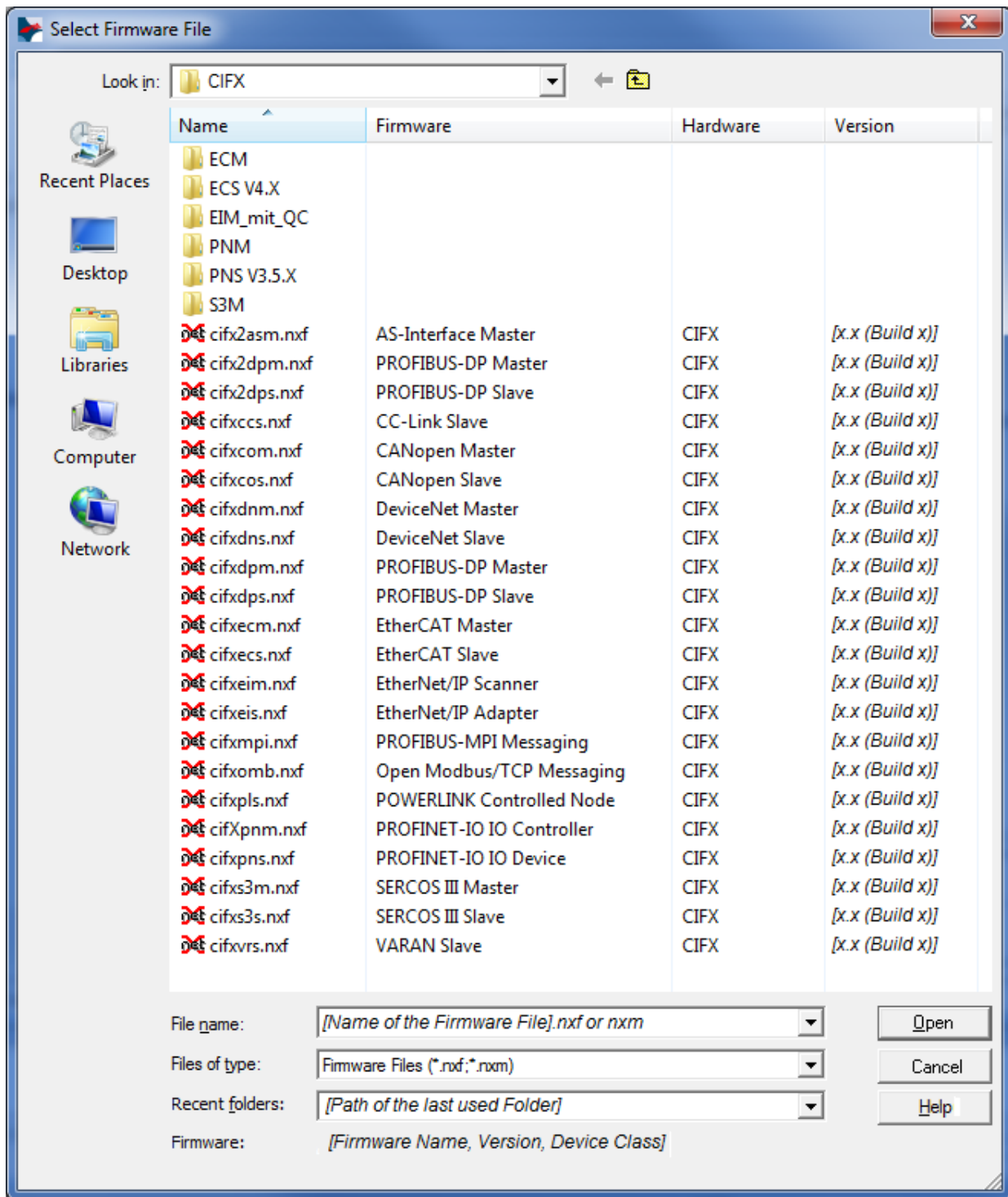


Figure 18: Window 'Select Firmware File' (Example CIFX)

Parameter	Meaning	Range of Value / Default Value
Column Name	File name of the firmware file To sort the entries of the window Select Firmware File by name click to the column head Name .	nxf, nxm
Column Firmware	Name of the firmware (consisting of the protocol name and protocol class)	
Column Hardware	Device class of the associated hardware	e. g. CIFX, COMX, COMX 51, NETJACK 10, NETJACK 50, NETJACK 51,

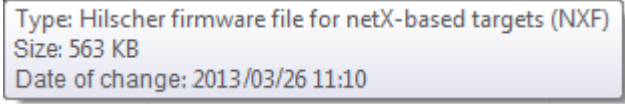
		NETJACK 100, NETTAP 50 (Gateway), NETTAP 100 (Gateway), NETBRICK 100 (Gateway)
Column Version	Firmware version	x.x (build x)
Tooltip	To view the tooltip information move with the mouse pointer over the selected firmware line. 	
Files of Type	„All Files (*.*)“ if before in the Device Assignment pane under Device selection <i>all</i> was selected. „Firmware Files (*.nxm)“ or <i>Firmware Files (*.nxf)</i> if before in the Device Assignment pane under Device selection <i>suitable only</i> was selected.	All Files (*.*), Firmware Files (*.nxm), Firmware Files (*.nxf)
Recent folders	Path of the recently opened folder	
Firmware	As soon as the firmware file has been selected, under Firmware the name, the version and the build version as well as the device class for the selected firmware is displayed.	Name, Version, Build Version, Device Class for the selected firmware
Help	Button, to open the online help of the DTM.	

Table 16: Parameters Select Firmware File




Further descriptions to the selection window **Select Firmware File** are included in the context sensitive help (**F1** key) of the Microsoft Corporation.




Note: After in the **Device Assignment** pane under **Device selection** *all* or *suitable only* has been set, during a subsequent firmware download in the selection window **Select Firmware File** the following data are displayed or set:

(for list box entry →)	all	suitable only
In the selection window Select Firmware File :	all files from the selected folder	only firmware files from the selected folder
Under Files of Type *:	„All Files (*.*)“	„Firmware Files (*.nxm)“, „Firmware Files (*.nxf)“
Validation:	A restricted validation will be performed if the selected firmware is applied for the download.	A validation is made whether the firmware file is suitable for the CANopen Master DTM.

*These settings in the selection window **Select Firmware File** can also be changed manually.

- In the selection window mark the firmware file to be loaded using the mouse.
-  In the selection window under **Firmware** the name and the version of the firmware are displayed.
- In the selection window select the **Open** button.

Validation

-  A validation is made, whether the selected firmware file is suitable for the CANopen Master device.

Invalid Firmware

NOTICE

Invalid Firmware

Loading invalid firmware files could render your device unusable.

- Only proceed with a firmware version valid for your device.
- If a firmware file is selected, which is not valid for the assigned device, the request **Select Firmware File** will be displayed.

‘Invalid firmware for assigned device!’

[detailed explication]

Shall firmware file nevertheless be applied for the download?’



Figure 19: Request Select Firmware File - Example Invalid Firmware

- Answer to the request with **No** and select a valid firmware.
- The selection window is closed.

Valid Firmware

➤ The selection window is directly closed (without dialog).

5. Start firmware upgrade.

⚠ WARNING

Communication Stop caused by Firmware Update, faulty System Operation possible, Overwriting of Firmware or Loss of Device Parameters

Before you initiate a firmware download process, while the bus is still in operation status:

- Stop the application program.
- Make sure that all network devices are placed in a fail-safe condition.

NOTICE

Firmware Corruption or Loss of Parameters caused by Power Disconnect during Firmware Download

- During firmware download process, do not interrupt the power supply to the PC, or to the device and do not perform a reset to the device!
- In the dialog pane **Firmware Download** click to the **Download** button, to download the firmware.
- The request **Do you really want to download the firmware?** is displayed.

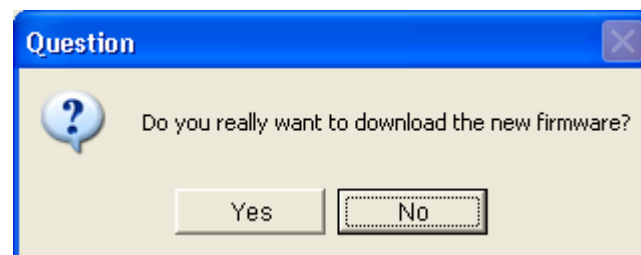


Figure 20: Request - Do you really want to download the firmware?

6. Click **Yes**.

- If you are sure, that you have selected the appropriate firmware file answer to the request with **Yes** otherwise with **No**.
- During the download a progress bar is displayed ('Download active, device performs initialization...'), in the status line a clock / green hook symbol is displayed and in the dialog pane **Firmware Download** **Download** is grayed out.

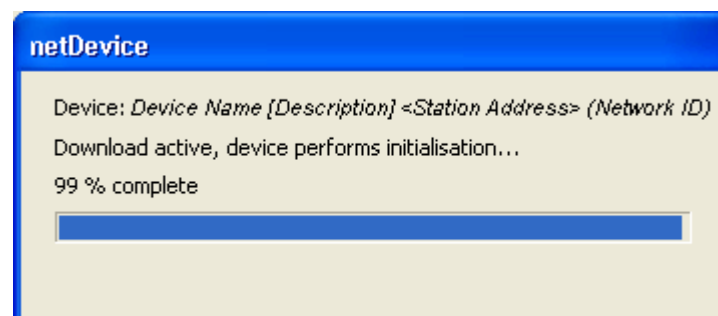


Figure 21: Firmware Download - Progress Bar

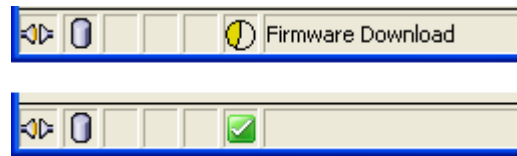


Figure 22: Clock Symbol and Hook Symbol green

- In the **Firmware-Download** dialog pane the path and name as well as the version of the selected firmware file are displayed.

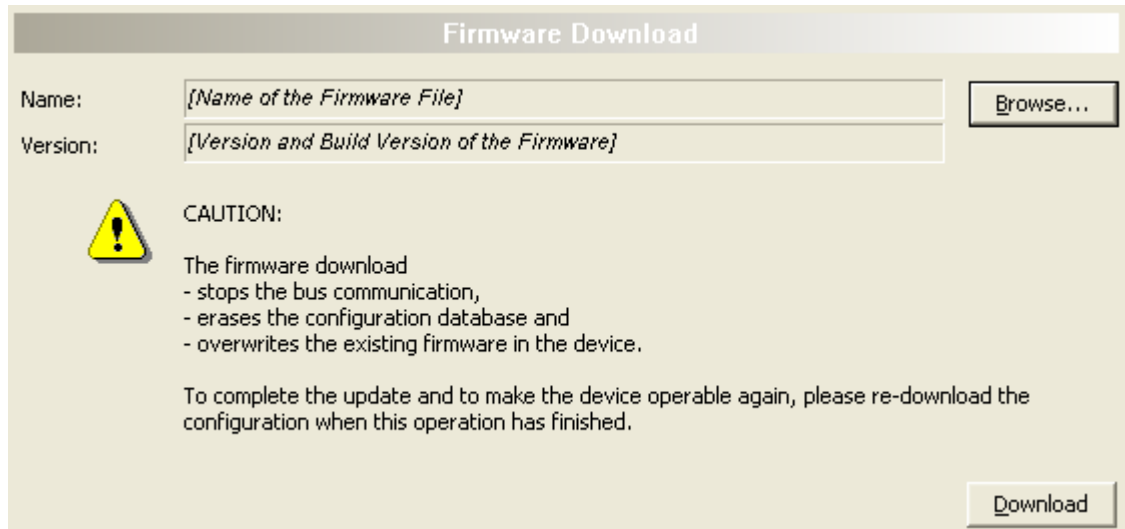


Figure 23: Firmware Download – Download

4.6 Licensing

To open the Licensing pane:

1. In the navigation area select **Settings > Licensing**.

➤ The dialog **Licensing** pane is displayed.

Licensing

License Type

	Existing	Order
Master protocols		
One General Master License	NO	<input type="checkbox"/>
Two General Master Licenses	NO	<input type="checkbox"/>
PROFIBUS Master	YES	<input type="checkbox"/>
CANopen Master	YES	<input type="checkbox"/>
DeviceNet Master	YES	<input type="checkbox"/>
AS-Interface Master	YES	<input type="checkbox"/>
PROFINET IO RT Controller	YES	<input type="checkbox"/>

Request Form, please fill out

Name	Value
License type	User Single Device License
Manufacturer*	00000001
Article number*	01250510
Serial number*	00020086
Chiptype*	00000002
Step*	00000000
Romcode revision*	00000002

Fields marked with '*' are mandatory.

Hilscher Germany

E-mail... license@hilscher.com

Print Fax Form... +49 6190 9907-50

Telephone... +49 6190 9907-0

Export License Request...

Download License

Figure 24: Licensing



For a detailed description refer to section *Licensing* on page95.

5 Configuration

5.1 Overview Configuration

Configuration Dialog Panes

The table below gives an overview for the **Configuration** dialog panes descriptions:

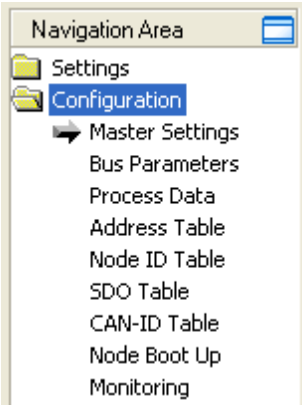
CANopen Master DTM	Folder Name / Section	Subsection	Manual Page
 <p>Navigation Area - Configuration</p>	Master Settings		57
		Start of Bus Communication	58
		Application Monitoring	58
		Process Image Storage Format	59
		Module Alignment	59
		Process Data Handshake	60
		Advanced	60
		Device Status Offset	61
	Bus Parameters		62
		Device Description	63
		Node Settings	63
		Sync Master Settings	64
		29 Bit COB-ID	64
	Process Data		66
	Address Table		67
	Node ID Table		69
	SDO Table		70
	CAN-ID Table		72
	Node BootUp Mapping Sequence		73
	Monitoring		75
		Node Guarding Protocol	76
		Heartbeat Protocol	77

Table 17: Descriptions of the Dialog Panes Configuration



Have in mind the descriptions in the section *Configuration Steps* on page 23.

5.2 Configuring Device Parameters



Important: For a 2-channel device consecutively channel 1 or channel 2 each must be configured individually.

The following steps are required to configure the parameters of the CANopen Master device using the CANopen Master DTM:

Master Settings

1. Set the Master Settings:
 - Select **Configuration > Master Settings** in the navigation area.
 - Under **Start of Bus Communication** select **Automatically by device** or **Controlled by application** option.
 - Under **Application Monitoring** set the **Watchdog time**.
 - Under **Module Alignment** set the options **Byte boundaries** or **2 Byte boundaries**.
 - Under **Process Data Handshakes** select the **Process Data Handshake** type to be used.
 - Under **Device Status Offset** select **Automatic calculation** or **Static** option.

Bus Parameters

2. Set the Bus Parameters.
 - Select **Configuration > Bus Parameters** in the navigation area.
 - Put the settings for the **Device Description**, the **Node Settings**, the **Sync Master Settings** and the **29 Bit COB-ID**.

Process Data

3. Set the process data:
 - Select **Configuration > Process Data** in the navigation area.
 - Set symbolic names for the configured modules or signals.

Node ID Table

4. Configure the nodes for the data exchange to the Master.
 - Select **Configuration > Node ID Table** in the navigation area.

Node BootUp

5. Configure the startup behavior of the Master in regard to every single node.
 - Select **Configuration > Node BootUp** in the navigation area.

Monitoring

6. Configure the device monitoring of the single nodes.
 - Select **Configuration > Monitoring** in the navigation area.

Close Master DTM Configuration Dialog

7. Click **OK** in order to close the Master DTM configuration dialog and to store your configuration.

Configuration Download to the CANopen Master Device

- Adhere to the necessary safety precautions to prevent personnel injury and property damage.



Note: In order to transfer the configuration to the CANopen Master device, download the data of the configuration parameters in the CANopen Master device. See section *Download Configuration* on page 92.

Further Information

For further information refer to section *Master Settings* on page 57, to section *Bus Parameters* on page 62, to section *Process Data* on page 66, to section *Node ID Table* on page 69, to section *Node BootUp Mapping Sequence* on page 73 and to section *Monitoring* on page 75, of this document.

5.3 Master Settings

At the **Master Settings** pane device related settings can be made. These settings only become active after the configuration was downloaded to the device.



Information about the download you find in section *Download Configuration* on page 92).

Figure 25: Configuration > Master Settings



Note: The setting options at the dialog pane **Master Settings** for client specific variants of the configuration software can differ from the setting options displayed here.

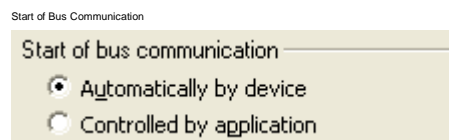


Figure 26: Master Settings > Start of Bus Communication

If **Automatically by device** is selected, the CANopen Master device will start with the data exchange on the bus after the initialization has been ended.

If **Controlled by application** is selected, the application program must activate the data exchange on the bus.



Note: The setting options under **Start of Bus Communication** for client specific variants of the configuration software can differ from the setting options displayed here.

5.3.1 Application Monitoring

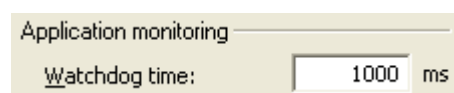


Figure 27: Master Settings > Application Monitoring

The **Watchdog time** determines the time within which the device watchdog must be re-triggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to 0 the watchdog is deactivated and the application program monitoring is deactivated too.

The permissible range of values of the watchdog time is 20 to 65535. By default the watchdog time value equals to 1000 ms.

Watchdog time	Range of Value / Default Value
Permissible range of values	20 ... 65535 ms
Default	1000 ms
The software watchdog is deactivated.	0 ms

Table 18: Range of Value / Value for the Watchdog time



Note: The setting options under **Application Monitoring** for client specific variants of the configuration software can differ from the setting options displayed here.

5.3.2 Process Image Storage Format

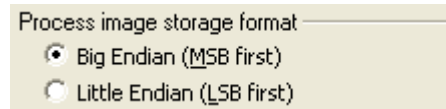


Figure 28: Master Settings > Process Image Storage Format

The **Process Image Storage Format** determines how the data words are stored in the process image.

For the data type Word it is possible to choose **Big Endian** or **Little Endian**.

Storage format (word module)	
Big Endian	MSB/LSB = higher/lower = Motorola format = Big Endian
Little Endian	LSB/MSB = lower/higher = Intel format = Little Endian

Table 19: Master Settings Pane Parameters - Process Image Storage Format



Note: The setting options under **Process Image Storage Format** for client specific variants of the configuration software can differ from the setting options displayed here.

5.3.3 Module Alignment

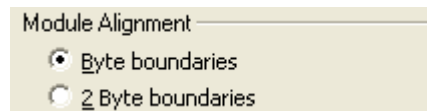


Figure 29: Master Settings > Module Alignment

The **Module Alignment** defines the addressing mode of the process data image. The addresses (offsets) of the process data are always interpreted as byte addresses. The **Module Alignment** then defines the addressing mode, **Byte boundaries** or **2 Byte boundaries**.

Parameter	Meaning
Byte boundaries	The module address can start at any byte offset.
2 Byte boundaries	The module address can only start at even byte offsets.

Table 20: Parameters Master Settings > Module Alignment



Note: The setting options under **Module Alignment** for client specific variants of the configuration software can differ from the setting options displayed here.

5.3.4 Process Data Handshake

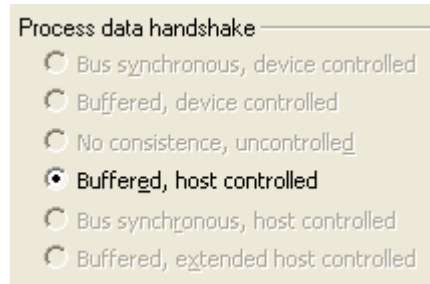


Figure 30: Master Settings > Process Data Handshake

The various types of **Process Data Handshakes** are used for setting the handshake of the process data for the CANopen Master device.

The selection of the used process data handshake is important for the correct data exchange between the application program and the device.

The used handshake of the process data needs to be supported by the used application program.

The **Buffered, host controlled** handshake is supported. The setting **No consistence, uncontrolled** works without handshake. The application program and the device access to the common memory (dual-port memory) without synchronization. This is leading to data inconsistency.



Note: The setting options under **Process Data Handshake** for client specific variants of the configuration software can differ from the setting options displayed here.

5.3.5 Advanced (For future Application)

The **Enable configuration download during network state “operate”** option for the CANopen network allows to change the configuration of a running CANopen network without resetting the devices.

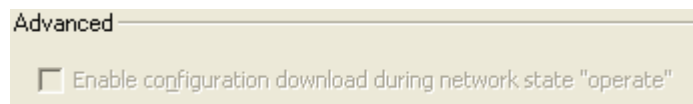


Figure 31: Master Settings > Advanced

➤ Check **Enable configuration download during network state “operate”** to enable configuration download during network state “operate”.

⇒ A new configuration is downloaded and the configuration of the CANopen network is changed and saved.



Note: The setting options under **Advanced** for customer specific variants of the configuration software can differ from the setting options displayed here.

5.3.6 Device Status Offset

Reference to Firmware: The option **Device Status Offset** was implemented since CANopen Master Firmware CFXCOM.NXF Version 2.4.1.0.

The option **Device Status Offset** allows via **Automatic calculation** to calculate the offset for the start address of the device status in the dual-port memory automatically or via **Static** to preset the offset.

Device status offset

☒ Automatic calculation

☐ Static: Starts bytes after last input data

Current offset address is:

Figure 32: Master Settings > Device Status Offset

Device Status Offset	Meaning
Automatic calculation:	Device status always after the last input byte. If further input data are added in the configuration, then the starting address of the device status in the dual-port memory will move.
Static:	Here, the distance (free buffer) between the last input byte and the start of the device status can be set. If further input data are added in the configuration, then the distance will be reduced, so that the start address of the device status in the dual-port memory remains the same. If more input data are added in the configuration as free buffer exists, then the start address of the device status in the dual-port memory must be moved.

Table 21: Option Master Settings > Device Status Offset



Note: The setting options under **Device Status Offset** for customer specific variants of the configuration software can differ from the setting options displayed here.

5.4 Bus Parameters

The **Bus Parameters** are the basis of an operating data exchange. This section contains information for setting the Bus Parameters as well as the description of the individual parameters.

Figure 33: Bus Parameters (Under Device description the symbolic name of the device is displayed.)



Note: Check that all CANopen Nodes also support the selected baud rate.



Basic rule: The baud rate must be set the same for all devices. The Node ID on the other hand must differ from Node to Node.

5.4.1 Device Description

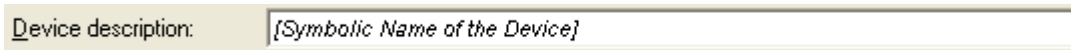


Figure 34: Bus Parameters > Device Description

The symbolic name of the device is displayed in the field **Device Description** and can be changed via entering.

5.4.2 Node Settings

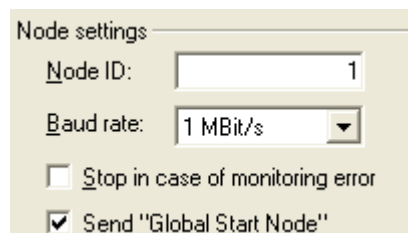


Figure 35: Bus Parameters > Node Settings

Bus Parameters	Meaning	Range of Value / Default Value
Node ID	The Node ID is necessary for the addressing of the device at the bus and has to be unique in the network. Therefore it is not allowed to use this number two times in the same network.	1 ... 127
Baud rate	The Baud rate is the data transfer speed: number of Bits per second.	10 kBit/s 20 kBit/s 50 kBit/s 100 kBit/s 125 kBit/s 250 kBit/s 500 kBit/s 800 kBit/s 1 Mbit/s
Stop in case of Monitoring Error	The Master will stop if a monitoring error (Node Guarding or Heartbeat Error) appears. This function will define the behavior of the Master if the communication is interrupted to at least one Node. If this function is enabled (basic setting: deactivated), the Master will also stop the communication to all further Nodes which were still responding and active. If this function is disabled, then the loss of communication to one Node will not influence the communication of the still present Nodes. For all the error affected Nodes the Master remains in the state to try the reestablishment of the communication again.	
Send "Global Start Node"	After the Master has started all configured Nodes individually, it will send a "Global Start Node" afterwards, in order to synchronize all Nodes again if this function is enabled.	

Table 22: Bus Parameters > Node Settings

5.4.3 Sync Master Settings

For the SYNC message sent from the CANopen Master **COB-ID** and **Cycle Period** can be set.

SYNC Master Settings

COB-ID: 128

Cycle Period: 100 ms

Figure 36: Bus Parameters > Sync Object

Sync Objekt	Meaning	Range of Value / Default Value
COB-ID	COB-ID of the SYNC message (synchronisation message)	0 ... 128, 1664 ... 1759, 1761 ... 1792, Default: 128
Cycle Period	Cycle Period of the SYNC message The value 0 for the Cycle Period deactivates the SYNC message sending.	0 ... 65535, Default 100

Table 23: Bus Parameters > Sync Object



Note: Cycle periods which are too small can overload the network by too many SYNC messages.

5.4.4 29 Bit COB-ID

Under **Enable 29 bit COB-ID** the size of the COB-ID can be set to 29 bit. The default setting is 11 bit.

29 Bit COB-ID

☐ Enable 29 bit COB-ID

Bit 28... Bit 0

Acceptance mask: 00 00 00 00

Acceptance code: 00 00 00 00

29 Bit COB-ID

☒ Enable 29 bit COB-ID

Bit 28... Bit 0

Acceptance mask: 00 00 00 00

Acceptance code: 00 00 00 00

Figure 37: Bus Parameters > 29 Bit COB-ID

- Check **29 bit COB-ID Aktiv**, to activate the 29 Bit identifier for the Master.

Acceptance Mask and Acceptance Code

To reduce the number of messages to be processed by the Master, a receiving filter can be defined by use of the Acceptance Mask and the Acceptance Code.

The **Acceptance Mask** defines which bits must be verified.

- Bits in the Acceptance Mask with the value 0 are „relevant“ and will be verified.
- Bits in the Acceptance Mask with the value 1 are „not relevant“ and will not be verified.

If a bit is verified, the **Acceptance Code** defines the value for the Message Bit to pass the filter.

Limits

- *Acceptance Mask = 1F FF FF FF:*
All COB-IDs will pass the filter.
- *Acceptance Mask = 00 00 00 00 & Acceptance Code = one COB-ID:*
Only this COB-ID will pass the filter.*

*This setting does not make any sense in practice, and serves only to describe the effect of the acceptance filter here.

Example

The acceptance filter is intended to ensure that messages pass the filter, which have lower 11 bits of any value and the upper 18 bits of a certain value.

These are the COB-IDs: 00000 00000000 00011XXX XXXXXXXX

Acceptance mask: 00 00 07 FF

Acceptance Code: 00 00 18 00

(Only) messages with the COB-ID 6144 ... 8192 can pass through the filter.

Diagram Acceptance Code / Acceptance Mask

The chart below shows the filter logic valid for all 29 bit of the input messages.

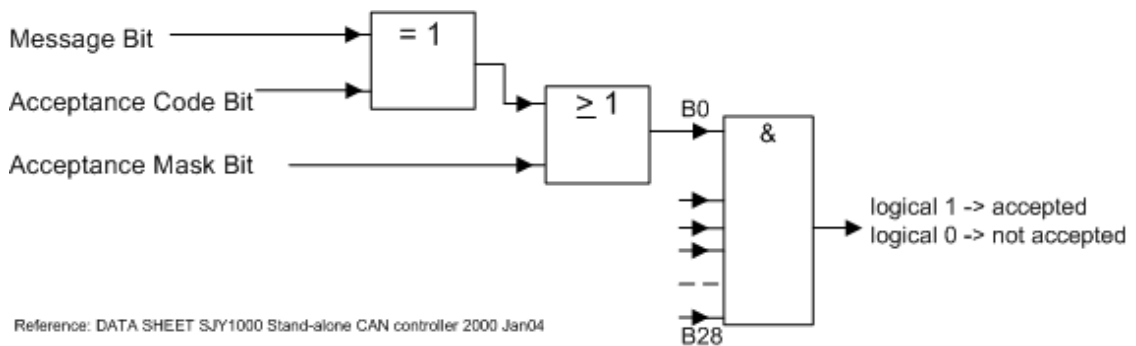


Figure 38: Diagram Acceptance Code / Acceptance Mask

“Message Bit” = one Bit of the received COB-ID

- *1. Examination of the individual bits:*
The Message bit and the Acceptance Code bit must be equal, e. i. both must have the value 0 or 1. Thereby the the tested bit gets the value 1. OR the Acceptance Mask bit must have a value equal to 1, then this bit will be accepted.
- *2. Examination of all 29 bits:*
The condition of the individual bits must be verified for all 29 bits, then the message can pass through the filter.

5.5 Process Data

For the CANopen Master DTM the **Process Data** pane serves as an external process data interface, e. g. for data transfer to a PLC unit. The process data pane lists the Slave devices connected to the Master, as well as the configured modules or input or output signals of the devices. This makes the fieldbus structure visible.

For the configured modules or signals names can be set (Column *Tag*).

In addition signal data available to the OPC server can be checked (Column *SCADA*).























Process Data				
		Typ	Tag	SCADA
		Slave Device*	Slave Device*	<input checked="" type="checkbox"/>
		2 word input con (0xD1) <Slot 1>	2 word input con (0xD1)	<input checked="" type="checkbox"/>
		(16 Bit) int input	INPUT_1	<input type="checkbox"/>
		(16 Bit) int input	INPUT_2	<input checked="" type="checkbox"/>
		8 byte input con (0x97) <Slot 2>	8 byte input con (0x97)	<input checked="" type="checkbox"/>
		(8 Bit) byte input	INPUT_1	<input type="checkbox"/>
		(8 Bit) byte input	INPUT_2	<input type="checkbox"/>
		(8 Bit) byte input	INPUT_3	<input checked="" type="checkbox"/>
		(8 Bit) byte input	INPUT_4	<input type="checkbox"/>
		(8 Bit) byte input	INPUT_5	<input type="checkbox"/>
		(8 Bit) byte input	INPUT_6	<input type="checkbox"/>
		(8 Bit) byte input	INPUT_7	<input checked="" type="checkbox"/>
		(8 Bit) byte input	INPUT_8	<input type="checkbox"/>
		1 byte input (0x10) <Slot 3>	1 byte input (0x10)	<input checked="" type="checkbox"/>
		(8 Bit) byte input	INPUT_1	<input checked="" type="checkbox"/>

Figure 39: Process Data (*The name of the Slave device is displayed.)

Column	Symbol	Meaning
Type	 device	Device labeling* provided by the hardware followed by the device's name of station in pointy brackets
	 module, submodule	Description of the modules or input or output signals configured to the device (not editable)
	 I/O signal	
Tag	 device	Symbolic name* of the device
	 module, submodule	Symbolic name for the modules or for the input or output signals configured to the device (editable)
	 I/O signal	
	 warning	Duplicate Tag at the same level can cause errors by use of OPC!
SCADA	Option which module or signal data shall be provided for the OPC server. „SCADA“ (= Supervisory Control and Data Acquisition), here used with the meaning „to provide for visualizing purposes“.	

*Depending on the protocol, either the device name or the symbolic name can be edited via the device symbol context menu.

Table 24: Process Data

5.6 Address Table

The **Address Table** dialog pane shows a list of all addresses used in the process data image. The displayed addresses refer to the used CANopen Master.

To configure the address data:

- Select **Configuration > Address Table** in the navigation area.

Address Table

Display mode: Hexadecimal CSV Export

Inputs:

Node ID	Device	Name	Obj. Idx	Obj. Name	COB-ID	Type	Length	Address
0x02	CIFX CO/C	CIFX CO/COS (C	0x1800	TxPDO_01	0x0182	IB	0x0008	0x0000
0x02	CIFX CO/C	CIFX CO/COS (C	0x1801	TxPDO_02	0x0282	IB	0x0008	0x0008
0x02	CIFX CO/C	CIFX CO/COS (C	0x1802	TxPDO_03	0x0382	IB	0x0008	0x0010
0x02	CIFX CO/C	CIFX CO/COS (C	0x1803	TxPDO_04	0x0482	IB	0x0008	0x0018
0x03	CIFX CO/C	CIFX CO/COS (C	0x1800	TxPDO_01	0x0183	IB	0x0008	0x0020
0x03	CIFX CO/C	CIFX CO/COS (C	0x1801	TxPDO_02	0x0283	IB	0x0008	0x0028
0x03	CIFX CO/C	CIFX CO/COS (C	0x1802	TxPDO_03	0x0383	IB	0x0008	0x0030
0x03	CIFX CO/C	CIFX CO/COS (C	0x1803	TxPDO_04	0x0483	IB	0x0008	0x0038

Outputs:

Node ID	Device	Name	Obj. Idx	Obj. Name	COB-ID	Type	Length	Address
0x02	CIFX CO/C	CIFX CO/COS (C	0x1401	RxPDO_02	0x0302	QB	0x0008	0x0000
0x02	CIFX CO/C	CIFX CO/COS (C	0x1402	RxPDO_03	0x0402	QB	0x0008	0x0008
0x02	CIFX CO/C	CIFX CO/COS (C	0x1403	RxPDO_04	0x0502	QB	0x0008	0x0010
0x03	CIFX CO/C	CIFX CO/COS (C	0x1400	RxPDO_01	0x0203	QB	0x0008	0x0018
0x03	CIFX CO/C	CIFX CO/COS (C	0x1401	RxPDO_02	0x0303	QB	0x0008	0x0020
0x03	CIFX CO/C	CIFX CO/COS (C	0x1402	RxPDO_03	0x0403	QB	0x0008	0x0028
0x03	CIFX CO/C	CIFX CO/COS (C	0x1403	RxPDO_04	0x0503	QB	0x0008	0x0030

Figure 40: Configuration > Address Table (In the Figure shown here, in the column Device or Name example devices are displayed.)

5.6.1 Description of the Parameters

Column	Meaning
Node-ID	Node-ID of the Slave device
Device	Device Name of the Slave device
Name	Designation for the Slave device
Obj. Idx	Object Index
Obj. Name	Object Name
COB-ID	COB-ID of the CAN message
Type	Input data type or output data type
Length	Input data length or output data length
Address	Output data offset address or input data offset address

Table 25: Parameters Address Table Pane - Inputs / Outputs

5.6.2 Display Mode, CSV Export

Display Mode

- Use the **Display Mode** drop-down list to select data display mode decimal or hexadecimal.

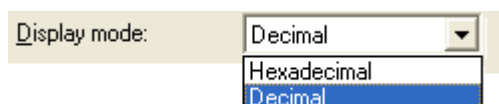


Figure 41: Configuration > Address Table - Display Mode

CSV Export

CSV Export allows to export input- and output addresses as CSV file (CSV = comma separated value). Therefore:

- Click **CSV Export**.
- A file saving dialog opens.
- Save the data as *.CSV file.

You can open the generated data by means of a spreadsheet application.

Sort Addresses

- To sort the address data, click on the respective column header.

5.7 Node ID Table

The **Node ID Table** shows a list of all Nodes configured in the Master.

Node ID Table					
	Activate	Node ID	Device	Name	Vendor
	<input checked="" type="checkbox"/>	2	CIF50-COS (C50COS.EDS)	CIF50-COS (C50COS.EDS)	[Name of the Manufacturer]
	<input checked="" type="checkbox"/>	3	CIF30-COS (C30COS.EDS)	CIF30-COS (C30COS.EDS)	[Name of the Manufacturer]

Figure 42: Node ID Table (In the Figure shown here, in the column Device or Name example devices are displayed.)

If the field **Activate** is checked, memory for process data of this Slave will be reserved in the Master and the Master will make a data exchange to this Slave on the bus.

If this setting is deactivated, the Master will reserve memory in the process data image for this Slave, but no data exchange to this Slave will be made at the bus.

The column **Node ID** shows the addresses of the devices in the network. The address of the selected device can be changed by clicking on it.

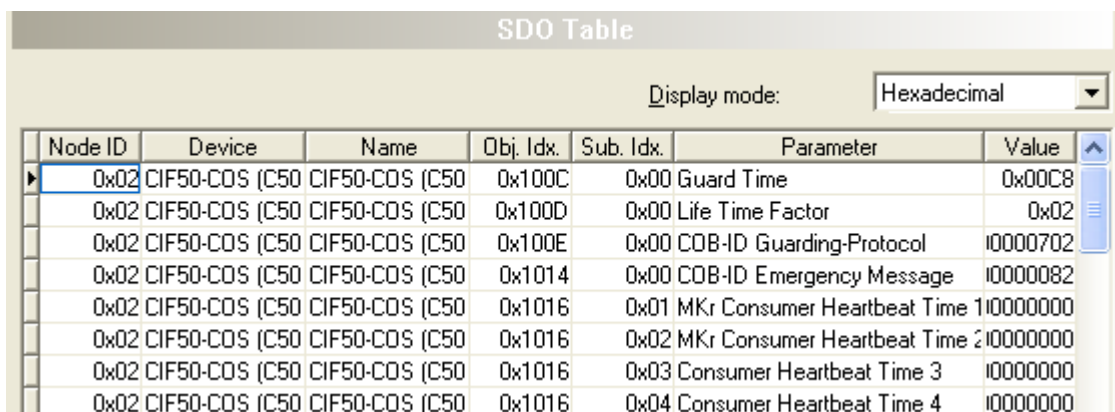
The columns **Device**, **Name** and **Vendor** are not editable and show the following information:

Column	Meaning
Device	name of the device from the EDS file
Name	symbolic name of device
Vendor	vendor of device

Table 26: Node ID Table

5.8 SDO Table

The **SDO Table** shows an overview of the transmitted objects during the Node BootUp phase for each Node. Apart from the **Node ID** for each entry the **Object-** and **Sub index** is displayed with parameter name and the pertinent value.



Node ID	Device	Name	Obj. Idx.	Sub. Idx.	Parameter	Value
0x02	CIF50-CDS (C50	CIF50-CDS (C50	0x100C	0x00	Guard Time	0x00C8
0x02	CIF50-CDS (C50	CIF50-CDS (C50	0x100D	0x00	Life Time Factor	0x02
0x02	CIF50-CDS (C50	CIF50-CDS (C50	0x100E	0x00	COB-ID Guarding-Protocol	0000702
0x02	CIF50-CDS (C50	CIF50-CDS (C50	0x1014	0x00	COB-ID Emergency Message	0000082
0x02	CIF50-CDS (C50	CIF50-CDS (C50	0x1016	0x01	MKr Consumer Heartbeat Time 1	00000000
0x02	CIF50-CDS (C50	CIF50-CDS (C50	0x1016	0x02	MKr Consumer Heartbeat Time 2	00000000
0x02	CIF50-CDS (C50	CIF50-CDS (C50	0x1016	0x03	Consumer Heartbeat Time 3	00000000
0x02	CIF50-CDS (C50	CIF50-CDS (C50	0x1016	0x04	Consumer Heartbeat Time 4	00000000

Figure 43: SDO Table (In the Figure shown here, in the column Device or Name example devices are displayed.)

Column	Meaning	Range of Value / Default Value
Node ID	Station address of the Slave device in the network	1 ... 127
Device	Name of device	from EDS file
Name	Symbolic name of device	(defined by the user)
Obj. Idx.	Object index of a SDO Note: The object index states the mandatory and optional indexes.	from EDS file
Sub. Idx	Sub index	from EDS file
Parameter	Parameter name identified by object index and sub index)	from EDS file
Value	Value of the Parameters (set in Slave DTM or in the CAN-ID table)	

Table 27: SDO Table

Display Mode

- Use the **Display Mode** to select data display mode decimal or hexadecimal.

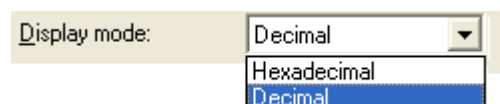


Figure 44: SDO Table > Display Mode

5.8.1 Mapping Method for PDO and PDO-Mapping

The mapping method for PDO and PDO mapping is implemented according to the DS301 V4 specification.

At first the configured PDO are deactivated and parameterized.

At the PDO mapping the mapped information in the node first is erased, then rewritten and finally set to valid.

After the PDO mapping the configured PDO will be reactivated if in the configuration the PDO is set to “exist”.

Mapping Method for the SDO Table	Description
In object 0x14xx (and 0x18xx) sub index 1 (COB-ID) Bit 31 is set	The configured PDO is deactivated
Sub index .02 etc are written	These PDO are parameterized
To delete the information of the PDO mapping in the node (and to set it back to the default mapping), in object 16xx (and object 1Axx) sub index 0 the value 0 is written.	PDO Mapping: The mapped information in the node are deleted
The objects to be mapped are written to object 16xx (and object 1Axx) sub index 1 to N.	PDO Mapping: The mapped information in the node are rewritten
The number of objects (value N) is written to object 16xx (and object 1Axx) sub index 0.	PDO Mapping: Finally the mapped information in the node are reset to valid
In object 0x14xx (and 0x18xx) sub index 1 (COB-ID) Bit 31 will be set to 0 if in the configuration the PDO is set to “exist”.	The configured PDO will be reactivated if in the configuration the PDO is set to “exist”.

Table 28: Mapping Method according to Specification DS301 V4

5.9 CAN-ID Table

In the **CAN-ID Table** a sorted list for each Node is shown, which message numbers (CAN-IDs) in the CAN network are occupied by the respective Nodes.

CAN-ID Table					
				Display mode:	Hexadecimal
Node ID	Device	Name	Message Type	CAN-ID	Auto Alloc
0x02	CIFX CO/COS (CIFX CO COS.ed\$	CIFX CO/COS (CIFX CO COS.ed\$ SYNC		0x00000080	<input type="checkbox"/>
0x02	CIFX CO/COS (CIFX CO COS.ed\$	CIFX CO/COS (CIFX CO COS.ed\$ TIME_STAMP		0x00000100	<input checked="" type="checkbox"/>
0x02	CIFX CO/COS (CIFX CO COS.ed\$	CIFX CO/COS (CIFX CO COS.ed\$ EMCY		0x00000082	<input checked="" type="checkbox"/>
0x02	CIFX CO/COS (CIFX CO COS.ed\$	CIFX CO/COS (CIFX CO COS.ed\$ RxPDO1		0x000002F2	<input checked="" type="checkbox"/>
0x02	CIFX CO/COS (CIFX CO COS.ed\$	CIFX CO/COS (CIFX CO COS.ed\$ RxPDO2		0x00000302	<input checked="" type="checkbox"/>
0x02	CIFX CO/COS (CIFX CO COS.ed\$	CIFX CO/COS (CIFX CO COS.ed\$ RxPDO3		0x00000402	<input checked="" type="checkbox"/>
0x02	CIFX CO/COS (CIFX CO COS.ed\$	CIFX CO/COS (CIFX CO COS.ed\$ RxPDO4		0x00000502	<input checked="" type="checkbox"/>
0x02	CIFX CO/COS (CIFX CO COS.ed\$	CIFX CO/COS (CIFX CO COS.ed\$ TxPDO1		0x00000182	<input checked="" type="checkbox"/>
0x02	CIFX CO/COS (CIFX CO COS.ed\$	CIFX CO/COS (CIFX CO COS.ed\$ TxPDO2		0x00000282	<input checked="" type="checkbox"/>
0x02	CIFX CO/COS (CIFX CO COS.ed\$	CIFX CO/COS (CIFX CO COS.ed\$ TxPDO3		0x00000382	<input checked="" type="checkbox"/>
0x02	CIFX CO/COS (CIFX CO COS.ed\$	CIFX CO/COS (CIFX CO COS.ed\$ TxPDO4		0x00000482	<input checked="" type="checkbox"/>

Figure 45: CAN-ID Table (In the Figure shown here, in the column Device or Name example devices are displayed.)

Column	Meaning	Range of Value / Default Value
Node ID	Station address of the Slave device in the network	1 ... 127
Device	Name of device from EDS file	from EDS file
Name	Symbolic name of device	(defined by the user)
Message Type	Indication of the message type	NODE GUARDING, EMCY, RXPDO[x], TXPDO[x]
COB-ID	Communication Object Identifier. If Auto Alloc is not checked, the COB-ID can be spezified manually.	For CAN-ID: 11 bit: 0x00000000 ... 0x000007FF (0 ... 2047); 29 bit: 0x00000000 ... 0x1FFFFFFF (0 ... 536870911)
Auto Alloc	Activate / deactivate automatic allocation If Auto Alloc is checked, the schema of the „pre-defined connection set“ of the CANopen specification ([2] page 86, Table 39) is valid.	checked, not checked, Default: checked

Table 29: CAN-ID Table

Display Mode

- Use the **Display Mode** to select decimal or hexadecimal data display mode.

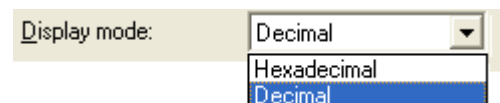


Figure 46: COB-ID-Table> Display Mode

5.10 Node BootUp Mapping Sequence

The **Node BootUp** mapping sequence defines the network startup behavior of the Master for the particular Node to get it operative. There are different states a Master is running through per Node, till the BootUp sequence is finished for the Node.

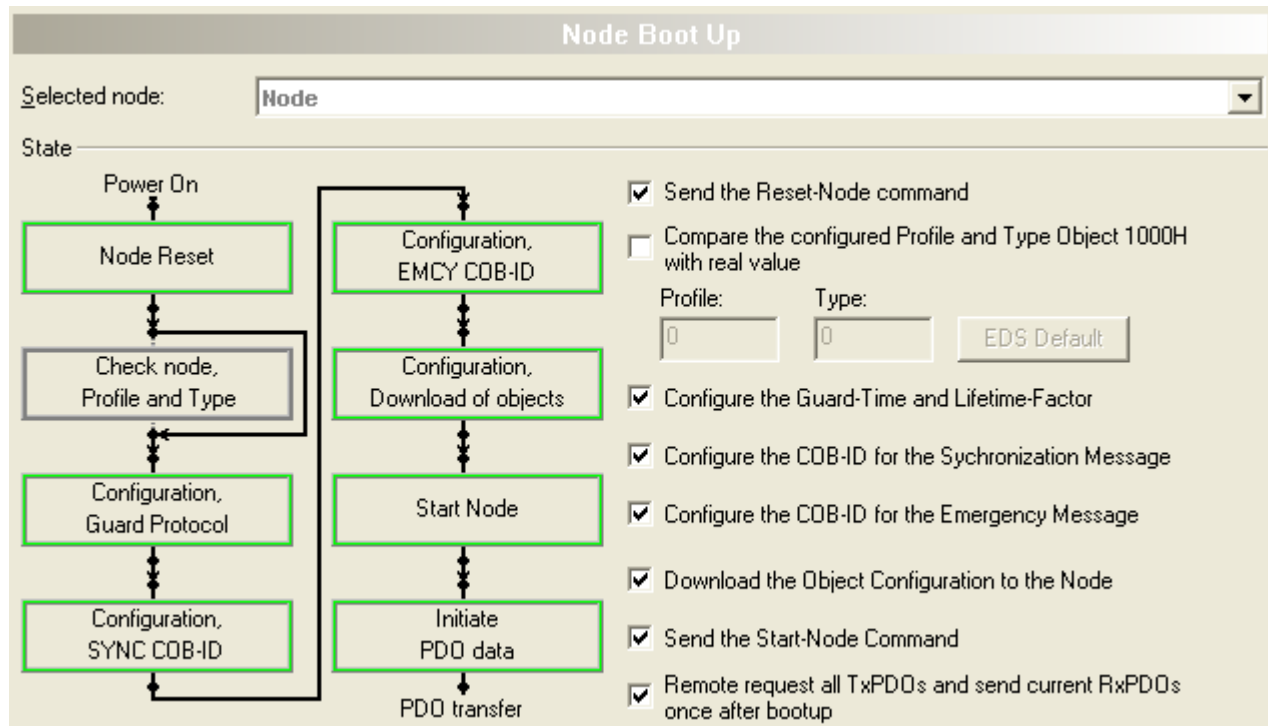


Figure 47: Node BootUp Mapping Sequence

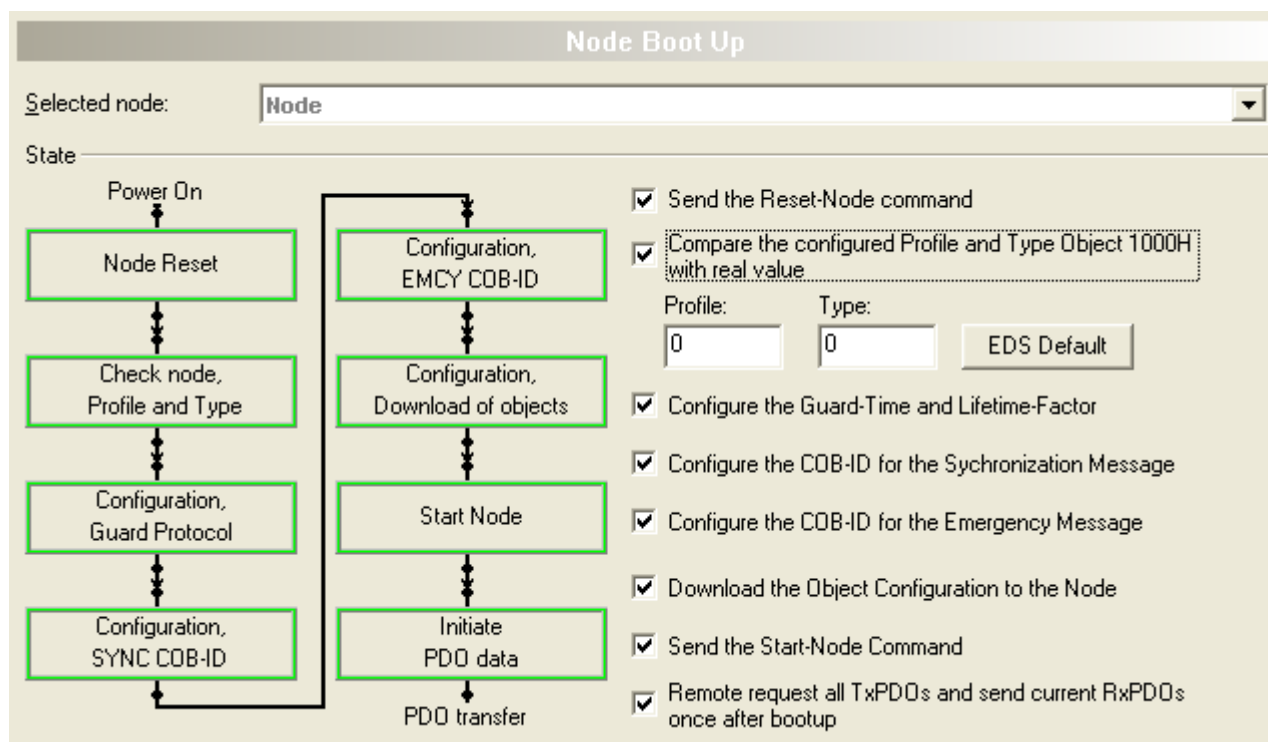


Figure 48: "Compare the configured Profile and Type Object 1000H with real value" enabled



Node: If there are no Slaves configured at the CANopen Master, all options of the **Node Boot Up** pane will be deactivated.

Selected Node

Actual selected node: List field to select the node to configure its boot up.

Status

Each status for the Node BootUp is configurable and can be enabled (checked) or disabled (unchecked).

Status	Description
Node Reset	If Send the Reset-Node command is enabled, the Master will send at first the CANopen specific Node reset communication command.
Check node, Profile and Type	<p>If Compare the configured Profile and Type Object 1000H with real value is enabled, the Master will compare the content of the mandatory Node device type object 1000H with the values that are configured within SYCON.net. If the values do not match, the Master will refuse the access to the nodes and will report a parameterization error.</p> <p>The option must be enabled if a Node object is not defined in the EDS file. In this case enter under Profile and Type the real values for the profile and for the device type of the Node according to the vendor's data.</p> <p>EDS Default resets the values for the profile and the device type of the Node to the original EDS values and deactivates the comparison for the object 1000H.</p>
Configuration, Guarding Protocol	<p>A CANopen has two specific registers responsible for the Node guarding protocol.</p> <p>If Configure Guard-Time and Lifetime-Factor is enabled, the Master will write the Guard Time factor and Life-Time factor of the Node configuration during startup into the corresponding Node objects.</p>
Configuration, SYNC COB-ID	If Configure the COB-ID for the Synchronization Message (Synchronization Message) is enabled, the Master will write the SYNC COB-ID of the Node configuration during startup into the corresponding Node objects.
Configuration, EMCY COB-ID	If Configure the COB-ID for the Emergency Message (Synchronization Message) is enabled, the Master will write the EMCY COB-ID of the Node configuration during startup into the corresponding Node objects.
Configuration, Download of objects	<p>To get a PDO communication to a Node working, the Master must send all relevant configuration objects to the Node. This includes for example data about the PDO mapping and the COB-IDs of the send PDOs and the receive PDOs.</p> <p>If Download the Object Configuration to the Node is enabled, the Master will download all these parameters and all configured objects from the Node configuration to the Node.</p>
Start Node	<p>To achieve the operational state in CANopen a Node must get the CANopen specific 'Start Node' command.</p> <p>If Send the Start Node Command is enabled, at the end of the boot-up procedure the Master will send the start Node command to the Node.</p>
Initiate PDO data	If Remote request all TxPDOs and send current RxPDOs once after bootup is enabled, the Master will automatically read and write the configured PDOs once directly after the startup. This ensures that the Master sends all current data of its process output data area to the Node and reads all current data from the Node and stores them to the Master's process input data area.

Table 30: Node BootUp Mapping Sequence

5.11 Monitoring

Under **Monitoring** the device monitoring is configured:

- Master monitors the single Nodes.
- The single Node monitors the Master.
- A Node monitors another Node.

For this purpose for every node the 'Node-Guarding' or 'Heart-Beat' protocol can be used in a different way.



Note: To be able to use a monitoring function, the node must support 'Node-Guarding' or 'Heart-Beat'.

Selected Node contains all configured nodes. For monitoring the **Use Node Guarding Protocol** option or the **Use Heartbeat Protocol** option must be selected.

- For **Node Guarding Protocol** under **Selected Node** select the node to be monitored and configure the **Guard time** and the **Life time factor**.
- For **Heartbeat Protocol** under **Selected Node** select the producer node. The other configured nodes are listed in the **Consumer** table as consumers. Under **Max consumer** the maximum amount of consumer is indicated.

Figure 49: Monitoring

Consumer Table

Consumer:

	Active	Node ID	Device	Name	Vendor	Consumer Time [ms]
	<input checked="" type="checkbox"/>	1	NETX-COM	CIF50-COM	{Manf. Name}	300
	<input type="checkbox"/>	3	CIF30-COS (C30COS	CIF30-COS (C30COS	{Manf. Name}	300

Figure 50: Monitoring – Consumer (In the Figure shown here, in the column Device or Name example devices are displayed.)

Column	Meaning
Active	Check to activate the monitoring of this node. Uncheck to deactivate the monitoring of this node.
Node ID	Station address of the slave device in the network
Device	Name of the device from the EDS file
Name	Symbolic name of the device
Vendor	Vendor of the device
Consumer Time [ms]	Time the node is supervised by the Master

Table 31: Consumer Table

5.11.1 Node Guarding Protocol

Functional Principle:

The Master sends cyclically polling messages (Remote Request) to the Node (Node Guarding), to check, whether the Node still exists on the bus. The Node sends its actual state as answer back to the Master. The Nodes can use the poll messages of the Master to supervise the Master on its part (Life Guarding).

Requirement:

The Node has to support the Node Guarding Protocol.

Settings:

The **Guarding Time** is the setting for the supervision of the Node in the view of the Master (Master controls Node). If the communication is running, the Master will poll the Node in the time interval Guarding Time, to check, whether the Node is still present in the network or not. If this value is configured with 0, the supervision will be disabled in the Master as well as in the Node.

The **Life Time Factor** is the setting for the supervision of the Master in the view of the Node (Node monitors Master). If the communication is running, the Node will monitor the Master in the Guarding Time multiplied with the Life Time Factor as time interval, to check, whether the Master is still present in the network or not. If this value is configured with 0, the supervision in the Node will be deactivated.



Important: To reach a stable communication of the Node on the CANopen, the Life Time Factor has to be set to minimal 2.



Note: A Life Guarding can only be used if the Master carries out a Node Guarding that means Life Guarding presumes Node Guarding.

5.11.2 Heartbeat Protocol

Functional Principle:

A Heartbeat Producer transmits the Heartbeat message cyclically with the time interval defined in the field **Producer Time**. One or more Heartbeat Consumer may receive the indication. The relationship between producer and consumer is configurable via the Object Dictionary entries.

Requirements:

In order to use the Heartbeat Protocol, the following requirements must be fulfilled:

- 1) In the Master device the firmware version V1.070 or higher has to be loaded.
- 2) The Node has to support the Heartbeat Protocol. (EDS object 1016).

Settings:

The table **Consumer** shows any consumer. Via **Selected node** consumer can be added to the table. Via **Active** the consumer to be monitored can be selected.



Note: The configuration of the consumer and the producer has effect to the objects 1016 (Consumer Heartbeat Time) and 1017 (Producer Heartbeat Time) of the object dictionary.

6 Online Functions

Getting Access to SYCON.net Online Functions



Note! If under 'Configuration' > 'Master Settings' > 'Start of bus communication' > 'Controlled by application' is selected, after 'Power on Reset' the Master device is in 'Offline' state! In this state you can not use the SYCON.net online functions 'Debug Mode' and 'Network Scan'. To get access to the online functions in case the Master device is used together with an application program, you must make sure that the application program will start the communication. Alternatively you can start the communication manually from the context menu 'Start communication' of the the Master device.

6.1 Connecting/Disconnecting Device



Note: Several CANopen Master DTM functions e. g. **Diagnosis** or the configuration download in the FDT Framework require an online connection from the CANopen Master DTM to the CANopen Master device.

Connecting Device

The following steps are needed to establish a connection from the CANopen Master DTM to a CANopen Master device:

Under **Settings** in the **Driver** pane:

1. Verify that the default driver is checked and respectively check another or multiple drivers.
2. Configure the driver if necessary.

Under **Settings** in the **Device Assignment** pane:

3. Scan for the devices (with or without firmware).
4. Select the device (with or without firmware) and apply the selection.



Before you download the firmware adhere to the necessary safety precautions to prevent personnel injury and property damage. For more refer to section *Safety Messages on Firmware or Configuration Download* on page 27.

Under **Settings** in the **Firmware Download** pane if not yet a firmware was loaded to the device:

5. Select and download the firmware.

Under **Settings** in the **Device Assignment** pane if not yet a firmware was loaded to the device:

6. Scan for the device (with firmware) once more.
7. Select the device (with firmware) once more.



An overview of the descriptions for these steps you find in the section *Overview Settings* on page 28.

8. In the DTM interface dialog select the **OK** button, to apply the selection and to close the DTM interface dialog.
 9. Put a right-click on the CANopen Master device icon.
 10. Select the **Connect** command from the context menu.
- The CANopen Master device now is connected to the CANopen Master DTM via an online connection. In the network view the device description at the device icon of the Master is displayed with a green colored background.

Disconnecting Device

To disconnect an online connection from the CANopen Master device to a CANopen Master DTM take the following steps:

1. In the DTM interface dialog select the **OK** button, to close the DTM interface dialog.
 2. Right-click on the CANopen Master device icon.
 3. Select the **Disconnect** command from the context menu.
- In the network view the device description at the device icon of the Master is not any more displayed with a green colored background. Now the CANopen Master device is disconnected from the DTM.



Important: For a 2-channel device channel 1 or channel 2 each must be connected to the DTM separately.




6.2 Debug Mode







Note: Depending by the software variant of the frame application the **debug mode** is available or not.

The **debug mode** allows identifying the status of the cyclical communication between a Master device and its Slave devices on a network based on the colors of the bus lines as well as the debug icons.

For the Master device or the Master bus line this is valid:

- Master device in operation, cyclical communication runs. (Bus line **light green**/debug icon „RUN“  next to the Master device icon)
- Master device not operable.
(Bus line **blue**/debug icon „ATTENTION“  next to the Master device icon)
- Master in STOP state.
(Bus line **red**/debug icon „STOP“  next to the Master device icon)

For the Slave device or the bus line from the Master bus line to the Slave device this is valid:

- Slave device in operation, cyclical communication to the Master device runs. (Bus line **light green**/debug icon „RUN“  next to the Slave device icon)
- Diagnosis message available at the Master device.
(Bus line **yellow** (yellow)/debug icon „RUN“  next to the Slave device icon)
- Slave device not found during cyclical communication boot up.
(Bus line **blue**/ debug icon „ATTENTION“  next to the Slave device icon)
- Error in the Slave-to-Master communication.
(Bus line **red**/debug icon „STOP“  next to the Slave device icon)
- Slave device is not configured.
(Bus line **gray**)

6.2.1 Requirements



Note: You first must:

- assign the Master device to the Master DTM,
- configure the Master or the Slave device parameters,
- and download the configuration to the Master.

Only then the debug mode can be used appropriately.



For details to the device assignment, refer to section *Settings for Driver and Device Assignment* on page 29.

For information on how to configure the Master device or how to download the configuration, refer to this Operating Instruction Manual and to the Operating Instruction Manual for the netDevice.

For information on how to configure the Slave device, refer to the Operating Instruction Manual of the Slave DTM.

6.2.2 Starting Debug Mode




Note: The menu entries for the **debug mode** will be only available if the debug mode is supported by the frame application and the Master DTM.

For a single network:

- Therefore open the context menu of the Master via right mouse click to the device icon.
- Select the **Start Debug Mode** command from the context menu.

For the entire project:

- Click to the menu **Network > Start Project Debug Mode** or in the netDevice toolbar **Debug** to the icon .
- In the debug mode, the bus lines in the network view are displayed depending on the status of the cyclical communications in various colors. Additionally next to the device icon different debug icons are displayed.

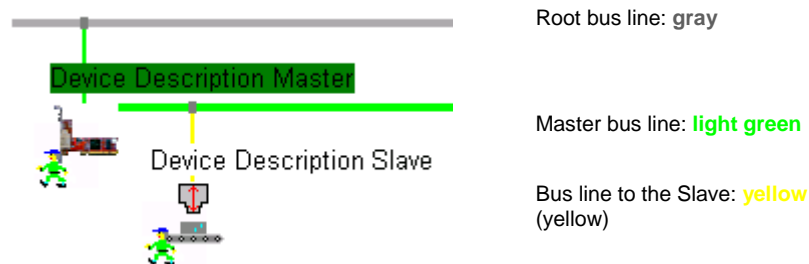


Figure 51: Example Debug Mode with pending Diagnosis, Network View of a Project with one Network (Master and Slave)

- **Root Busline:** This busline is displayed always in **gray**.
- **Branch Line of the Master device** (Busline from the Root to the Master Busline): During the debug mode this busline is displayed **light green**, **blue** or **red** colored. If the debug mode is not used, this bus line will be displayed **gray** colored.
- **Master Busline or Branch Line of the Slave device:** During the debug mode these buslines are displayed **light green**, **yellow** (yellow), **blue** or **red** colored. If the debug mode is not used, this bus line will be displayed in the respective *fieldbus* or *protocol specific* color (PROFIBUS: **violet**).

6.2.3 Debug Mode Busline Colors and Debug Icons

The following table contains information about the displayed colors of the bus lines and on the icons in the debug mode.








Debug Mode Busline Colors	Debug Icon	Meaning
Master Busline	Next to the Master device icon	
light green		The Master device has cyclic communication
blue		The Master device is not operable. This may have different causes. For example: - There is no valid license in the Master device. - There is no valid firmware in the Master device.
red		The Master device is in STOP state. The cyclic communication has been stopped.
Branch Line of the Slave device	Next to the Slave device icon	Meaning
light green		The Master device has cyclic communication to this Slave device.
yellow (yellow)		The Master device has cyclic communication to this Slave device, but in the diagnostic buffer of the Master device a diagnostic information about this Slave is pending.
blue		The device was not found. This may have different causes. For example: - The configuration download to the Master device is still missing. - The Slave device is not available in the network. - There is no valid firmware in the Master device. - Parameter or configuration error at the Master device. - The cyclic communication to this Slave device has been stopped (without diagnosis information at the Master).
red		Due to a communication error, the cyclic communication from the Master device to this Slave device is not possible. This may have different causes. For example: - The cyclical communication to this Slave device has been stopped. - Parameter or configuration error at the Slave device. - The validation of the manufacturer or device ID shows different values. - The diagnostic buffer of the Master device still a diagnostic information about this Slave is pending and the cyclic communication to this Slave device has been stopped.
gray	-	The Slave device is not configured. I. e., in the configuration of the Master device there are no configuration parameters to this Slave available.

Table 32: Debug Mode Busline Colors and Debug Icons

6.2.4 Reset of the Diagnosis Information and of the Station Status

To analyze the diagnosis information:

- Select the diagnosis windows in the Master DTM dialog.
- Therefore select **Diagnosis** > [*diagnosis window*] in the navigation area.

To read the diagnosis buffer of the Master device and thus to reset the device:

- Select **Diagnosis** > **Station diagnosis** in the navigation area.
- In the window **Station diagnosis** click with the right mouse button to the station status LED for the device (yellow).
- From the context menu select **Reset** or **Reset All**.

In the window **Station diagnosis** the station status LED for the device is displayed in green and the network will display the bus line from the Master device icon to the Slave device icon in **light green**.

6.2.5 Stopping Debug Mode

For a single network:

- Therefore open the context menu of the Master via right mouse click to the device icon.
- Select the **Stop Debug Mode** command from the context menu.

For the entire project:

- Click to the menu **Network** > **Stop Project Debug Mode** or in the netDevice toolbar **Debug** to the icon .

6.3 'Network Scan'

Reference to Firmware: The option **Network Scan** was implemented since CANopen Master Firmware CIFXCOM.NXF Version 2.4.1.0.

With the function **Network Scan...** of the CANopen Master DTM you can find out automatically which CANopen Slaves are attached to the CANopen Master device. During the network scan the Master device requests the ident codes of the Slave devices found at the bus. For each connected Slave device its ident code is read out.

In the **Scan Response** dialog of the Master DTM the assigned device description files or DTM devices are displayed. Exactly one ident code is assigned to each device description file and to each DTM device. Different versions (also language versions) of the same device description file are defined by the same ident code. For every identified device you can select the assigned DTM device according to the firmware loaded in that Slave device. Via **Create Devices** for each Slave device the selected DTM device is created.

Requirements

The CANopen Master device must be configured. Under **Settings > Bus Parameters** you must set the baud rate and the MAC ID of the Master.



Important: The configuration of the Master device must be downloaded in the Master device. For more see section *Configuration Steps* on page 23.

Overview Steps

1. Starting the **Network Scan** function of the Master DTM.
2. Make the settings in the **Scan Response** dialog of the Master DTM.
3. Click **Create Devices**.

For all Slave devices:

4. Via the **Download** function of the Master DTM, download the current configurations of the Slave devices to the Master device.

6.3.1 Starting 'Network Scan'

1. Starting the **Network Scan** function of the Master DTM.

- In netDevice: right-click on the device symbol of the CANopen Master DTM.
- Select **Network Scan...** from the context menu.

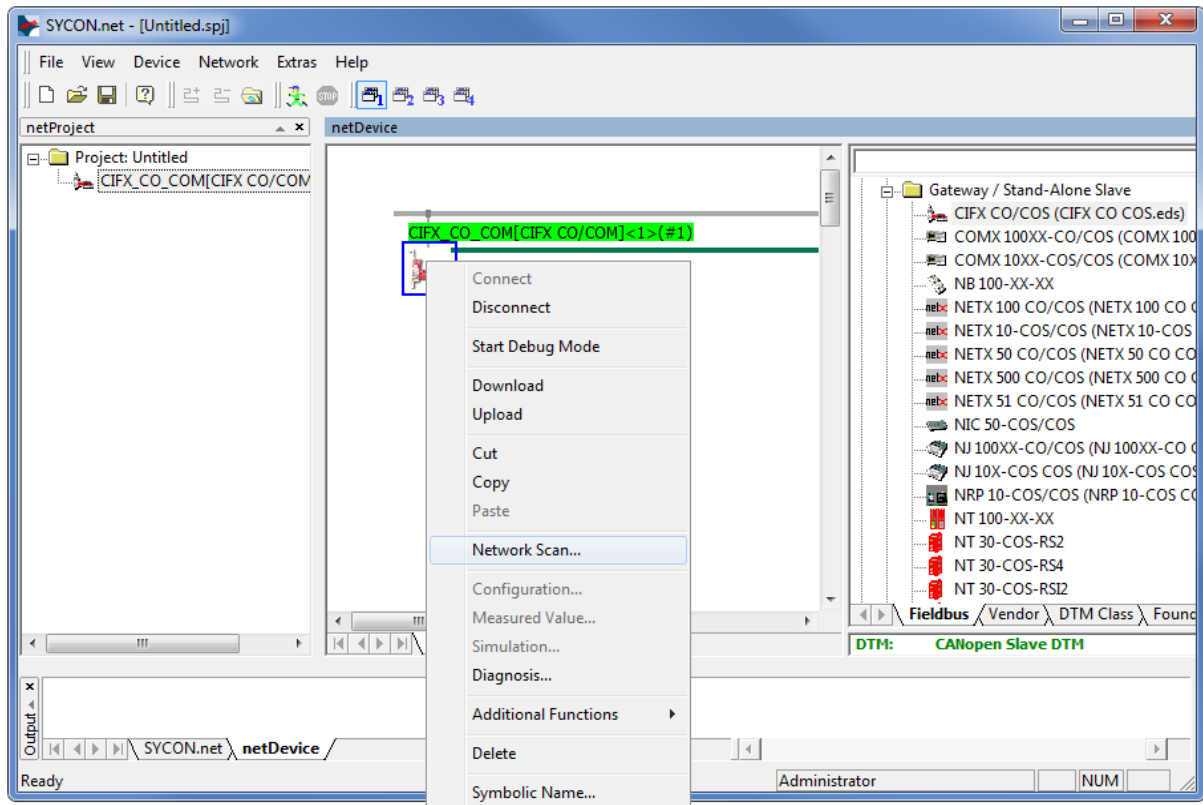


Figure 52: Starting 'Network Scan' (Example)

- Wait for a short time.



Note: It can last some seconds, until the **Network Scan...** dialog is displayed.

Via **Network Scan...** an online connection from the CANopen Master DTM to the CANopen Master device is established. The configuration software scans, which CANopen Slave devices are attached to the PROFIBUS network or to the CANopen Master device.

➤ The **Scan Response** dialog of the Master DTM is displayed.

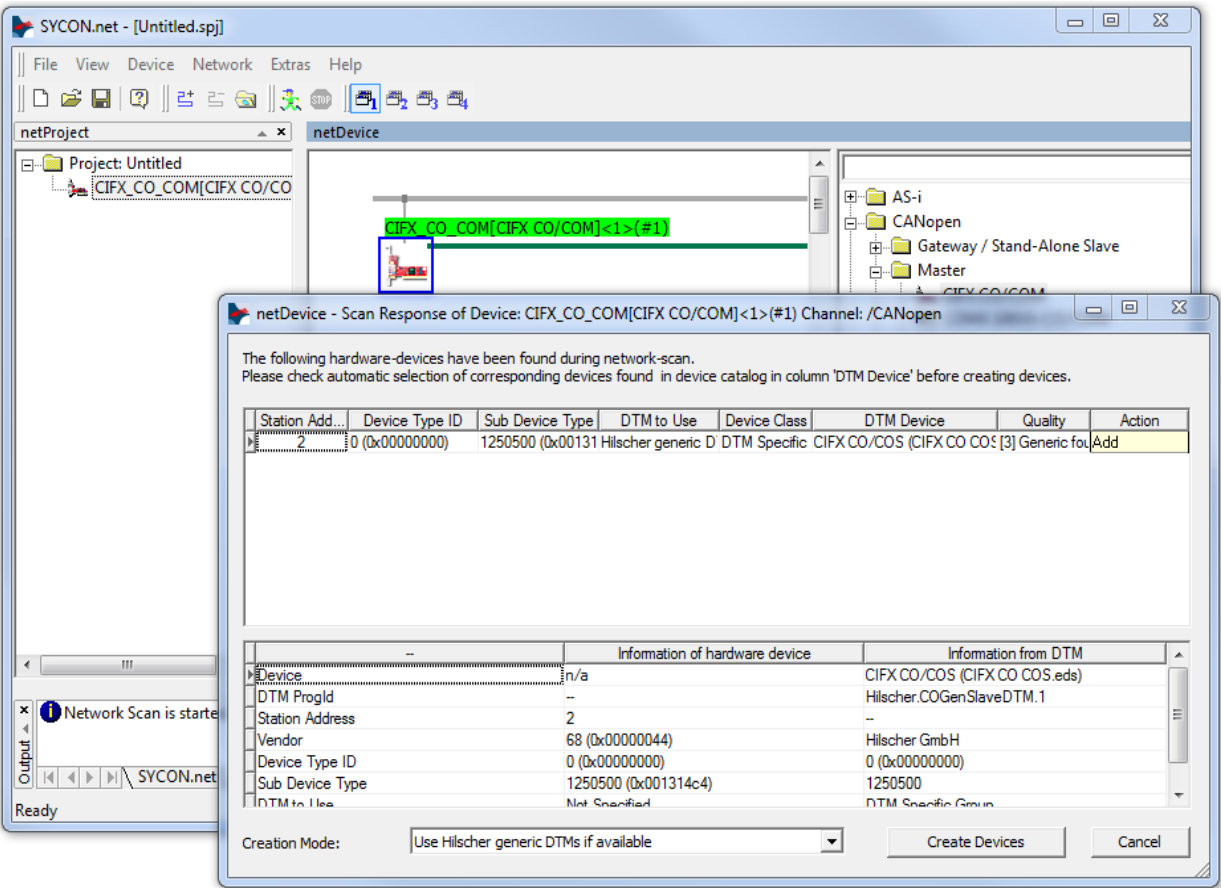


Figure 53: Scan Response dialog of the Master DTM (Example)

6.3.2 Settings in the Scan Response Dialog of the Master DTM

2. Make the settings in the **Scan Response** dialog of the Master DTM.

- In the **DTM to Use** column the DTM devices assigned to the found ident codes appear.

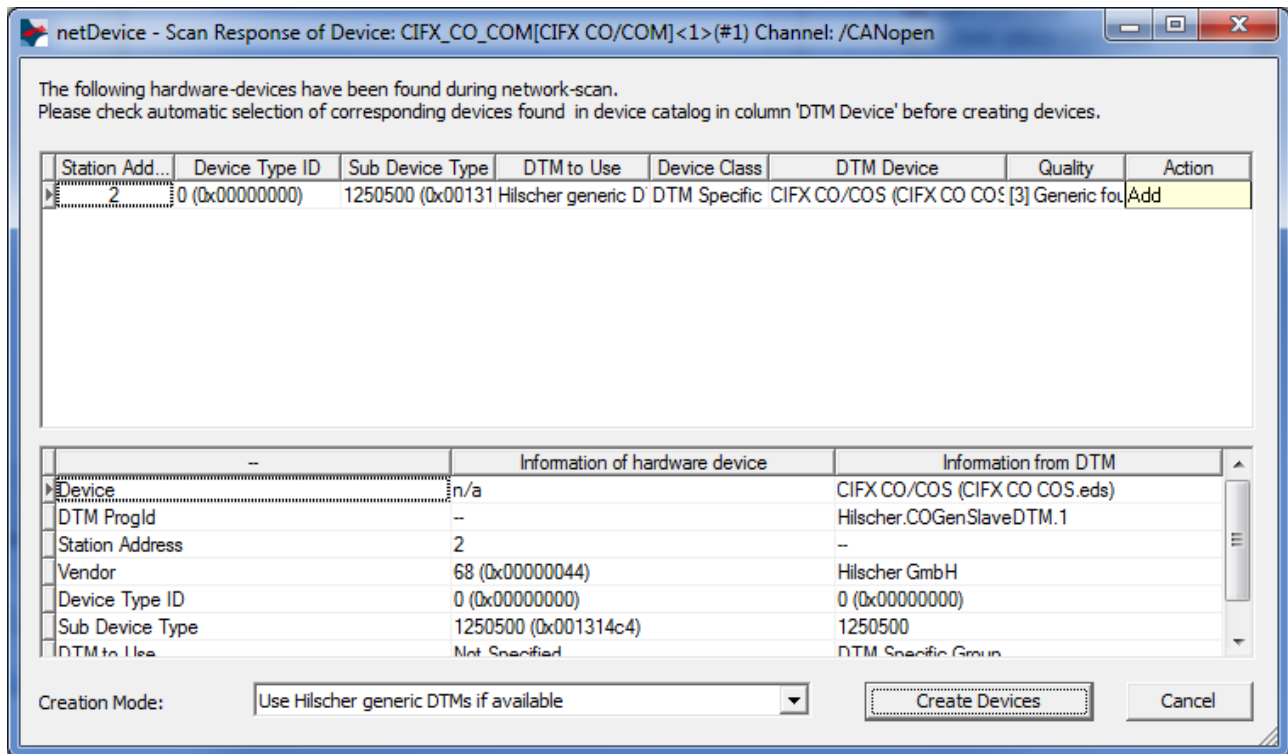


Figure 54: Scan Response dialog of the Master DTM (Example)

- In the **DTM to Use** column select for every identified device the DTM device corresponding to the firmware loaded in this Slave device.
- If in the **DTM to Use** column no DTM device or a DTM device not desired is displayed, add the required DTM devices to the device catalog.
- or adapt the creation mode under **Creation Mode**.
- In the **Action** column select, whether the found DTM device shall be:
 - *added or skipped*
(if a device is not yet present in the project),
 - *replaced or skipped*
(if a device is already present in the project)

6.3.3 Description on the Scan Response Dialog of the Master DTM

In the following table you find a description about the **Scan Response** dialog of the Master DTM.

Column	Description
Title Bar	With the text: <i>Symbolic Name of the Master Device [Device Description] <Device Address> (#Network ID) channel//CANopen.</i>
Instruction	In the Network Scan window the instruction text is displayed: The following hardware-devices have been found during network scan. Please check automatic selection of corresponding devices found in device catalog in column 'DTM Devices' before creating devices.
Station Address	CANopen station address, which displays the logical sequence of the devices within a CANopen network. <i>Note:</i> The station address displayed here is not identical to the station address of the device stored in the hardware configuration.
Colors	Meaning of colors in the Scan Response dialog of the Master DTM: <div style="display: flex; align-items: flex-start;"> <div style="margin-right: 10px;"> <div style="background-color: red; width: 20px; height: 15px; display: inline-block;"></div> Red <div style="background-color: yellow; width: 20px; height: 15px; display: inline-block;"></div> Yellow </div> <div> <p>If a field marked in red appears in column Station address, the respective DTM device is already present on the network.</p> <p>If a field appears marked in yellow, a selection can be made by a combo box.</p> </div> </div>
Device Type ID	Identification (ID): Ident code read out from each device (Unique Identifier)
Sub Device Type	Sub type of the device type if applicable
DTM to Use	<p>Display of the DTM devices, which are assigned to the ident codes found during scanning:</p> <p>If <u>Use Hilscher generic DTMs if available</u> is displayed <i>without color marking</i>, there is no selection possibility.</p> <p>If <u>Use Hilscher generic DTMs if available</u> is displayed <i>marked in yellow</i>, the following selection can be made:</p> <div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px;"> <div style="background-color: #f0f0f0; padding: 2px;">Use Hilscher generic DTMs if available</div> <div style="background-color: #0000ff; color: white; padding: 2px;">Use Hilscher generic DTMs if available</div> <div style="background-color: #ffffff; padding: 2px;">Use vendors DTMs if available</div> </div> <div style="margin-left: 10px;">(In the figure shown example DTMs are displayed.)</div> </div> <p>A selection will only be displayed if under Creation Mode > Choose for each device was selected and if another DTM has been found for the respective device.</p>
Device Class	Device class of theCANopen Slave devices.
DTM Device	<p>Found DTM device (the device name as taken from the DTM)</p> <p>Only the device description files or DTM devices can be displayed within the column DTM Devices:</p> <ul style="list-style-type: none"> • Which are available in the device catalog for the scanned ident code, • Respectively, which belong to the selection made under Creation Mode • and which belong to the selection made under Creation Mode > Choose for each device under DTM to create. <div style="display: flex;"> <div style="flex: 1;"> <p>For each device type ID in the column DTM Device the following is displayed:</p> <ul style="list-style-type: none"> • <u>no</u> device, • <u>one</u> single device • or <u>multiple</u> devices (within a combobox). </div> <div style="flex: 1;"> <p>This means, within the device catalog of netDevice for the found ident code and the selected Creation Mode these alternatives are available:</p> <ul style="list-style-type: none"> • no DTM • A device description file or a DTM device of a manufacturer • One or more device description files or DTM devices of the manufacturer </div> </div>
Quality	Associated quality information Display: [1] DTM found, [3] Generic found


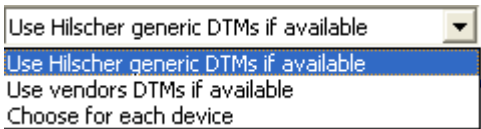
Column	Description
Action	<p>The action to be performed with the corresponding device during the process of device creation.</p> <ul style="list-style-type: none"> If no device is present within the current project, the selection Add/Skip will appear. If there is already a device present within the current project, the selection Replace/Skip will appear. <p>Add adds a new instance for the selected DTM during the process of creation of a device.</p> <p>Skip skips the process of creation of a device for the respective device address.</p> <p>Replace erases the instance of the DTM currently located at this address during the process of creation of a device, and replaces it with the instance of the chosen DTM.</p>
Table below	<p>The lower table in the Scan Response dialog of the Master DTM shows a comparison of possible differences in device information taken from:</p> <ul style="list-style-type: none"> The hardware device (displayed in central column of 3) and the DTM (displayed in right column of 3) <p>The left column contains which information is compared between the information sources 'Hardware Device' and 'DTM'.</p> <hr/> <p> Note: If a field contains the text 'n/a', the corresponding information is not applicable in the current context (fieldbus).</p>
Creation Mode	<p>Under Creation Mode one of the following options can be selected:</p> <ul style="list-style-type: none"> User Hilscher generic DTM if available Use vendors DTMs if available Choose for each device <div data-bbox="379 996 861 1124">  </div> <p><i>'Scan Response dialog of the Master DTM > 'Creation Mode'</i></p>
Create Devices	<p>About Create Devices ...</p> <ul style="list-style-type: none"> for each Slave device the previously selected DTM device is created. an upload of the Slave configuration can be proceeded.
Cancel	Click Cancel to leave the dialog without creating a Device.

Table 33: Description on the Scan Response dialog of the Master DTM

6.3.4 Creating Devices

3. Click **Create Devices**

- In the **Scan Response** dialog of the Master DTM click **Create Devices**.

➤ For each Slave device the previously selected DTM device is created.

➤ The dialog **Network Scan – Creating Devices** appears.



Note: Depending on the manufacturer of the respective device, also a dialog with some slight deviations from this one may be displayed.

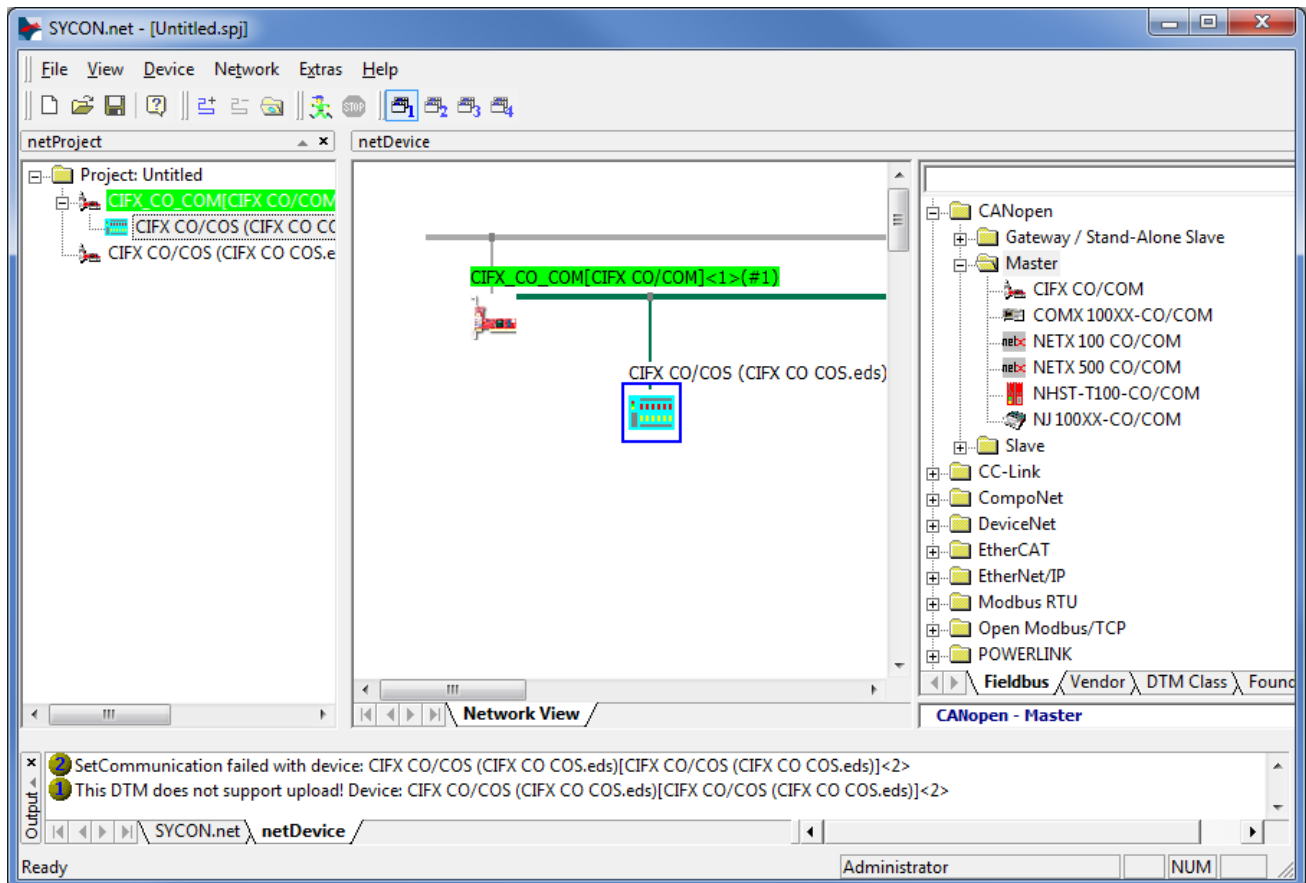


Figure 55: DTM device is created (Example)

6.3.5 Download to the CANopen Master Device



Adhere to the necessary safety precautions to prevent personnel injury and property damage. For more refer to section *Safety Messages on Firmware or Configuration Download* on page 27.

4. Via the **Download** function of the Master DTM, download the current configurations of the Slave devices to the Master device.

- In netDevice: right-click on the device symbol of the CANopen Master DTM.
- Select **Download** from the context menu.

6.4 Download Configuration

The device configuration is created *offline* in the DTM (application program). A download to the device is required, to transfer the configuration with the parameter data to the device.



Note: To download configuration parameter data to the CANopen Master device an online connection from the CANopen Master DTM to the CANopen Master device is required. Further information can be found in the *Connecting/Disconnecting Device* section on page 78.

Safety Precautions

If you plan to perform a configuration download via the CANopen Master DTM be aware of the following:

⚠ WARNING

Communication Stop caused by Configuration Download, faulty System Operation possible or Loss of Device Parameters

Before you initiate a configuration download process, while the bus is still in operation status:

- Stop the application program.
- Make sure that all network devices are placed in a fail-safe condition.

⚠ WARNING

Mismatching System Configuration, faulty System or Device Operation possible

- In the device use only a configuration suitable for the system.

NOTICE

Loss of Parameters caused by Power Disconnect during Configuration Download

- During configuration download process, do not interrupt the power supply to the PC, or to the device and do not perform a reset to the device!

For more see next page.

Download Steps

To transfer the configuration with the corresponding parameter data to the CANopen Master device you download the data using the frame application of the configuration software.

For netDevice the download is made via **Device > Download** or use the context menu with **Download**.

1. Select **Download** in the context menu of the device.

⇒ If the download is started as long as the Slave devices are connected to the Master device, the following message is displayed: 'If you attempt to download during bus operation, communication between Master and Slaves is stopped. Do you really want to download?'

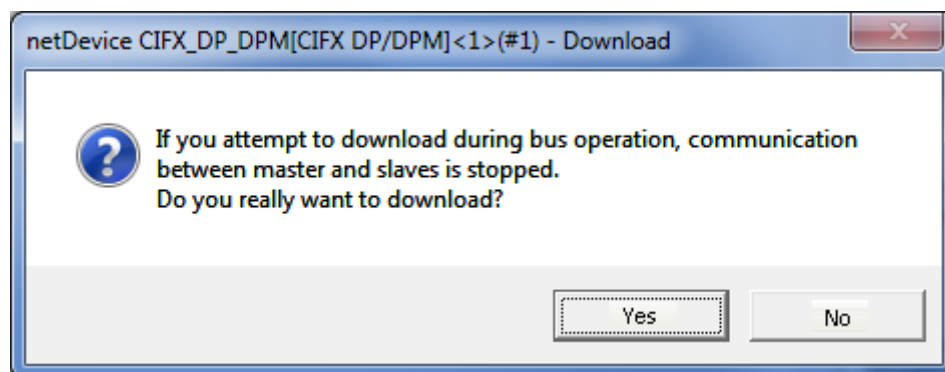


Figure 56: netDevice Message: Download



Important: If the communication between the Master and the Slave devices is stopped, the data exchange between the Master device and the Slave devices is stopped.

2. Click to **Yes** if you intend to download the configuration.
- ⇒ Then the current configuration in the application program is downloaded to the device.
3. Otherwise click to **No**.

6.5 Start /Stop Communication

You can manually start or stop the communication between a CANopen Master device and CANopen Slave devices.

- **Start Communication** can be enabled if the communication was stopped before or if the configuration requires this (Controlled release of communication).
- **Stop Communication** can be enabled if the communication was started.

➤ To start or to stop the communication, proceed as follows:

Start Communication

1. Connecting device:



Note: To start the communication of the device at the bus manually, an online connection from the CANopen Master DTM to the CANopen Master device is required. Further information can be found in the *Connecting/Disconnecting Device* section on page 78.

2. Select **Additional Functions > Service > Start Communication** from the context menu (right mouse click).

➤ The device communicates at the bus.

Stop Communication



Faulty System Operation possible, Overwriting of Firmware or Loss of Device Parameters

Before you stop the communication:

- Stop the PLC program.
- Make sure that all network devices are placed in a fail-safe condition.

1. Select **Additional Functions > Service > Stop Communication** from the context menu (right mouse click).

➤ The communication of the device at the bus is stopped.

6.6 Licensing

Using the license dialog, you can order licenses for **Master protocols** or **Utilities** and download them to your device.

6.6.1 Open License Dialog

You first open the **License** window.



Note: You first need to assign the Master device to the DTM. Only then the device data and the licenses already present in the device are displayed in the **License** dialog.

How to proceed:

A.) Assigning the Master device to the DTM

1. Open the DTM configuration dialog.
 - In the FDT container **netDevice** double click to the device icon.
2. Select one or several drivers.
 - Select **Settings > Driver**.
 - Check the drivers.
3. Configure the driver if necessary.
 - Select **Settings > Driver > [Name of the assigned driver]**.
 - Configure the driver settings.
4. Scan for and select the devices.
 - Select **Settings > Device Assignment**.
 - Under **Device selection** select *suitable only* or *all* and then **Scan**.
 - In the table check the required devices.
 - Select **Apply**.
5. Close the DTM configuration dialog via **OK**.



For details to the device assignment, refer to section **Overview Settings** on page 28.

B.) Open the License pane

- In the FDT container **netDevice** right click to the device icon.
- From the context menu select **Additional Functions > License**.
- The **License pane** opens.

6.6.2 License Dialog

In the **License**¹ pane you can:

- check, which licenses for Master protocols or Utilities are present in the device (Position ① in the figure below),
- order licenses (Positions ② to ⑪),
- transfer licenses to the device ⑫.

License Type

	Existing	Order
Master protocols		
One General Master License	NO	<input type="checkbox"/>
Two General Master Licenses	NO	<input type="checkbox"/>
PROFIBUS Master	YES	<input type="checkbox"/>
CANopen Master	YES	<input type="checkbox"/>
DeviceNet Master	YES	<input type="checkbox"/>
AS-Interface Master	YES	<input type="checkbox"/>
PROFINET I/O RT Controller	YES	<input type="checkbox"/>

Request Form, please fill out

Name	Value
License type	User Single Device License
Manufacturer*	0x0001
Article number*	1251100
Serial number*	20007
Chiptype*	0x00000001
Step*	0x00000000
Romcode revision*	0x00000000

Fields marked with "*" are mandatory.

E-mail...
Print Fax Form...
Telephone...
Export License Request...

Download License

Close Help

Figure 57: License Pane



Note: To display further entries under **License Type**, move the scroll box ① downwards or upwards. To display further entries under **Request Form, please fill out**, move the scroll box ② downwards or upwards.

¹ The title bar contains the notation of the **device description**:
Symbolic Name [Device Description] <Station Address> (#Network ID).

6.6.3 Which Licenses are present in the Device?

Check, which licenses are present in the device.

How to proceed:

- Open the **License** pane as described.

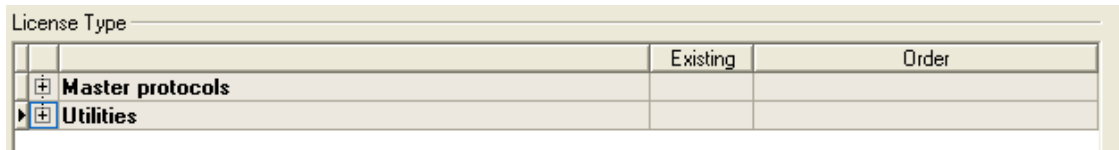


Figure 58: License Pane - License Type

- Under **License Type** click at **Master protocols**.

➤ The **Master protocols** overview opens:

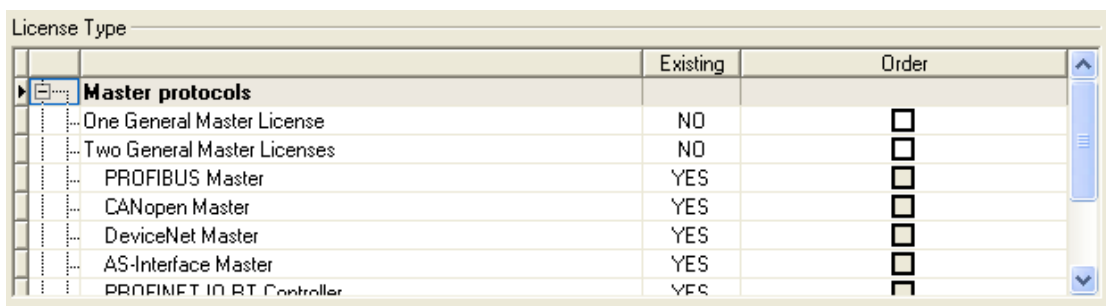


Figure 59: License Pane – License Type / Master protocols

- Or click at **Utilities**.

➤ The **Utilities** overview opens:

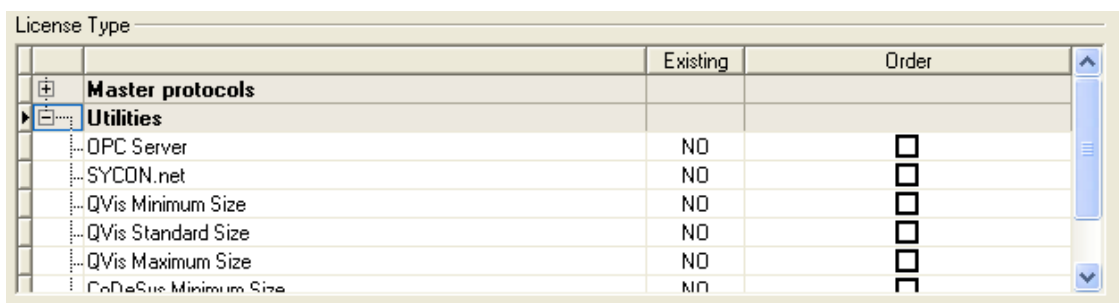


Figure 60: License Pane – License Type / Utilities

➤ The column **Existing** indicates which licenses are present in the device.

Yes = License is present in the device.

No = License is not present in the device.



Note: In newer versions of the present configuration software under **License Type** may be displayed additional licenses or other protocols that can be ordered later.

6.6.3.1 License for Master Protocols

One General Master License:

On the device maximally 1 communication protocol with master function can be implemented.

Two General Master Licenses:

On the device maximally 2 communication protocols with master function can be implemented.

The license includes the following Master protocols:

- AS-Interface Master
- CANopen Master
- DeviceNet Master
- EtherCat Master
- EtherNet/IP Scanner
- PROFIBUS Master
- PROFINET IO RT Controller
- Sercos Master

6.6.3.2 License for Utilities

- SYCON.net
- OPC Server
- QVis Minimum Size
- QVis Standard Size
- QVis Maximum Size
- CoDeSys Minimum Size
- CoDeSys Standard Size
- CoDeSys Maximum Size

For the utilities QVis and CoDeSys, only one license each may be chosen alternatively as:

- *Minimum Size,*
- *Standard Size or*
- *Maximum Size.*

6.6.4 How to order a License


To order a license, proceed as follows:

	<i>Refer to Section:</i>	<i>Page</i>
1. Open the license dialog.	<i>Open License Dialog</i>	95
2. Select the required licenses.	<i>Selecting License</i>	99
3. Enter the ordering data.	<i>Ordering Data</i>	100
4. Place your order.	<i>Ordering the License</i>	102


6.6.5 Selecting License(s)

You can select licenses for Master protocols and / or utilities.

1. Selecting license(s) for Master protocol(s):

- Under **License Type** click  at **Master protocols** in the **License** pane.
- Under **Order** check as many licenses must run simultaneously on your device:
One General Master License or
Two General Master Licenses.

2. And/or select license(s) for utility(utilities):

- In the **License** pane under **License Type** click  at **Utilities**.
- Under **Order** check the required utility(utilities) (*single or several*)²:
 - SYCON.net
 - OPC Server
 - QVis Minimum Size*
 - QVis Standard Size*
 - QVis Maximum Size*
 - CoDeSys Minimum Size**
 - CoDeSys Standard Size**
 - CoDeSys Maximum Size**

² For *) and **) minimum size, standard size or maximum size can be selected only as an alternative.

6.6.6 Ordering Data

1. Device Information

- The *Device Information* required for the order are read from the device and automatically filled in the order.

2. Ordering Data

Enter the *Ordering Data* into the **License** pane.

- Enter the **Data to manage the Order**.

6.6.6.1 Device Information (Ordering data read from the Device)

The following ordering data are read from the device and displayed in the **License** pane:

- Manufacturer
- Device number
- Serial number
- Chiptype
- Step (chip revision)
- Romcode revision
- Checksum (checksum of the device data)

- The gray fields under **Request Form, please fill out** contain the ordering data read from the device:

Request Form, please fill out

Name	Value
Manufacturer*	0x0001
Article number*	1251100
Serial number*	20007
Chiptype*	0x00000001
Step*	0x00000000
Romcode revision*	0x00000000
Checksum*	G

Fields marked with '*' are mandatory.

Figure 61: License Pane - Request Form, please fill out / Device Information

- These ordering data read out from the device are displayed automatically from the device.

6.6.6.2 Data to manage the Order (License Information)

For your order you must enter the following data to the **License** pane:

1. License Type (User Single Device License).

Name	Value
License type	User Single Device License

Figure 62: License Pane - Request Form, please fill out / License Type

- Select the license type under **Request Form, please fill out > Value**, (for future application, currently only *User Single Device License* can be selected).
- 2. Mandatory data to the order request (editable fields):
 - First Name
 - Surname
 - E Mail (address, to which the license download link shall be send.)
 - Telephone
 - Company
 - Address
 - Country
 - City, State, Zip

Name	Value
First name*	John
Surname*	Doe
E-Mail*	License@doe.com
Telephone*	0011223344-55
Fax	0011223344-100
Customer number	123456789
Company*	Doe Example LTD

Fields marked with '*' are mandatory.

Figure 63: License Pane - Request Form, please fill out / Mandatory data

- Enter all mandatory fields under **Request Form, please fill out > Value** (marked with*).
- 3. Additional order data, not mandatory (editable fields):
 - Fax
 - Customer Number
 - Order Number
 - Value added tax identification number
- Under **Request Form, please fill out > Value** enter all fields for the additional data, which are not mandatory.

6.6.7 Ordering the License

Place your order in the **License** pane. Therefore:



Figure 64: License Pane – Selecting the Subsidiary / Ordering / Contacts

1. Select the **Subsidiary** (4), to which the order shall be send.
2. Place the order:

	Refer to Section:	Page
• by E-Mail (5),	<i>Ordering the License <u>by E Mail</u></i>	103
• or by Fax (6) or by Telephone (7),	<i>Ordering the License <u>by Fax or by Telephone</u></i>	104
• or in a File (8).	<i><u>Exporting License Request to a File</u></i>	106
➤ The Contact Data of the selected subsidiary are displayed under Position (9), (10) and (11).		

6.6.7.1 Ordering the License by E Mail

You can place your order by e-mail.



Figure 65: License Pane – placing the order by E-mail

- Click **E-mail...** 5.
- The order E-mail **License request** opens:

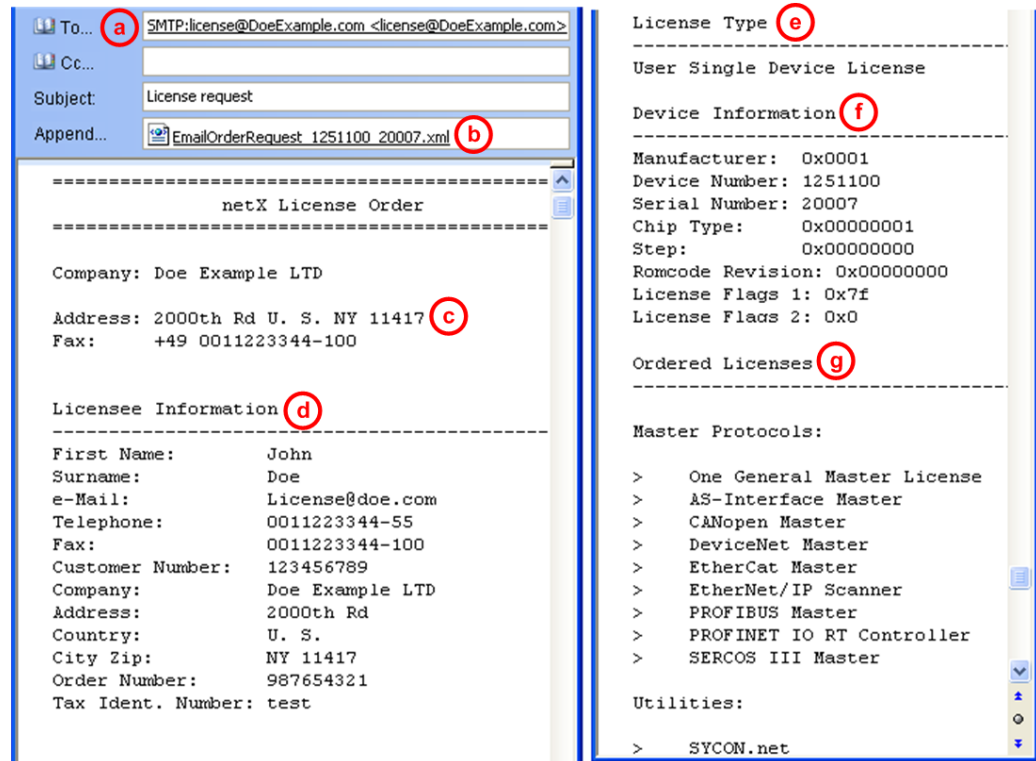


Figure 66: Example: Order E-Mail License request

- The order e-mail **License request** contains:
 - the **E-mail...** of the selected subsidiary a,
 - the automatically generated **XML file** b *EmailOrderRequest_[Devicenumbr][Serialnumber].xml* with a summary info of the **order information**,
 - the **Order Address** c,
 - the **License Information** d,
 - the **License Type** e,
 - the **Device Data** f,
 - the **ordered Licenses** g.
- Send the order e-mail **License request**.
- The order process is complete.

6.6.7.2 Ordering the License by Fax or by Telephone

You can place your order by Fax or by Telephone.



Figure 67: License Pane - placing the order by Fax or by Telephone

- Click **Print Fax Form** ⑥ or **Telephone...** ⑦.
- The summary of the ordering data *PrintOrderRequest_[Devicenum-ber]_[Serialnumber].html* is opened in a browser window.



Note: If your browser does not display the order data or the window **Move Element** or **Copy Element** are displayed, check the safety settings of your system.

netX License Order Form

Doe Example LTD
2000th Rd

NY 11417
U. S.
fax: +11223344-100

Licensee Information ④

<i>First Name:</i>	John
<i>Surname:</i>	Doe
<i>e-Mail:</i>	License@doe.com
<i>Telephone:</i>	0011223344-55
<i>Fax:</i>	0011223344-100
<i>Customer No:</i>	123456789
<i>Company:</i>	Doe Example LTD
<i>Address:</i>	2000th Rd
<i>Country:</i>	U. S.
<i>City Zip:</i>	NY 11417
<i>Order Number:</i>	987654321
<i>Tax Ident. Number:</i>	test

License Type ⑤

User Single Device License

Device Information ⑥

<i>Manufacturer:</i>	0x0001
<i>Device Number:</i>	1251100
<i>Serial Number:</i>	20007
<i>Chip Type:</i>	0x00000001
<i>Step:</i>	0x00000000
<i>Romcode Revision:</i>	0x00000000
<i>License Flags 1:</i>	0x7f
<i>License Flags 2:</i>	0x0

Ordered Licenses ⑧

Master Protocols

- One General Master License
- Sercos III Master

Utilities

- SYCON.net

Date: _____

Signature: _____

Figure 68: Example: Order Data Form *PrintOrderRequest*

- The order data form contains:
- the **Order Address** ^c,
- the **License Information** ^d,
- the **License Type** ^e,
- the **Device Data** ^f,
- the **ordered Licenses** ^g.
- Print the order data form, sign it and send it by Fax.

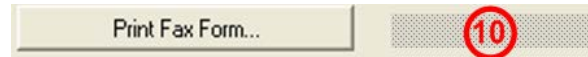


Figure 69: License Pane – Fax Number of the selected Subsidiary

- Use the Fax number ¹⁰, which is displayed after the subsidiary was selected in the **License** pane.

Or:

- Keep ready the order data form and communicate the order data via telephone.



Figure 70: License Pane – Telephone Number of the selected Subsidiary

- Use the telephone number ¹¹, which is displayed after the subsidiary was selected in the **License** pane.
- The order process is complete.

6.6.7.3 Exporting License Request to a File

If you are working on a process computer without an e-mail client, you can export your order information to a file, save the file to a removable disk and place your order manually via e-mail from a different PC.



Figure 71: License Pane - Ordering by exported File and E-Mail

- Click **Export License Request...** 8.
- The window **Browse For Folder** is displayed.
- Choose for or create a new folder on a removable disk.
- Save the automatically generated **XML file** *EmailOrderRequest_- [Devicenumbrer]_[Serialnumber].xml* with a summary info of the **order information** to this folder.
- Send this file from a PC with an e-mail client manually via e-mail.
- Therefore, use an e-mail address, which is displayed after the subsidiary was selected in the **License** pane (see Position 9 Figure *License Pane*).
- The order process is complete.

6.6.8 How to get the License and transfer it to the Device



Note: License files can only be delivered via e-mail. The e-mail contains a link to download the license file.

According to the license you ordered, you will receive an e-mail containing a **Link to download the License File**. This leads to a server PC on which the license file is provided. Using the received link you will have to save the license file on your PC and then transfer the license to your device. If your e-mail client is on another PC as your device, you must save your license file e. g. to an USB stick.

Steps how to proceed


1. Save the license file to a PC or a disk.
 - Click to the **Link to download the License File** in the e-mail.
 - Save the license file *.nxl to a PC or a removable disk.
2. Download the license file to the device.
 - Respectively connect the removable disk with the license file to the PC, which is connected to your device.
 - Click **Download License**  in the **License** pane in the configuration software.



Figure 72: License Pane - Download License

- The File selection window **Open** is displayed.
 - Therein select the license file *netX License Files (*.nxl)*.
 - Click **Open**.
 - The license file is transferred to the device.
 - After this the license is present in the device and is activated with the next device reset.
3. Activate Device Reset



Hint: To activate the license in the first device, a device reset is required.

- To check whether the license has been activated, follow the steps in section *Which Licenses are present in the Device?*

7 Diagnosis

7.1 Overview Diagnosis

The dialog **Diagnosis** serves to diagnose the device behavior and communication errors. For diagnosis the device must reside in online state.

Diagnosis Panes

The table below gives an overview for the individual **Diagnosis** dialog panes descriptions:

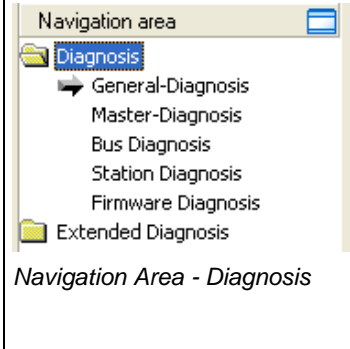
CANopen Master DTM	Folder Name / Section	Manual Page
	<i>General Diagnosis</i>	109
	<i>Master Diagnosis</i>	111
	<i>Bus Diagnosis</i>	112
	<i>Station Diagnosis</i>	113
	<i>Firmware Diagnosis</i>	114
<i>Navigation Area - Diagnosis</i>		

Table 34: Descriptions of the Diagnosis Panes

Online Connection to the Device



Note: Accessing the **Diagnosis** panes of the CANopen Master DTM requires an online connection from the CANopen Master DTM to the CANopen Master device. For further information refer to section *Connecting/Disconnecting Device* on page 78.



Important: For a 2-channel device channel 1 or channel 2 each must be connected to the DTM separately.

How to proceed

1. In the Master DTM diagnosis dialog check whether the communication is OK:

Diagnosis > General Diagnosis > Device status "Communication" must be green!

2. **"Communication"** is green: Open the **IO Monitor** and test the input or output data..
3. **"Communication"** is not green: Use **Diagnosis** and **Extended diagnosis** for troubleshooting.

Extended Diagnosis

The **Extended Diagnosis** helps to find communication and configuration errors, when default diagnosis fails. For further information refer to section *Overview Extended Diagnosis* on page 115.

7.2 General Diagnosis

Information regarding the Device State and other general diagnosis parameters are displayed in the **General Diagnosis** dialog.

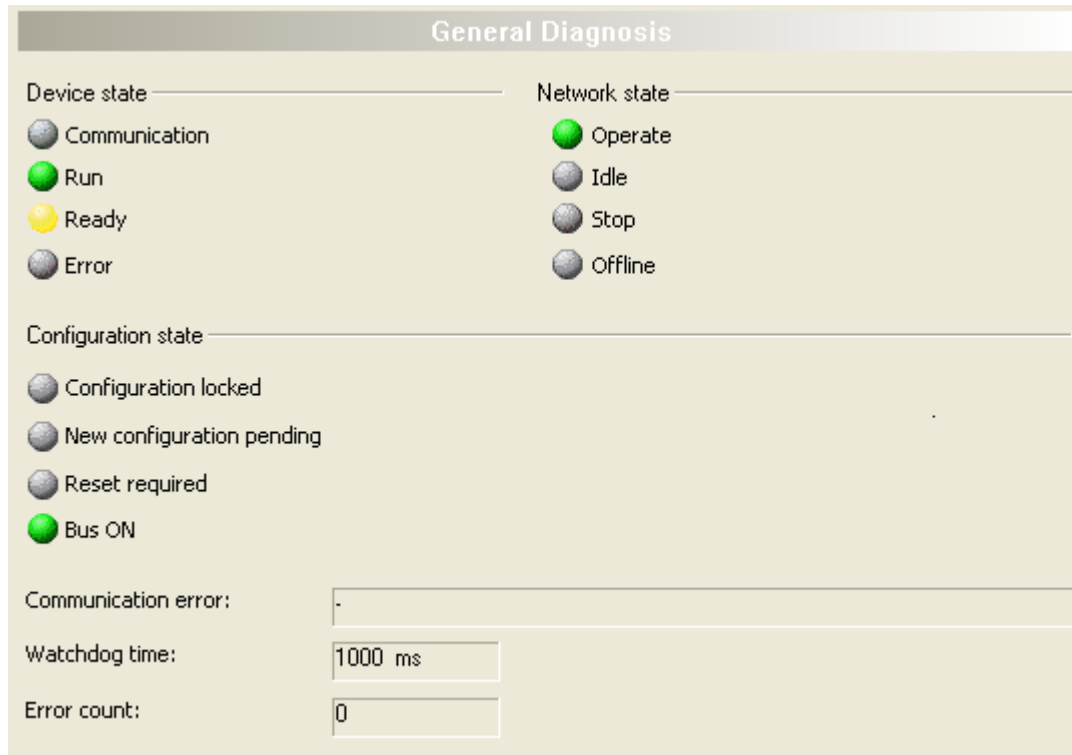














Figure 73: General Diagnosis

LED	Meaning	Color	State
Device State			
Communication	Shows whether the CANopen device executes the network communication.	 (green)	In COMMUNICATION state
		 (gray)	Not in COMMUNICATION state
Run	Shows whether the CANopen device has been configured correctly.	 (green)	Configuration OK
		 (gray)	Configuration not OK
Ready	Shows whether the CANopen device has been started correctly. The CANopen device waits for a configuration.	 (yellow)	Device READY
		 (gray)	Device not READY
Error	Shows whether the CANopen device records a device status error. For further information about the error characteristics and the number of counted errors refer to the extended diagnosis.	 (red)	ERROR
		 (gray)	No ERROR
Network State			
Operate	Shows whether the CANopen device is in data exchange. In a cyclic data exchange the input data or the output data of the CANopen Master are transmitted to the CANopen Slave.	 (green)	In OPERATION state
		 (gray)	Not in OPERATION state
Idle	Shows whether the CANopen device is in idle state.	 (yellow)	In IDLE state
		 (gray)	Not in IDLE state













LED	Meaning	Color	State
Stop	Shows whether the CANopen device is in Stop state: There is no cyclic data exchange at the CANopen network. The CANopen device was stopped by the application program or it changed to the Stop state because of a bus error.	 (red)	In STOP state
		 (gray)	Not in STOP state
Offline	The CANopen Master is offline as long as it does not have a valid configuration.	 (yellow)	In OFFLINE state
		 (gray)	Not in OFFLINE state
Configuration State			
Configuration locked	Shows whether the CANopen device configuration is locked, to avoid the configuration data are typed over.	 (yellow)	Configuration LOCKED
		 (gray)	Configuration not LOCKED
New Configuration pending	Shows whether a new CANopen device configuration is available.	 (yellow)	New Configuration pending
		 (gray)	No new Configuration pending
Reset required	Shows whether a firmware reset is required as a new CANopen device configuration has been loaded into the device.	 (yellow)	RESET required
		 (gray)	No RESET required
Bus ON	Shows whether the bus communication was started or stopped. I. e., whether the device is active on the bus or no bus communication to the device is possible and no response telegrams are sent.	 (green)	Bus ON
		 (gray)	Bus OFF

Table 35: Indication General Diagnosis

Parameter	Meaning
Communication Error	Shows the name of the communication error. If the cause of error is resolved, the value will be set to zero again.
Watchdog time	Shows the watchdog time in ms.
Error Count	This field holds the total number of errors detected since power-up, respectively after reset. The protocol stack counts all sorts of errors in this field no matter whether they were network related or caused internally.

Table 36: Parameter General Diagnosis

7.3 Master Diagnosis

Information regarding the Slave State, slave errors and slaves configured, active or in diagnostic are displayed in the **Master Diagnosis** dialog.

Master Diagnosis	
Slave state	failed
Slave error log indicator	available
Configured slaves	2
Active slaves	0
Slaves with diagnostic	2

Figure 74: Master Diagnosis

Parameter	Meaning	Range of Value / Default Value
Slave state	Shows whether slave state is ok or not. The Slave state field for Master diagnosis shows whether the Master is in cyclic data exchange to all configured slaves. In case there is at least one slave missing or if the slave has a diagnostic request pending, the status will be set to FAILED. For protocols that support non-cyclic communication only, the slave state is set to OK as soon as a valid configuration is found.	UNDEFINED, OK, FAILED
Slave error log indicator	Shows whether the Slave Error Log Indicator is available. The error log indicator field holds the number of entries in the internal error log. If all entries are read from the log, the field will be set to zero.	EMPTY, AVAILABLE
Configured slaves	Shows number of configured slaves. Number of configured slaves in the network according to the slave list derived from the configuration database created by the configuration software . The list includes the slaves to which the Master has to open a connection.	
Active slaves	Shows number of active slaves. Number of slaves in data exchange mode. The list includes the slaves to which the Master has successfully opened a connection.	
Slaves with diagnostic	Shows number of slaves with diagnostic. Number of Slaves with diagnosis or error slaves.	

Table 37: Parameter Master Diagnosis

7.4 Bus Diagnosis

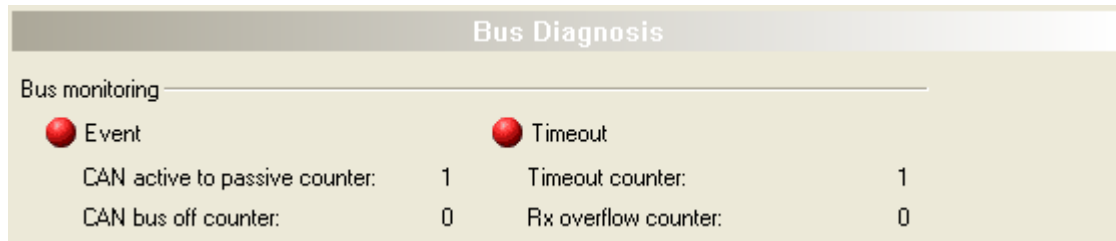


Figure 75: Bus Diagnosis - Bus Monitoring





LED	Meaning	Color	State
Bus Diagnosis - Bus Monitoring			
Event	The used CAN chip has detected transmission errors. The number of detected events is counted in the CAN bus off counter and the CAN active to passive counter . The bit will be set when the first event was detected and will not be deleted until the Master is reset.	 (red)	EVENT (transmission errors detected)
		 (gray)	no EVENT (no transmission errors detected)
CAN active to passive counter	This value shows a bad transmission quality at the CAN bus. This counter is based on a Transmission Rejected Error of the CAN chip, which is always reported, whenever an increased number of faulty CAN frames were detected.	-	-
CAN bus off counter	This counter counts the number of reported bus off events. A bus off event occurs whenever the CAN bus was disturbed substantially during the transmission of CAN frames and therefore the communication chip has to withdraw itself from the bus.	-	-
Timeout	A timeout was detected. The number of errors is counted in the Timeout counter and in the RX overflow counter . The bit will be set if the first error occurs and can be reset only by a reset of the Master.	 (red)	TIMEOUT (timeout errors detected)
		 (gray)	no TIMEOUT (no timeout errors detected)
Timeout counter	Whenever a CAN message could not be sent within 250 msec, the transmission is aborted and the counter increases.	-	-
Rx overflow counter	An Rx overflow will occur if the Master cannot handle a CAN message before the next message is received because the bus demand is too high.	-	-

Table 38: Bus Diagnosis - Bus Monitoring

7.5 Station Diagnosis

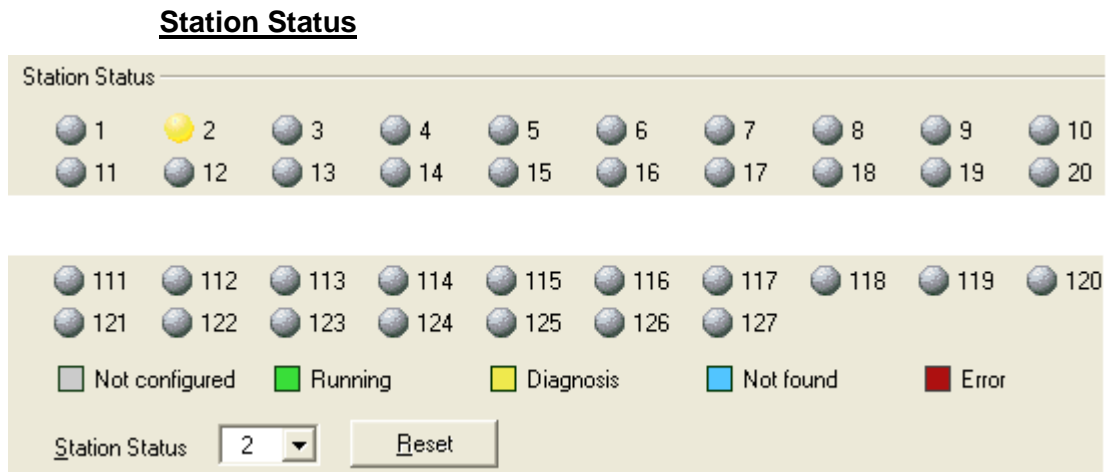


Figure 76: Station Diagnosis - Station Status

Under **Station Status** all disposal station addresses (0-127) and their states are shown as LED. If the device is connected, the DTM will update this display cyclically.

The legend below describes the possible values for the state of a device, which is assigned to a station address.

Color	Name	Meaning
gray	Not configured	This station address is not configured with a device.
green	Running	The device associated with this station address is running.
yellow	Diagnosis	Diagnosis is available for the device associated with this station address.
blue	Not found	The device associated with this station address was parameterized, but not found.
red	Error	An error message is available for the device associated with this station address.

Table 39: Possible Values for the Status

Reset Station Status for Status Diagnosis (yellow):

To check whether the station status Diagnosis (yellow) persists, the **Station status** can be reset for every device separately. Therefore:

1. Select in the list field **Stations Status** the station address of the device.
2. Select **Reset**.

7.6 Firmware Diagnosis

In the dialog **Firmware Diagnosis** the current task information of the firmware is displayed.

Under **Firmware** or **Version** the name of the firmware and version (including the date) are indicated.

Firmware Diagnosis					
Firmware:	Firmware-Name*				
Version:	2.4.0 (Build 1)				
Date:	29.1.2010				
Task information:					
Task	Name of task	Version	Prior...	Description	State
0	RX_IDLE	1.0	63	RX IDLE Task.	Der Task Status ist OK. (0x00000000)
1	RX_TIMER	0.0	1	Der Task-Identifizier ...	Der Task Status ist OK. (0x00000000)
2	RX_SYSTEM	1.16	8	Middleware System...	Der Task Status ist OK. (0x00000000)
3	DPM_COMO_...	1.0	50	TLR-Router DPM.	Der Task Status ist OK. (0x00000000)
4	DPM_COMO_...	1.0	51	TLR-Router DPM.	Der Task Status ist OK. (0x00000000)
5	TLR_TIMER	0.0	30	Der Task-Identifizier ...	Der Task Status ist OK. (0x00000000)
6	CAN_DL	1.2	28	CAN DL Task (Data ...	Der Task Status ist OK. (0x00000000)
7	CANOPEN_M...	1.0	29	CANopen Master T...	Die Task kommuniziert momentan nicht. (0x...
8	CANOPEN_APM	1.0	31	CANopen Master A...	Der Task Status ist OK. (0x00000000)

Figure 77: Firmware Diagnosis (*The name of the Firmware is displayed.)

Task Information:

The table **Task Information** is listing the task information of the single firmware tasks.

Column	Meaning
Task	Task number
Name of task	Name of the task
Version	Version number of the task
Prio	Priority of the task
Description	Description of the task
Status	Current status of the task

Table 40: Description Table Task Information

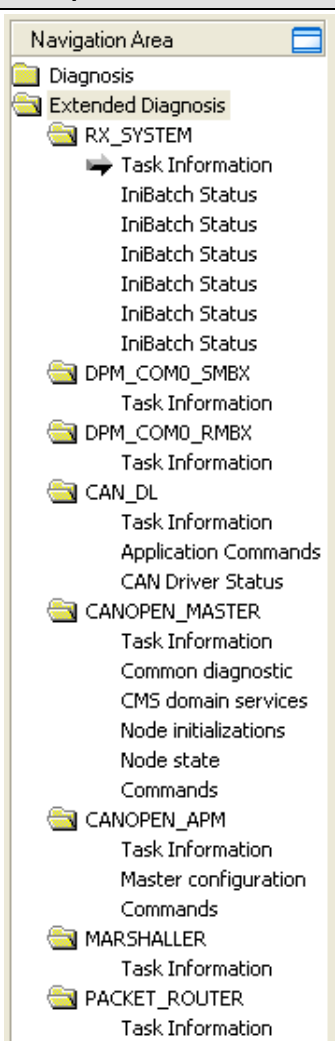
8 Extended Diagnosis

8.1 Overview Extended Diagnosis

The **Extended Diagnosis** of the CANopen Master DTM helps to find communication and configuration errors. Therefore it contains a list of diagnosis structures as online counter, states and parameters.

Dialog Panes “Extended Diagnosis”

The table below gives an overview for the **Extended Diagnosis** dialog panes descriptions:

CANopen Master DTM	Folder Name / Section	Subsection	Manual Page
	<i>RX-SYSTEM</i>	<i>Task Information</i>	116
		<i>IniBatch Status</i>	117
	<i>DPM_COMO_SMBX</i>	<i>Task Information</i>	116
	<i>DPM_COMO_RMBX</i>	<i>Task Information</i>	116
	<i>CAN_DL</i>	<i>Task Information</i>	116
		<i>Application Commands</i>	118
		<i>CAN Driver Status</i>	119
	<i>CANOPEN_MASTER</i>	<i>Task Information</i>	116
		<i>Common diagnostic</i>	120
		<i>CMS domain services</i>	121
		<i>Node Initializations</i>	122
		<i>Node state</i>	122
		<i>Commands</i>	123
	<i>CANOPEN_APM</i>	<i>Task Information</i>	116
		<i>Master Configuration</i>	125
		<i>Commands</i>	126
	<i>MARSHALLER</i>	<i>Task Information</i>	116
	<i>PACKET_ROUTER</i>	<i>Task Information</i>	116

Navigation Area - Extended Diagnosis

Table 41: Descriptions of the Dialog Panes Extended Diagnosis

Online Connection to the Device



Note: Accessing the **Extended Diagnosis** dialog panes of the CANopen Master DTM requires an online connection from the CANopen Master DTM to the CANopen Master device. For further information refer to section *Connecting/Disconnecting Device* on page 78.

8.2 Task Information

Task Information	
Task states	
Name	Value
Identifier	
Major version	<i>[The displayed values depend from the corresponding task]</i>
Minor version	
Maximum Packet size	
Default Que	
Unique identifier	
Init result	

Figure 78: Extended Diagnosis > [Folder Name] > Task Information Example Display

Name	Description
Identifier	Identification number of the task
Major version	Task version, contains incompatible changes
Minor version	Task version, contains compatible changes
Maximum packet size	Maximum packet size, which the task sends
Default Queue	Queue handle, which is accessible via DPM by mailbox.
UUID	Unique user ID, 16 Byte indicator used for task identification and its affiliation e. g. to a stack (therein different identification data are coded in).
Init result	Error Code, 0= no Error The description of the error codes can be found in this manual or in the corresponding software reference manuals.

Table 42: Extended Diagnosis > [Folder Name] > Task Information

8.3 IniBatch Status

IniBatch-Status	
Task states	
Name	Value
Communication Channel	0
Current State	Error
IniBatch Result	No DBM file
OpenDbm Result	24975
SendPacket Result	0
Confirmation Result	0
Last Packet Number	0
Last Packet Command	0
Last Packet Length	0
Last Packet Destination	0

Figure 79: Extended Diagnosis > [Folder Name] > IniBatch Status Example Display

Name	Description
Communication Channel	Number of the communication channel used by the device.
Current State	Idle; IniBatch packets in progress; Retrying to send last packet; Error
IniBatch Result	Ok; No DBM file; No Packet table; No data set available; Data set is shorter than packet length; Packet Buffer is shorter than Packet length; Invalid packet destination; Logical queue not defined Send packet failed; Too many retries; Error in confirmation packet status
OpenDbm Result	Error when opening the IniBatch data base Under "OpenDbm Result" the error code is typed in, when "IniBatch Result" == "No DBM file" (1) is.
SendPacket Result	Error when sending a packet Under "SendPacket Result" the error code is typed in, when "IniBatch Result" == "send packet failed" (8) is.
Confirmation Result	Confirmation error when sending packets Under "Confirmation Result" the packet specific error code from the ulSta is typed in, when "IniBatch Result" == "Error in confirmation packet status" (10) is.
Last Packet Number	Value depends by the communication system.
Last Packet Command	Value depends by the communication system.
Last Packet Length	Value depends by the communication system.
Last Packet Destination	Value depends by the communication system.

Table 43: Extended Diagnosis > [Folder Name] > IniBatch Status

The task status "Confirmation Result" is bus specific. The other task statuses are rcx-related error codes.

8.4 CAN_DL

8.4.1 Application Commands

Application Commands	
Task_states	
Name	Value
Data Request	0
Positive Confirmations	0
Negative Confirmations	0
Can DL Indications	0
Can DL Responses	0
Can DL Start Request	0
Positive Start Confirmations	0
Negative Start Confirmations	0
Stop Requests	0
Positive Stop Confirmations	0
Negative Stop Confirmations	0
Application Register Requests	0
Positive Application Register ...	0
Negative Application Register...	0
Set Parameter Requests	0
Positive Set Parameter Confi...	0
Negative Set Parameter Con...	0
Set Filter Requests	0
Positive Set Filter Confirmati...	0
Negative Set Filter Confirmat...	0
Enable Receive Id Requests	0
Positive Enable Receive Id C...	0
Negative Enable Receive Id ...	0
Event Indications	0
Event Responses	0
Event Acknowledge Request	0
Positive Event Confirmations	0
Negative Event Confirmations	0

Figure 80: Extended Diagnosis >CAN_DL > Application Commands

Name	Description
[Service]	Diagnosis counter of the CAN layer. Indicates the services processed. (The services of the single packets are described in the API manual.)

Table 44: Extended Diagnosis > CAN_DL > Application Commands

8.4.2 CAN Driver Status

CAN Driver Status	
Task states	
Name	Value
Can Status	0x00000000
Bus Off	false
Error Warning	false
Error Passive	false
Transmit Frame Succeeded	0
Transmit Error Summary	0
Receive Frame Succeeded	0
Receive Error Summary	0
Transmit Error Counter	0
Receive Error Counter	0
Arbitration Lost	0
Indications Dropped due to Fi...	0
Confirmations Dropped due t...	0
Receive Standardframes filt...	0
Receive extended frames filt...	0
Receive Standardframes pas...	0
Receive extended frames pa...	0

Figure 81: Extended Diagnosis > CAN_DL > CAN Driver Status

Name	Description	
CAN Status	true / false CAN status – collection bits for Bus Off, Error Warning and Error Passive	Diagnosis status of CAN specific error levels. Indicates the respective status of the CAN bus. (For further information refer to the API manual.)
Bus Off	The CAN is in Bus Off state	
Error Warning	true / false Error Warning - The CAN is in the status Error Warning	
Error Passive	true / false Error Warning - The CAN is in Error Passive	
[Service]	Diagnosis counter of CAN errors. Indicates the services processed. (The services of the single packets are described in the API manual.)	

Table 45: Extended Diagnosis > CAN_DL > CAN Driver Status

8.5 CANOPEN_MASTER

8.5.1 Common diagnostic

Common diagnostic	
Task states	
Name	Value
Last received CAN-ID	0
CAN Messages sent	10552
CAN Messages received	0
Number of detected CAN errors	1
SYNC timer reload [ms]	100
Baudrate	1MBaud
Valid bus parameter configured	Yes
Number of valid node parameter	1
Number of invalid node parameter	0

Figure 82: Extended Diagnosis > CANOPEN_MASTER > Common diagnostic

Name	Description	
Last received CAN-ID	Last received CAN-Message Header-ID	
CAN telegrams sent	Number of sent CAN-Messages	
CAN telegrams received	Number of received CAN-Messages	
Number of detected CAN errors	Number of detected CAN errors	
SYNC timer reload [ms]	This value shows the actual configured and handled value that was configured as Sync Object Cycle Period in the Bus Parameters.	
Bauderate	Baud rate of CANopen connection	
	Available Baud Rate:	
	1 MBaud 800 KBaud 500 KBaud 250 KBaud 125 KBaud	100 KBaud 50 KBaud 20 KBaud 10 KBaud
Valid bus parameter configured	Yes: Valid bus parameter are configured No: There are no valid bus parameter configured	
Number of valid node parameter	Number of valid node parameter configured	
Number of invalid node parameter	Number of invalid node parameter configured	

Table 46: Extended Diagnosis > CANOPEN_MASTER > Common diagnostic

8.5.2 CMS domain services

CMS domain services	
Task states	
Name	Value
Start node	2
Stop node	0
Reset node	0
Reset communication	1
Set preoperational	11642
SDO-Upload request	11643
SDO-Upload confirmation	1
SDO-Download request	184
SDO-Download confirmation	184

Figure 83: Extended Diagnosis > CANOPEN_MASTER > CMS domain services

Name	Description
[Service]	Domain Services diagnosis counter. Indicates the services processed. (The services of the single packets are described in the API manual.)

Table 47: Extended Diagnosis > CANOPEN_MASTER > CMS domain services

The CANopen protocol defines different services which are summarized under the name **Domain Services**. All Domain Services that are transmitted and were received are counted in this table and shown online by the DTM.

8.5.3 Node Initializations

Node initializations	
Task states	
Name	Value
Node	0:0 1:0 2:0 3:0 4:0 5:0 6:0 7:0 8:0 9:0 10:0 11:0 12:0 13:0 14:0 15:0 1...

Figure 84: Extended Diagnosis > CANOPEN_MASTER > Node Initializations

Name	Description
Node	Left value: numbering of every node initialization data byte Right value: value of the respective node initialization data byte

Table 48: Extended Diagnosis > CANOPEN_MASTER > Node Initializations



Note: To view the **Value** column completely, double click to the right border of the column head. Move to the left or the right column side using the scroll bar.

8.5.4 Node state

Node state	
Task states	
Name	Value
Node	0:0 1:0 2:24 3:0 4:0 5:0 6:0 7:0 8:0 9:0 10:0 11:0 12:0 13:0 14:0 15:0 16...

Figure 85: Extended Diagnosis > CANOPEN_MASTER > Node state

Name	Description
Node	Left value: numbering of every node state data byte Right value: value of the respective node state data byte

Table 49: Extended Diagnosis > CANOPEN_MASTER > Node state



Note: To view the **Value** column completely, double click to the right border of the column head. Move to the left or the right column side using the scroll bar.

8.5.5 Commands

Commands	
Task states	
Name	Value
Register req.	1
Register cnf.	1
Start/Stop req.	1
Start/Stop cnf.	1
Nodeparam req.	4
Nodeparam cnf.	4
Busparam req.	1
Busparam cnf.	1
Get buffer req.	1
Get buffer cnf.	1
State change ind.	50498
State change res.	50498
Set watchdog fail req.	0
Set watchdog fail cnf.	0
Data exch. req.	0
Data exch. cnf.	0
Node diag req.	0
Node diag cnf.	0
SDO upload req.	0
SDO upload cnf.	0
SDO download req.	0
SDO download cnf.	0
Send emergency req.	0
Send emergency cnf.	0
NMT command req.	0
NMT command cnf.	0
CAN_DL stop req.	1
CAN_DL stop cnf. pos.	1
CAN_DL stop cnf. neg.	0
CAN_DL register req.	1
CAN_DL register cnf. pos.	1
CAN_DL register cnf. neg.	0
CAN_DL set param req.	1
CAN_DL set param cnf. pos.	1
CAN_DL set param cnf. neg.	0
CAN_DL start req.	1
CAN_DL start cnf. pos.	1
CAN_DL start cnf. neg.	0
CAN_DL event ind.	10
CAN_DL event res.	10
CAN_DL send data req.	64201
CAN_DL send data cnf. pos.	64201
CAN_DL send data cnf. neg.	0
CAN_DL enable id req.	6
CAN_DL enable id cnf. pos.	6
CAN_DL enable id cnf. neg.	0
CAN_DL event ack. req.	0
CAN_DL event ack. cnf. pos.	0
CAN_DL event ack. cnf. neg.	0
CAN_DL recv data ind.	6007
CAN_DL recv data res.	6007
Unknown req./cnf.	0
Cyclic ind.	10475104
Get packet failed	104742
Send packet failed	0

Figure 86: Extended Diagnosis > CANOPEN_MASTER > Commands

Name	Description
[Service]	Diagnosis counter of the encapsulation layer. Indicates the services processed. (The services of the single packets are described in the API manual.)
Get packet failed	Number of errors during requesting a packet
Send packet failed	Number of errors during sending a packet

Table 50: Extended Diagnosis > CANOPEN_MASTER > Commands

8.6 CANOPEN_APM

8.6.1 Master Configuration

Master configuration	
Task states	
Name	Value
Flags	1
Database found	yes
Warmstart configuration	no
Initialization state	Complete
Initialization result	0x00000000

Figure 87: Extended Diagnosis > CANOPEN_APM > Master configuration

Name	Description
Flags	Bit0 set: Configuration data base found not set: No configuration data base found Bit1 set: Configuration by packets not set: No packets for configuration
Database found	Yes: Configuration data base found No: No configuration data base found
Warmstart configuration	Yes: Configuration by packets No: No packets for configuration
Initialization state	0 = Idle; 1 = Send initialize request; 2 = Wait for initialize confirmation; 3 = Send register request; 4 = Wait for register confirmation; 5 = Send get buffer request; 6 = Wait for get buffer confirmation; 7 = Send bus parameter request; 8 = Wait for bus parameter confirmation; 9 = Complete; 10 = Failed
Initialization result	Initialization error code, 0 = no error

Table 51: Extended Diagnosis > CANOPEN_APM > Master configuration

8.6.2 Commands

Commands	
Task states	
Name	Value
Register req.	1
Register cnf.	1
Start/Stop req.	1
Start/Stop cnf.	1
Init req.	1
Init cnf.	1
Nodeparam req.	4
Nodeparam cnf.	4
Busparam req.	1
Busparam cnf.	1
Get buffer req.	1
Get buffer cnf.	1
State change ind.	50498
State change res.	50498
Set watchdog fail req.	0
Set watchdog fail cnf.	0
Config pck. routed	0
Command pck. routed	45
Unknown req./cnf.	0
Cyclic ind.	10621443
Get packet failed	0
Send packet failed	0

Figure 88: Extended Diagnosis > CANOPEN_APM > Commands

Name	Description
[Service]	Diagnosis counter of the encapsulation layer. Indicates the services processed. (The services of the single packets are described in the API manual.)
Get packet failed	Number of errors during requesting a packet
Send packet failed	Number of errors during sending a packet

Table 52: Extended Diagnosis > CANOPEN_APM > Commands

9 Tools

9.1 Overview Tools

Under **Tools** the Packet Monitor and the IO Monitor are provided for test and diagnosis purposes.

Tools Panes

The table below gives an overview for the individual **Tools** dialog panes descriptions:

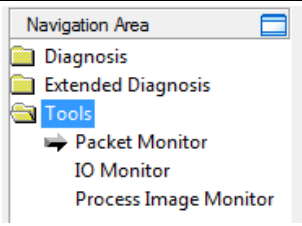
CANopen Master DTM	Folder Name / Section	Manual Page
 <p>Navigation Area - Tools</p>	Packet Monitor	128
	IO Monitor	145
	Process Image Monitor	146

Table 53: Descriptions of the Diagnosis Panes

Online Connection to the Device



Note: Accessing the **Tools** dialog panes of the CANopen Master DTM requires an online connection from the CANopen Master DTM to the CANopen Master device. For further information refer to section *Connecting/Disconnecting Device* on page 78.

9.2 Packet Monitor

The **Packet Monitor** serves for test and diagnosis purposes.

Data packets, i. e. messages are self-contained blocks of defined data length. The packets are used to communicate with the firmware and they are exchanged between the application (configuration software) and the firmware in the device. Packets can be sent once or cyclically to the connected device controlled by the user and packets received can be displayed.

Data packets comprise from a **Packet Header** and the **Send Data** or from a **Packet Header** and the **Receive Data**. The packet header can be evaluated by the receiver of the packet and contain the sender and receiver address, the data length, an ID number, status and error messages and the command or response code. The minimum packet size amounts 40 Byte for the packet header. The sending and receiving data are added.



For further information to the packet description refer to the *Protocol API Manual*.

- Open the **Packet Monitor** via **Tools > Packet Monitor**.

Figure 89: Packet Monitor

Display Mode switches the representation of the send and reception data between decimal and hexadecimal.

- Select **Reset Counter** to reset the packet counter.

9.2.1 Sending Packet

The screenshot shows a 'Send' dialog box with two main sections: 'Packet header' and 'Send data'.

Packet header:

- Dest: 00000001 (dropdown)
- Src: 00000000
- Dest ID: 00000000
- Src ID: 00000000
- Len: 00000012
- ID: 00000001
- State: 00000000
- Cmd: 00002F00
- Ext: 00000000
- Rout: 00000000
- Auto Increment ID: ☒

Send data: Counter: 0

A table for sending data with columns 0-9 and rows 0-60. The first row (0) has a value of 0 in column 0. The rest of the table is empty.

Buttons: Put cyclic, Put packet

Figure 90: Send > Packet Header and Send Data

Packet Header

Under **Send > Packet Header** the elements of the packet header of the sending packet are displayed, which is transmitted from the application (configuration software) to the device. The packet header of the sending packets contain the elements described in the following table.

Element		Description
Dest	Destination Queue Handle	Contains the identifier of the receiver for the packet (destination task queue of the firmware).
Src	Source Queue Handle	Contains the identifier of the sender of the packet (sending task).
Dest ID	Destination Queue Reference	Contains an identifier for the receiver of unsolicited sent packets from the firmware to the application (configuration software).
Src ID	Source Queue Reference	Contains an identifier of the sender.
Len	Packet Data Length (in Bytes)	Length of the send respectively receive data.
ID	Packet Identification As Unique Number	Identifies identical data packets among each other.
State	Status / Error Code	Transmits status or error codes to the packet sender.
Cmd	Command / Response Code	Command or respond code.
Ext	Extension	Field for extensions (reserved).
Rout	Routing Information	Internal value of the firmware.

Table 54: Descriptions Packet Header

- Under **Dest** select the receiver (*destination task queue*).
- Under **Cmd** select the command identification (*Request*).

Auto Increment ID is an increment for the identifier of the data packets and increments the ID by 1 for each newly sent packet.

Send Data

- Under **Send > Send data** enter the send data of the packet, which shall be transmitted from the application (configuration software) to the mailbox of the device. The meaning of the transmitted data depends on the command or response code.

Sending Packets once or cyclic

- To send packet once, select **Put packet**.
- To send packet cyclic, select **Put cyclic**.

9.2.2 Receiving Packet

The screenshot shows a software interface for receiving packets. On the left, under 'Receive', is the 'Packet header' section with fields for: Dest (00000001), Src (00000000), State (00000000), Dest ID (00000000), Cmd (00002F01), Src ID (00000000), Ext (00000000), Len (00000012), Rout (00000000), and ID (0000003E). On the right, under 'Receive data:', is a table with columns 0-9 and a 'Counter: 0' label. The data row shows: 0: 00, 1: 00, 2: 00, 3: 00, 4: 00, 5: 00, 6: 00, 7: 00, 8: 00, 9: 04. Below the data row are rows for indices 10, 20, 30, 40, 50, and 60, all currently empty.

Figure 91: Packet Header and Receive Data

Packet Header

Under **Receive > Packet Header** the elements of the packet header of the receiving packet are displayed, which are transmitted back from the device to the application (configuration software). The packet header of the receiving packets contain the elements described in the following table.

Element		Description
Dest	Destination Queue Handle	Contains the identifier of the receiver for the packet (destination task queue of the firmware).
Src	Source Queue Handle	Contains the identifier of the sender of the packet (sending task).
Dest ID	Destination Queue Reference	Contains an identifier for the receiver of unsolicited sent packets from the firmware to the application (configuration software).
Src ID	Source Queue Reference	Contains an identifier of the sender.
Len	Packet Data Length (in Bytes)	Length of the send respectively receive data.
ID	Packet Identification As Unique Number	Identifies identical data packets among each other.
State	Status / Error Code	Transmits status or error codes to the packet sender.
Cmd	Command / Response Code	Command or respond code.
Ext	Extension	Field for extensions (reserved).
Rout	Routing Information	Internal value of the firmware.

Table 55: Descriptions Packet Header

Receive Data

Under **Receive > Receive Data** the receiving data of the packet, which is transmitted back from the device to the application (configuration software) are displayed.

9.2.3 Example – Reading Data via SDO Upload

To read data from a Slave device via **SDO Upload**, using the packet monitor, you must proceed as described below. The single steps are explained with the help of sample data.



For more information on the steps under **Settings** and **Configuration** refer to the corresponding chapters in this manual. Required information is easy to find via the overview in the section *Configuration Steps* on page 23.

Requirements

The function **SDO Upload** can only be used if:

- The used device supports **SDO Upload** and answers SDO Upload requests.
- The used Master and Slave devices are configured.
- The used Master and Slave devices are operational.

How to proceed

In the Master DTM under **Settings**:

- Select the cifX device driver and assign the Master device.

Alternative to the connection via the cifX device driver via the netX driver a USB, serial or TCP connection can be used.

Packet Description Read Request

Structure Information CANOPEN_MASTER_PACKET_SDO_REQ_T			Type: Request
Variable	Type	Value / Range	Description
Structure TLR_PACKET_HEADER_T			
ulDest	UINT32	0x20/ QUE_CANOPENMST	Destination Queue-Handle of CANopen Master-Task Process Queue
ulSrc	UINT32	0 ... 2 ³² -1	Source Queue-Handle of AP-Task Process Queue
ulDestId	UINT32	ulCANOPENMSTId	Destination End Point Identifier, specifying the final receiver of the packet within the Destination Process
ulSrcId	UINT32	ulAPMS0Id	Source End Point Identifier, specifying the origin of the packet inside the Source Process
ulLen	UINT32	16	Packet Data Length in bytes
ulId	UINT32	0 ... 2 ³² -1	Packet Identification as unique number generated by the Source Process of the Packet
ulSta	UINT32		See section 6.2 Codes of the CANopen Master-Task
ulCmd	UINT32	0x00002814	CANOPEN_MASTER_SDO_UPLOAD_REQ - Command
ulExt	UINT32	0	Reserved
ulRout	UINT32	x	Routing Information
Structure CANOPEN_MASTER_SDO_REQ_DATA_T			
ulNodeId	UINT32	1 ... 127	Node ID of the node of the CANopen network to be read
ulIndex	UINT32	0 ... 65535	Index
ulSubIndex	UINT32	0 ... 255	Sub index
ulDataCnt	UINT32	1 ... 512	Number of data bytes to read
abSdoData [512]	UINT8[]		Unused

Figure 92: Packet Description CANOPEN_MASTER_PACKET_SDO_REQ_T – SDO Upload Request



Further information on the packet description for this example are comprised in the *CANopen Master Protocol API Manual*, in section *CANOPEN_MASTER_SDO_UPLOAD_REQ/CNF – SDO Upload* (e. g. *Revision 12* of the API manual, in section 5.2.10.) .

1. Open Packet Monitor.
- **Tools > Packet Monitor.**
2. Enter data for Packet header.

Send > Packet header

- Under **Dest** enter or select the receiver.
- Under **Cmd** enter the read command identification.

	<i>Sample Data</i>	
(A) Receiver	<i>0x20</i>	(Destination Queue Handle)
(C) Read Command Identification	<i>00002814</i>	CANOPEN_MASTER_PACKET_ SDO_REQ_T (Request)

Send

Packet header

Dest: **(A)** 00000020

Src: 00000000 State: 00000000

Dest ID: 00000000 Cmd: **(C)** 00002814

Src ID: 00000000 Ext: 00000000

Len: 00000010 Rout: 00000000

ID: 00000002 Auto Inkrement ID ☐

Figure 93: Example - Reading data via SDO-Upload – Send > Packet header

3. Enter send data.

Send > Send Data

	Range of Value	Sample Data	
(D) Node ID	0 ... 127	02 00 00 00*	(for Slave 2)
(E) Object Index	0 ... 65535	18 10 00 00*	(for object index 0x1018)
(F) Subindex	0 ... 255	04 00 00 00*	(for Sub index 4)
(G) Indication of Length	1 ... 512	04 00 00 00*	(for the Indication of Length 4 Bytes)
	(UINT32 = 4 Bytes)	*Intel format, e. g. LSB first;	

Send data: Counter: 3

	00	01	02	03	04	05	06	07	08	09
0000	02	00	(D)00	00	18	10	(E)00	00	04	(F)00
000A	(F)00	04	00	(G)00	00					
0014										
001E										
0028										
0032										
003C										

Put cyclic Put packet

Figure 94: Example - Reading data via SDO-Upload –Send > Send Data

4. Sending/Receiving Packets.

- To send packet once, select **Put packet**.
- The entered values are sent from the packet monitor to the connected Slave device and received packets are displayed.

Packet Description Read Confirmation







Structure Information CANOPEN_MASTER_PACKET_SDO_CNF_T			Type: Confirmation
Variable	Type	Value / Range	Description
Structure TLR_PACKET_HEADER_T			
ulDest 	UINT32		Destination Queue-Handle, untouched
ulSrc	UINT32		Source Queue-Handle, untouched
ulDestId	UINT32	ulAPMS0Id	Destination End Point Identifier, untouched
ulSrcId	UINT32	ulCANOPENMST0Id	Source End Point Identifier, untouched
ulLen	UINT32	16 ... 528	Packet Data Length in bytes
ulId	UINT32	0 ... $2^{32}-1$	Packet Identification as unique number generated by the Source Process of the Packet
ulSta	UINT32		See section 6.2 Codes of the CANopen Master-Task
ulCmd 	UINT32	0x00002815	CANOPEN_MASTER_SDO_UPLOAD_CNF - Command
ulExt	UINT32		Extension, reserved
ulRout	UINT32		Routing Information, do not change
Structure CANOPEN_MASTER_SDO_CNF_DATA_T			
ulNodeId 	UINT32	1 ... 127	Node ID of the node to be read
ulIndex 	UINT32	0 ... 65535	Index
ulSubIndex 	UINT32	0 ... 255	Sub index
ulDataCnt 	UINT32	1 ... 512	Data count
abSdoData [512]	UINT8[]		SDO upload data or if available, the error code the node returns via CANopen

Figure 95: Packet Description CANOPEN_MASTER_PACKET_SDO_CNF_T – SDO Upload Confirmation



Further information on the packet description for this example are comprised in the *CANopen Master Protocol API Manual*, in section *CANOPEN_MASTER_SDO_UPLOAD_REQ/CNF – SDO Upload* (e. g. *Revision 12* of the API manual, in section 5.2.10.) .

5. Evaluate received packet.

Receive > Packet header

- Under **Dest** the receiver is displayed.
- Under **State** the status code or possibly an error code is displayed.



All status and error codes are either to be found in this manual via the section *Overview Error Codes* on page 149, in the *Dual Port Memory Manual* or in the *CANopen Slave Protocol API Manual*.

- Under **Cmd** the response identification read request is displayed.

Receive	
Packet header	
Dest:	A 00000020
Src:	00000000
State:	B 00000000
Dest ID:	00000000
Cmd:	C 00002815
Src ID:	00000000
Ext:	00000000
Len:	00000014
Rout:	00000000
ID:	00000002

Figure 96: Example - Reading data via SDO-Upload – Receive > Packet header

	Sample Data	
A Receiver	0x20	(Destination Queue Handle)
B State	00000000	indicates that the read request could be executed without error. If during the execution of the read request an error was detected, an error code will be displayed.
C Response Identification Read Request	00002815	CANOPEN_MASTER_PACKET_SDO_CNF_T (Confirmation)

Receive > Receive data

➤ Under **Receive data** the data of the receive packet are displayed.

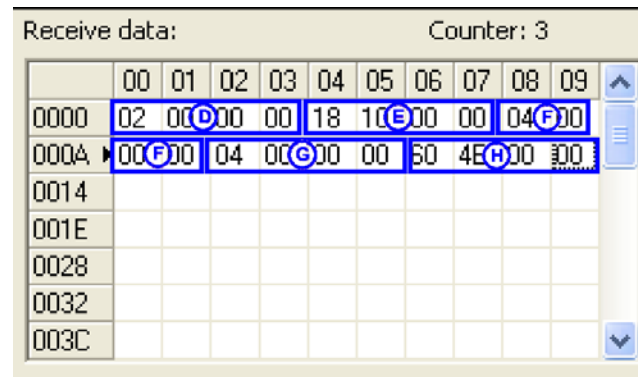


Figure 97: Example - Reading data via SDO-Upload – Receive > Receive data

	Range of Value	Sample Data	
(D) Node ID	0 ... 127	02 00 00 00*	(for Slave 2)
(E) Object Index	0 ... 65535	18 10 00 00*	(for object index 0x1018)
(F) Subindex	0 ... 255	04 00 00 00*	(for Sub index 4)
(G) Indication of Length	1 ... 512	04 00 00 00*	(for the Indication of Length 4 Bytes)
(H) Receive Data		60 4E 00 00*	(4 Bytes)
	(UINT32 = 4 Bytes)	*Intel format, e. g. LSB first;	

9.2.4 Example – Writing Data via SDO Download

To read data from a Slave device via **SDO Download**, using the packet monitor, you must proceed as described below. The single steps are explained with the help of sample data.



For more information on the steps under **Settings** and **Configuration** refer to the corresponding chapters in this manual. Required information is easy to find via the overview in the section *Configuration Steps* on page 23.

Requirements

The function **SDO Download** can only be used if:

- The used device supports **SDO Download** and answers SDO Upload requests.
- The used Master and Slave devices are configured.
- The used Master and Slave devices are operational.

How to proceed

In the Master DTM under **Settings**:

- Select the cifX device driver and assign the Master device.

Alternative to the connection via the cifX device driver via the netX driver a USB, serial or TCP connection can be used.

Packet Description Write Request

Structure Information CANOPEN_MASTER_PACKET_SDO_REQ_T			Type: Request
Variable	Type	Value / Range	Description
Structure TLR_PACKET_HEADER_T			
ulDest (A)	UINT32	0x20/ QUE_CANOPENMST	Destination Queue-Handle of CANopen Master-Task Process Queue
ulSrc	UINT32	0 ... $2^{32}-1$	Source Queue-Handle of AP-Task Process Queue
ulDestId	UINT32	ulCANOPENMSTId	Destination End Point Identifier, specifying the final receiver of the packet within the Destination Process
ulSrcId	UINT32	ulAPMS0Id	Source End Point Identifier, specifying the origin of the packet inside the Source Process
ulLen	UINT32	17 ... 528	Packet Data Length in bytes
ulId	UINT32	0 ... $2^{32}-1$	Packet Identification as unique number generated by the Source Process of the Packet
ulSta	UINT32		See section 6.2 Codes of the CANopen Master-Task
ulCmd (C)	UINT32	0x00002816	CANOPEN_MASTER_SDO_DOWNLOAD_REQ - Command
ulExt	UINT32	0	Reserved
ulRout	UINT32	x	Routing Information
Structure CANOPEN_MASTER_SDO_REQ_DATA_T			
ulNodeId (D)	UINT32	1 ... 127	Node ID of the node of the CANopen network to be read
ulIndex (E)	UINT32	0 ... 65535	Index
ulSubIndex (F)	UINT32	0 ... 255	Sub index
ulDataCnt (G)	UINT32	1 ... 512	Number of data bytes to write
abSdoData [512]	UINT8[]		SDO download data

Figure 98: Packet Description CANOPEN_MASTER_PACKET_SDO_REQ_T – SDO Download Request



Further information on the packet description for this example are comprised in the *CANopen Master Protocol API Manual*, in section *CANOPEN_MASTER_SDO_DOWNLOAD_REQ/CNF – SDO Download* (e. g. *Revision 12* of the API manual, in section 5.2.11.) .

1. Open Packet Monitor.
- **Tools > Packet Monitor.**
2. Enter data for Packet header.

Send > Packet header

- Under **Dest** enter or select the receiver.
- Under **Cmd** enter the write command identification.

	<i>Sample Data</i>	
(A) Receiver	<i>0x20</i>	(Destination Queue Handle)
(C) Write Command Identification	<i>00002816</i>	CANOPEN_MASTER_PACKET_SDO_REQ_T (Request)

Send

Packet header

Dest: **(A)** 00000020

Src: 00000000 State: 00000000

Dest ID: 00000000 Cmd: **(C)** 00002816

Src ID: 00000000 Ext: 00000000

Len: 00000014 Rout: 00000000

ID: 00000002 Auto Inkrement ID ☐

Figure 99: Example - Writing data via SDO-Download – Send > Packet header

3. Enter send data.

Send > Send Data

	Range of Value	Sample Data	
(D) Node ID	0 ... 127	02 00 00 00*	(for Slave 2)
(E) Object Index	0 ... 65535	00 60 00 00*	(for object index 0x6000)
(F) Subindex	0 ... 255	00 00 00 00*	(for Sub index 0x0000)
(G) Indication of Length	1 ... 512	04 00 00 00*	(for the Indication of Length 4 Bytes)
(I) Data to be sent		82 00 00 00*	(4 Bytes)
	(UINT32 = 4 Bytes)	*Intel format, e. g. LSB first;	

Send data: Counter: 3

	00	01	02	03	04	05	06	07	08	09
0000	02	00	(D)00	00	60	00	(E)00	00	00	(F)00
0004	(F)00	04	00	(G)00	00	82	00	(I)00	00	00
0014										
001E										
0028										
0032										
003C										

Buttons:

Figure 100: Example - Writing data via SDO-Download – Send > Send Data

4. Sending/Receiving Packets.

- To send packet once, select **Put packet**.
- The entered values are sent from the packet monitor to the connected Slave device and received packets are displayed.

Packet Description Write Confirmation

Structure Information CANOPEN_MASTER_PACKET_SDO_CNF_T			Type: Confirmation
Variable	Type	Value / Range	Description
Structure TLR_PACKET_HEADER_T			
ulDest (A)	UINT32		Destination Queue-Handle, untouched
ulSrc	UINT32		Source Queue-Handle, untouched
ulDestId	UINT32	ulAPMS0Id	Destination End Point Identifier, untouched
ulSrcId	UINT32	ulCANOPENMST0Id	Source End Point Identifier, untouched
ulLen	UINT32	16, 20	Packet Data Length in bytes
ulId	UINT32	0 ... $2^{32}-1$	Packet Identification as unique number generated by the Source Process of the Packet
ulSta	UINT32		See section 6.2 Codes of the CANopen Master-Task
ulCmd (C)	UINT32	0x00002817	CANOPEN_MASTER_SDO_DOWNLOAD_CNF - Command
ulExt	UINT32		Extension, reserved
ulRout	UINT32		Routing Information, do not change
Structure CANOPEN_MASTER_SDO_CNF_DATA_T			
ulNodeId (D)	UINT32	1 ... 127	Node ID of the node to be read
ulIndex (E)	UINT32	0 ... 65535	Index
ulSubIndex (F)	UINT32	0 ... 255	Sub index
ulDataCnt (G)	UINT32	1 ... 512	Data count
abSdoData [512]	UINT8[]		The error code the node returns via CANopen

Figure 101: Packet Description CANOPEN_MASTER_PACKET_SDO_CNF_T – SDO Download Confirmation



Further information on the packet description for this example are comprised in the *CANopen Master Protocol API Manual*, in section *CANOPEN_MASTER_SDO_DOWNLOAD_REQ/CNF – SDO Download* (e. g. *Revision 12* of the API manual, in section 5.2.11.) .

5. Evaluate received packet.

Receive > Packet header

- Under **Dest** the receiver is displayed.
- Under **State** the status code or possibly an error code is displayed.



All status and error codes are either to be found in this manual via the section *Overview Error Codes* on page 149, in the *Dual Port Memory Manual* or in the *CANopen Slave Protocol API Manual*.

- Under **Cmd** the response identification write request is displayed.

Receive	
Packet header	
Dest:	A 00000020
Src:	00000000
State:	B 00000000
Dest ID:	00000000
Cmd:	C 00002817
Src ID:	00000000
Ext:	00000000
Len:	00000010
Rout:	00000000
ID:	00000002

Figure 102: Example - Writing data via SDO-Download – Receive > Packet header

	Sample Data	
A Receiver	0x20	(Destination Queue Handle)
B State	00000000	indicates that the write request could be executed without error. If during the execution of the write request an error was detected, an error code will be displayed.
C Response Identification Writing Request	00002817	CANOPEN_MASTER_PACKET_SEND_EMCY_REQ_T (Confirmation)

Receive > Receive data

Under **Receive data** the data of the receive packet are displayed.

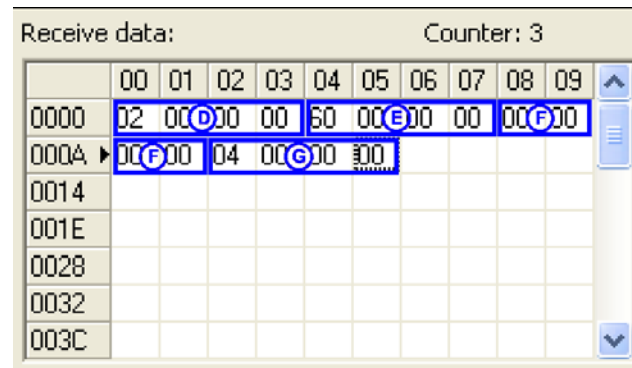


Figure 103: Example – Writing data via SDO-Download – Receive > Receive data

		Range of Value	Sample Data	
D	Node ID	0 ... 127	02 00 00 00*	(for Slave 2)
E	Object Index	0 ... 65535	00 60 00 00*	(for object index 0x6000)
F	Subindex	0 ... 255	00 00 00 00*	(for Sub index 0x0000)
G	Indication of Length	1 ... 512	04 00 00 00*	(for the Indication of Length 4 Bytes)
		(UINT32 = 4 Bytes)	*Intel format, e. g. LSB first;	

9.3 IO Monitor

The **IO Monitor** serves for test and diagnosis purposes. It provides to view data of the process data image and to change output data easily. The display is always in a Byte manner.



Note: Only change and write output data if you know that no plant disturbances are caused by this. All output data written by the IO Monitor are transmitted at the bus and have effect on subordinate drives, IO etc.

IO Monitor

Columns: 10 Display mode: Decimal

Input data

Offset: 0 Go

	0	1	2	3	4	5	6	7	8	9
0	227	207	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0
40	0	0	0	0	0	0	0	0	0	0
50	0	0	0	0	0	0	0	0	0	0
60	n	n	n	n	n	n	n	n	n	n

Output data

Offset: 0 Go

	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0
40	0	0	0	0	0	0	0	0	0	0
50	0	0	0	0	0	0	0	0	0	0
60	n	n	n	n	n	n	n	n	n	n

Update

Figure 104: IO Monitor

Columns switches the number of columns.

Display mode switches the representation of the input and output data between decimal and hexadecimal.

Offset / Go moves the indication of the data to the entered offset value.

➤ Enter the output value and select **Update**.

➤ Always the data of the process image are displayed, also when these Bytes have not been reserved by the configuration.

9.4 Process Image Monitor

The window **Process Image Monitor** lists the Slave devices connected to the Master, as well as the configured modules or input or output signals of the devices. This makes visible the fieldbus structure and the data structure of the device's input and output data transmitted at the bus. Furthermore the values of the signal data provided to the OPC server are displayed here.

➤ Open **Tools > Process Image Monitor**.

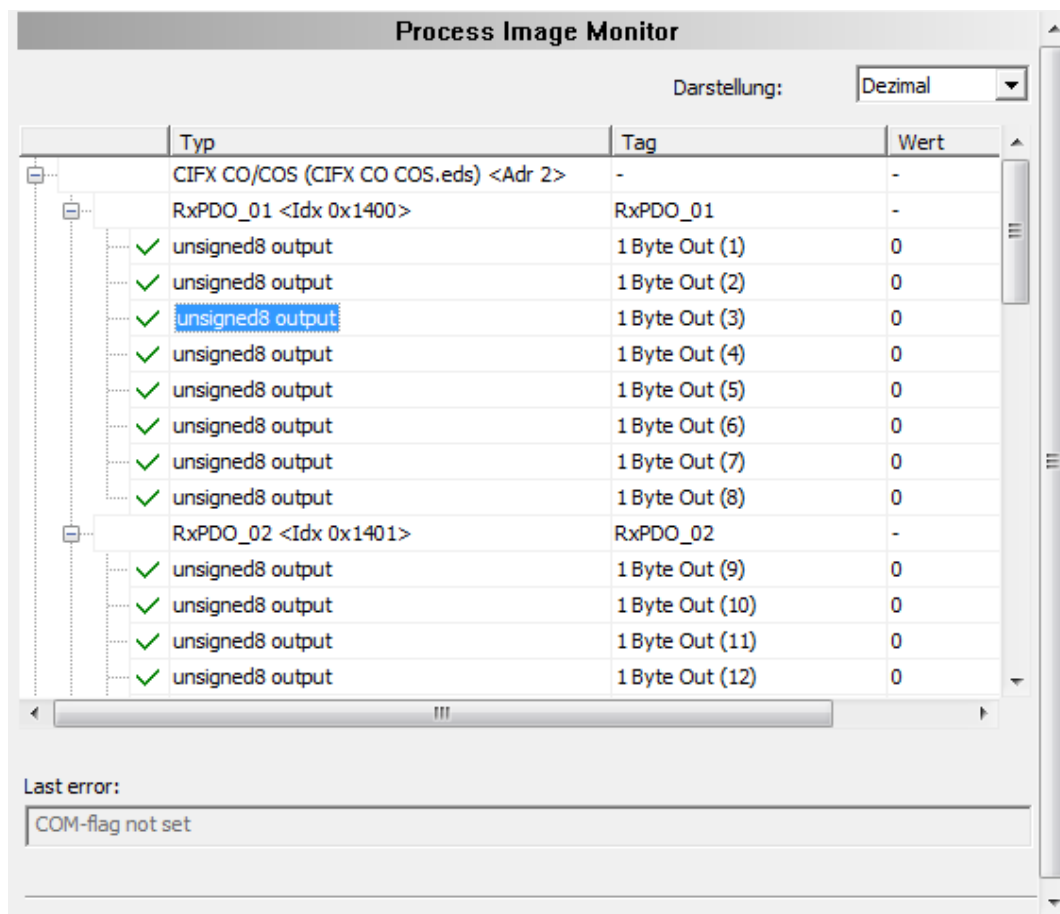


Figure 105: Window 'Process Image Monitor'

Parameter	Meaning	Range of Value / Value
Display Mode	Display of the values in the column Value in decimal or hexadecimal mode.	Decimal (Default), Hexadecimal
	The tree shows the structure of the devices (1), modules (2) and the input data (3) and output data (4).	
	Display when the input and output data are not completely read and analyzed.	
	Display when the input and output data are not valid.	
	Display when the input and output data are valid.	
Typ	Device labeling provided by the hardware: Also description of the modules or input or output signals configured to the device.	

Parameter	Meaning	Range of Value / Value
TAG	Device name provided by the hardware (not changeable in the FDT container) or symbolic name for the modules configured to the device or for the input or output signals (changeable in the window Configuration > Process Data).	
Value	Display of the valid input and output data values.	
Last Error	Last occurred error (Description see appropriate Application Programming Manual)	

Table 56: Notes to the Window 'Process Image Monitor'

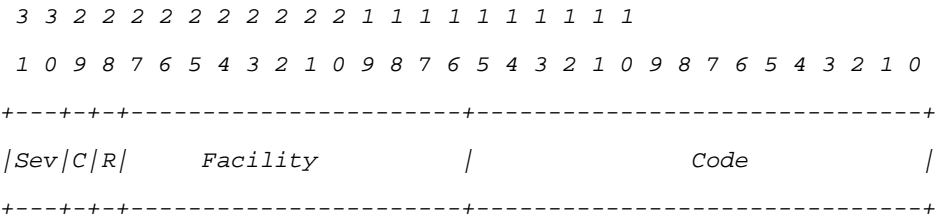
10 Error Codes

10.1 Error Code Definition

For COM based application, like the ODM Server and ODM drivers, a common error definition is used, similar to the Microsoft Windows® HRESULT definition.

Error Code Structure:

COM Errors are HRESULTs, which are 32 bit values using the following layout:



where

- Sev - is the severity code:
 - 00 - Success
 - 01 - Informational
 - 10 - Warning
 - 11 - Error
- C - is the Customer code flag
- R - is a reserved bit
- Facility - is the facility code
- Code - is the facility's status code

In this common error definition, several error code regions are already reserved by Windows® itself, the ODM and some other modules.

10.2 Overview Error Codes

Overview Error Codes	Range
General Hardware Errors RCX Operating System	<i>RCX General Task:</i> 0xC02B0001 to 0xC02B4D52
	<i>RCX Common Status & Errors Codes:</i> 0x00000000 to 0xC002000C
	<i>RCX Status & Error Codes:</i> 0x00000000 to 0xC0000008
ODM Server	<i>General ODM Error Codes:</i> 0x8004C700 to 0x8004C761
	<i>General ODM Driver Error Codes:</i> 0x8004C7A0 to 0x8004C7C2
ODM Drivers	<i>cifX Driver Specific ODM Error:</i> 0x8004C001 to 0x8004C0A4
cifX Device Driver and netX Driver	<i>Generic Error:</i> 0x800A0001 to 0x800A0017
	<i>Generic Driver Error:</i> 0x800B0001 to 0x800B0042
	<i>Generic Device Error:</i> 0x800C0010 to 0x800C0041
netX Driver	<i>CIFX API Transport:</i> 0x800D0001 to 0x800D0013
	<i>CIFX API Transport Header State Error:</i> 0x800E0001 to 0x800E000B
DBM	<i>ODM Error Codes:</i> 0xC004C810 to 0xC004C878

Table 57: Overview Error Codes and Ranges



The fieldbus specific error codes are described in the manuals of the corresponding protocol tasks.

10.3 General Hardware Error Codes

10.3.1 RCX General Task Errors

Error Code (Definition)	Value	Description
RCX_E_QUE_UNKNOWN	0xC02B0001	Unknown Queue
RCX_E_QUE_INDEX_UNKNOWN	0xC02B0002	Unknown Queue Index
RCX_E_TASK_UNKNOWN	0xC02B0003	Unknown Task
RCX_E_TASK_INDEX_UNKNOWN	0xC02B0004	Unknown Task Index
RCX_E_TASK_HANDLE_INVALID	0xC02B0005	Invalid Task Handle
RCX_E_TASK_INFO_IDX_UNKNOWN	0xC02B0006	Unknown Index
RCX_E_FILE_XFR_TYPE_INVALID	0xC02B0007	Invalid Transfer Type
RCX_E_FILE_REQUEST_INCORRECT	0xC02B0008	Invalid File Request
RCX_E_TASK_INVALID	0xC02B000E	Invalid Task
RCX_E_SEC_FAILED	0xC02B001D	Security EEPROM Access Failed
RCX_E_EEPROM_DISABLED	0xC02B001E	EEPROM Disabled
RCX_E_INVALID_EXT	0xC02B001F	Invalid Extension
RCX_E_SIZE_OUT_OF_RANGE	0xC02B0020	Block Size Out Of Range
RCX_E_INVALID_CHANNEL	0xC02B0021	Invalid Channel
RCX_E_INVALID_FILE_LEN	0xC02B0022	Invalid File Length
RCX_E_INVALID_CHAR_FOUND	0xC02B0023	Invalid Character Found
RCX_E_PACKET_OUT_OF_SEQ	0xC02B0024	Packet Out Of Sequence
RCX_E_SEC_NOT_ALLOWED	0xC02B0025	Not Allowed In Current State
RCX_E_SEC_INVALID_ZONE	0xC02B0026	Security EEPROM Invalid Zone
RCX_E_SEC_EEPROM_NOT_AVAIL	0xC02B0028	Security EEPROM Eeprom Not Available
RCX_E_SEC_INVALID_CHECKSUM	0xC02B0029	Security EEPROM Invalid Checksum
RCX_E_SEC_ZONE_NOT_WRITEABLE	0xC02B002A	Security EEPROM Zone Not Writeable
RCX_E_SEC_READ_FAILED	0xC02B002B	Security EEPROM Read Failed
RCX_E_SEC_WRITE_FAILED	0xC02B002C	Security EEPROM Write Failed
RCX_E_SEC_ACCESS_DENIED	0xC02B002D	Security EEPROM Access Denied
RCX_E_SEC_EEPROM_EMULATED	0xC02B002E	Security EEPROM Emulated
RCX_E_INVALID_BLOCK	0xC02B0038	Invalid Block
RCX_E_INVALID_STRUCT_NUMBER	0xC02B0039	Invalid Structure Number
RCX_E_INVALID_CHECKSUM	0xC02B4352	Invalid Checksum
RCX_E_CONFIG_LOCKED	0xC02B4B54	Configuration Locked
RCX_E_SEC_ZONE_NOT_READABLE	0xC02B4D52	Security EEPROM Zone Not Readable

Table 58: RCX General Task Errors

10.3.2 RCX Common Status & Errors Codes

Error Code (Definition)	Value	Description
RCX_S_OK	0x00000000	Success, Status Okay
RCX_E_FAIL	0xC0000001	Fail
RCX_E_UNEXPECTED	0xC0000002	Unexpected
RCX_E_OUTOFMEMORY	0xC0000003	Out Of Memory
RCX_E_UNKNOWN_COMMAND	0xC0000004	Unknown Command
RCX_E_UNKNOWN_DESTINATION	0xC0000005	Unknown Destination
RCX_E_UNKNOWN_DESTINATION_ID	0xC0000006	Unknown Destination ID
RCX_E_INVALID_PACKET_LEN	0xC0000007	Invalid Packet Length
RCX_E_INVALID_EXTENSION	0xC0000008	Invalid Extension
RCX_E_INVALID_PARAMETER	0xC0000009	Invalid Parameter
RCX_E_WATCHDOG_TIMEOUT	0xC000000C	Watchdog Timeout
RCX_E_INVALID_LIST_TYPE	0xC000000D	Invalid List Type
RCX_E_UNKNOWN_HANDLE	0xC000000E	Unknown Handle
RCX_E_PACKET_OUT_OF_SEQ	0xC000000F	Out Of Sequence
RCX_E_PACKET_OUT_OF_MEMORY	0xC0000010	Out Of Memory
RCX_E_QUE_PACKETDONE	0xC0000011	Queue Packet Done
RCX_E_QUE_SENDPACKET	0xC0000012	Queue Send Packet
RCX_E_POOL_PACKET_GET	0xC0000013	Pool Packet Get
RCX_E_POOL_GET_LOAD	0xC0000015	Pool Get Load
RCX_E_REQUEST_RUNNING	0xC000001A	Request Already Running
RCX_E_INIT_FAULT	0xC0000100	Initialization Fault
RCX_E_DATABASE_ACCESS_FAILED	0xC0000101	Database Access Failed
RCX_E_NOT_CONFIGURED	0xC0000119	Not Configured
RCX_E_CONFIGURATION_FAULT	0xC0000120	Configuration Fault
RCX_E_INCONSISTENT_DATA_SET	0xC0000121	Inconsistent Data Set
RCX_E_DATA_SET_MISMATCH	0xC0000122	Data Set Mismatch
RCX_E_INSUFFICIENT_LICENSE	0xC0000123	Insufficient License
RCX_E_PARAMETER_ERROR	0xC0000124	Parameter Error
RCX_E_INVALID_NETWORK_ADDRESS	0xC0000125	Invalid Network Address
RCX_E_NO_SECURITY_MEMORY	0xC0000126	No Security Memory
RCX_E_NETWORK_FAULT	0xC0000140	Network Fault
RCX_E_CONNECTION_CLOSED	0xC0000141	Connection Closed
RCX_E_CONNECTION_TIMEOUT	0xC0000142	Connection Timeout
RCX_E_LONELY_NETWORK	0xC0000143	Lonely Network
RCX_E_DUPLICATE_NODE	0xC0000144	Duplicate Node
RCX_E_CABLE_DISCONNECT	0xC0000145	Cable Disconnected
RCX_E_BUS_OFF	0xC0000180	Network Node Bus Off
RCX_E_CONFIG_LOCKED	0xC0000181	Configuration Locked
RCX_E_APPLICATION_NOT_READY	0xC0000182	Application Not Ready
RCX_E_TIMER_APPL_PACKET_SENT	0xC002000C	Timer App Packet Sent

Table 59: RCX Common Status & Errors Codes

10.3.3 RCX Status & Error Codes

Error Code (Definition)	Value	Description
RCX_S_OK	0x00000000	SUCCESS, STATUS OKAY
RCX_S_QUE_UNKNOWN	0xC02B0001	UNKNOWN QUEUE
RCX_S_QUE_INDEX_UNKNOWN	0xC02B0002	UNKNOWN QUEUE INDEX
RCX_S_TASK_UNKNOWN	0xC02B0003	UNKNOWN TASK
RCX_S_TASK_INDEX_UNKNOWN	0xC02B0004	UNKNOWN TASK INDEX
RCX_S_TASK_HANDLE_INVALID	0xC02B0005	INVALID TASK HANDLE
RCX_S_TASK_INFO_IDX_UNKNOWN	0xC02B0006	UNKNOWN INDEX
RCX_S_FILE_XFR_TYPE_INVALID	0xC02B0007	INVALID TRANSFER TYPE
RCX_S_FILE_REQUEST_INCORRECT	0xC02B0008	INVALID FILE REQUEST
RCX_S_UNKNOWN_DESTINATION	0xC0000005	UNKNOWN DESTINATION
RCX_S_UNKNOWN_DESTINATION_ID	0xC0000006	UNKNOWN DESTINATION ID
RCX_S_INVALID_LENGTH	0xC0000007	INVALID LENGTH
RCX_S_UNKNOWN_COMMAND	0xC0000004	UNKNOWN COMMAND
RCX_S_INVALID_EXTENSION	0xC0000008	INVALID EXTENSION

Table 60: RCX Status & Error Codes

10.3.3.1 RCX Status & Error Codes Slave State

Error Code (Definition)	Value	Description
RCX_SLAVE_STATE_UNDEFINED	0x00000000	UNDEFINED
RCX_SLAVE_STATE_OK	0x00000001	OK
RCX_SLAVE_STATE_FAILED	0x00000002	FAILED (at least one slave)

Table 61: RCX Status & Error Codes Slave State

10.4 ODM Error Codes

10.4.1 General ODM Error Codes

Error Code (Definition)	Value	Description
CODM3_E_INTERNALERROR	0x8004C700	Internal ODM Error
ODM3_E_DESCRIPTION_NOTFOUND	0x8004C701	Description not found in ODM database
CODM3_E_WRITEREGISTRY	0x8004C710	Error writing to the registry
CODM3_E_BAD_REGULAR_EXPRESSION	0x8004C711	Invalid regular expression
CODM3_E_COMCATEGORIE_MANAGER_FAILED	0x8004C712	Component Category Manager could not be instantiated
CODM3_E_COMCATEGORIE_ENUMERATION_FAILED	0x8004C713	Driver could not be enumerated by the Category Manager
CODM3_E_CREATE_LOCAL_BUFFER	0x8004C714	Error creating local buffers
CODM3_E_UNKNOWNHANDLE	0x8004C715	Unknown handle
CODM3_E_QUEUE_LIMIT_REACHED	0x8004C717	Queue size limit for connection reached
CODM3_E_DATASIZE_ZERO	0x8004C718	Zero data length passed
CODM3_E_INVALID_DATA	0x8004C719	Invalid data content
CODM3_E_INVALID_MODE	0x8004C71A	Invalid mode
CODM3_E_DATABASE_READ	0x8004C71B	Error reading database
CODM3_E_CREATE_DEVICE_THREAD	0x8004C750	Error creating device thread
CODM3_E_CREATE_DEVICE_THREAD_STOP_EVENT	0x8004C751	Error creating device thread stop event
CODM3_E_CLIENT_NOT_REGISTERED	0x8004C752	Client is not registered at the ODM
CODM3_E_NO_MORE_CLIENTS	0x8004C753	Maximum number of clients reached
CODM3_E_MAX_CLIENT_CONNECTIONS_REACHED	0x8004C754	Maximum number of client connections reached
CODM3_E_ENTRY_NOT_FOUND	0x8004C755	Driver/device not found
CODM3_E_DRIVER_NOT_FOUND	0x8004C757	The requested driver is unknown to the ODM
CODM3_E_DEVICE_ALREADY_LOCKED	0x8004C758	Device is locked by another process
CODM3_E_DEVICE_UNLOCKED_FAILED	0x8004C759	Device could not be unlocked, lock was set by another process
CODM3_E_DEVICE_LOCK_NECESSARY	0x8004C75A	Operation requires a device lock to be set
CODM3_E_DEVICE_SUBSCRIPTIONLIMIT	0x8004C75B	Maximum number of servers registered for this device reached
CODM3_E_DEVICE_NOTSUBSCRIBED	0x8004C75C	Process is not registered as a server on this device
CODM3_E_DEVICE_NO_MESSAGE	0x8004C75D	No message available
CODM3_E_TRANSFERTIMEOUT	0x8004C760	Message transfer timeout
CODM3_E_MESSAGE_INSERVICE	0x8004C761	Message in service

Table 62: ODM Error Codes - General ODM Error Codes

10.4.2 General ODM Driver Error Codes

Error Code (Definition)	Value	Description
CODM3_E_DRV_OPEN_DEVICE	0x8004C7A0	Packet type unsupported by driver
CODM3_E_DRV_INVALID_IDENTIFIER	0x8004C7A1	Invalid device identifier
CODM3_E_DRV_DEVICE_PARAMETERS_MISMATCH	0x8004C7A3	Parameters differ from requested device
CODM3_E_DRV_BROWSE_NO_DEVICES	0x8004C7A4	No devices found
CODM3_E_DRV_CREATE_DEVICE_INST	0x8004C7A5	Device instance could not be created
CODM3_E_DRV_DEVICE_NOMORE_TX	0x8004C7A6	Device connection limit reached
CODM3_E_DRV_DEVICE_DUPLICATE_TX	0x8004C7A7	Duplicate transmitter ID
CODM3_E_DRV_DEVICE_NOT_CONFIGURED	0x8004C7A8	Device is not configured
CODM3_E_DRV_DEVICE_COMMUNICATION	0x8004C7A9	Device communication error
CODM3_E_DRV_DEVICE_NO_MESSAGE	0x8004C7AA	No message available
CODM3_E_DRV_DEVICE_NOT_READY	0x8004C7AB	Device not ready
CODM3_E_DRV_INVALIDCONFIGURATION	0x8004C7AC	Invalid driver configuration
CODM3_E_DRV_DLINVALIDMODE	0x8004C7C0	Invalid download mode
CODM3_E_DRV_DLINPROGRESS	0x8004C7C1	Download is active
CODM3_E_DRV_ULINPROGRESS	0x8004C7C2	Upload is active

Table 63: ODM Error Codes - General ODM Driver Error Codes

10.4.3 cifX Driver Specific ODM Error Codes

cifX Driver Specific ODM Error Codes		
Error Code (Definition)	Value	Description
DRV_E_BOARD_NOT_INITIALIZED	0x8004C001	DRIVER Board not initialized
DRV_E_INIT_STATE_ERROR	0x8004C002	DRIVER Error in internal init state
DRV_E_READ_STATE_ERROR	0x8004C003	DRIVER Error in internal read state
DRV_E_CMD_ACTIVE	0x8004C004	DRIVER Command on this channel is active
DRV_E_PARAMETER_UNKNOWN	0x8004C005	DRIVER Unknown parameter in function
DRV_E_WRONG_DRIVER_VERSION	0x8004C006	DRIVER Version is incompatible with DLL
DRV_E_PCI_SET_CONFIG_MODE	0x8004C007	DRIVER Error during PCI set configuration mode
DRV_E_PCI_READ_DPM_LENGTH	0x8004C008	DRIVER Could not read PCI dual port memory length
DRV_E_PCI_SET_RUN_MODE	0x8004C009	DRIVER Error during PCI set run mode
DRV_E_DEV_DPM_ACCESS_ERROR	0x8004C00A	DEVICE Dual port ram not accessible(board not found)
DRV_E_DEV_NOT_READY	0x8004C00B	DEVICE Not ready (ready flag failed)
DRV_E_DEV_NOT_RUNNING	0x8004C00C	DEVICE Not running (running flag failed)
DRV_E_DEV_WATCHDOG_FAILED	0x8004C00D	DEVICE Watchdog test failed
DRV_E_DEV_OS_VERSION_ERROR	0x8004C00E	DEVICE Signals wrong OS version
DRV_E_DEV_SYSERR	0x8004C00F	DEVICE Error in dual port flags
DRV_E_DEV_MAILBOX_FULL	0x8004C010	DEVICE Send mailbox is full
DRV_E_DEV_PUT_TIMEOUT	0x8004C011	DEVICE PutMessage timeout
DRV_E_DEV_GET_TIMEOUT	0x8004C012	DEVICE GetMessage timeout
DRV_E_DEV_GET_NO_MESSAGE	0x8004C013	DEVICE No message available
DRV_E_DEV_RESET_TIMEOUT	0x8004C014	DEVICE RESET command timeout
DRV_E_DEV_NO_COM_FLAG	0x8004C015	DEVICE COM-flag not set. Check if Bus is running
DRV_E_DEV_EXCHANGE_FAILED	0x8004C016	DEVICE I/O data exchange failed
DRV_E_DEV_EXCHANGE_TIMEOUT	0x8004C017	DEVICE I/O data exchange timeout
DRV_E_DEV_COM_MODE_UNKNOWN	0x8004C018	DEVICE I/O data mode unknown
DRV_E_DEV_FUNCTION_FAILED	0x8004C019	DEVICE Function call failed
DRV_E_DEV_DPMSIZE_MISMATCH	0x8004C01A	DEVICE DPM size differs from configuration
DRV_E_DEV_STATE_MODE_UNKNOWN	0x8004C01B	DEVICE State mode unknown
DRV_E_DEV_HW_PORT_IS_USED	0x8004C01C	DEVICE Output port already in use
DRV_E_USR_OPEN_ERROR	0x8004C01E	USER Driver not opened (device driver not loaded)
DRV_E_USR_INIT_DRV_ERROR	0x8004C01F	USER Can't connect to device
DRV_E_USR_NOT_INITIALIZED	0x8004C020	USER Board not initialized (DevInitBoard not called)
DRV_E_USR_COMM_ERR	0x8004C021	USER IOCTL function failed
DRV_E_USR_DEV_NUMBER_INVALID	0x8004C022	USER Parameter DeviceNumber invalid
DRV_E_USR_INFO_AREA_INVALID	0x8004C023	USER Parameter InfoArea unknown
DRV_E_USR_NUMBER_INVALID	0x8004C024	USER Parameter Number invalid
DRV_E_USR_MODE_INVALID	0x8004C025	USER Parameter Mode invalid
DRV_E_USR_MSG_BUF_NULL_PTR	0x8004C026	USER NULL pointer assignment
DRV_E_USR_MSG_BUF_TOO_SHORT	0x8004C027	USER Message buffer too small

cifX Driver Specific ODM Error Codes		
Error Code (Definition)	Value	Description
DRV_E_USR_SIZE_INVALID	0x8004C028	USER Parameter Size invalid
DRV_E_USR_SIZE_ZERO	0x8004C02A	USER Parameter Size with zero length
DRV_E_USR_SIZE_TOO_LONG	0x8004C02B	USER Parameter Size too long
DRV_E_USR_DEV_PTR_NULL	0x8004C02C	USER Device address null pointer
DRV_E_USR_BUF_PTR_NULL	0x8004C02D	USER Pointer to buffer is a null pointer
DRV_E_USR_SENDSIZE_TOO_LONG	0x8004C02E	USER Parameter SendSize too large
DRV_E_USR_RECVSIZE_TOO_LONG	0x8004C02F	USER Parameter ReceiveSize too large
DRV_E_USR_SENDBUF_PTR_NULL	0x8004C030	USER Pointer to send buffer is a null pointer
DRV_E_USR_RECVBUF_PTR_NULL	0x8004C031	USER Pointer to receive buffer is a null pointer
DRV_E_DMA_INSUFF_MEM	0x8004C032	DMA Memory allocation error
DRV_E_DMA_TIMEOUT_CH4	0x8004C033	DMA Read I/O timeout
DRV_E_DMA_TIMEOUT_CH5	0x8004C034	DMA Write I/O timeout
DRV_E_DMA_TIMEOUT_CH6	0x8004C035	DMA PCI transfer timeout
DRV_E_DMA_TIMEOUT_CH7	0x8004C036	DMA Download timeout
DRV_E_DMA_DB_DOWN_FAIL	0x8004C037	DMA Database download failed
DRV_E_DMA_FW_DOWN_FAIL	0x8004C038	DMA Firmware download failed
DRV_E_CLEAR_DB_FAIL	0x8004C039	DMA Clear database on the device failed
DRV_E_DEV_NO_VIRTUAL_MEM	0x8004C03C	DMA USER Virtual memory not available
DRV_E_DEV_UNMAP_VIRTUAL_MEM	0x8004C03D	DMA USER Unmap virtual memory failed
DRV_E_GENERAL_ERROR	0x8004C046	DRIVER General error
DRV_E_DMA_ERROR	0x8004C047	DRIVER General DMA error
DRV_E_WDG_IO_ERROR	0x8004C048	DRIVER I/O WatchDog failed
DRV_E_WDG_DEV_ERROR	0x8004C049	DRIVER Device Watchdog failed
DRV_E_USR_DRIVER_UNKNOWN	0x8004C050	USER Driver unknown
DRV_E_USR_DEVICE_NAME_INVALID	0x8004C051	USER Device name invalid
DRV_E_USR_DEVICE_NAME_UNKNOWN	0x8004C052	USER Device name unknown
DRV_E_USR_DEVICE_FUNC_NOTIMPL	0x8004C053	USER Device function not implemented
DRV_E_USR_FILE_OPEN_FAILED	0x8004C064	USER File could not be opened
DRV_E_USR_FILE_SIZE_ZERO	0x8004C065	USER File size zero
DRV_E_USR_FILE_NO_MEMORY	0x8004C066	USER Not enough memory to load file
DRV_E_USR_FILE_READ_FAILED	0x8004C067	USER File read failed
DRV_E_USR_INVALID_FILETYPE	0x8004C068	USER File type invalid
DRV_E_USR_FILENAME_INVALID	0x8004C069	USER Invalid filename
DRV_E_FW_FILE_OPEN_FAILED	0x8004C06E	USER Firmware file could not be opened
DRV_E_FW_FILE_SIZE_ZERO	0x8004C06F	USER Not enough memory to load firmware file
DRV_E_FW_FILE_NO_MEMORY	0x8004C070	USER Not enough memory to load firmware file
DRV_E_FW_FILE_READ_FAILED	0x8004C071	USER Firmware file read failed
DRV_E_FW_INVALID_FILETYPE	0x8004C072	USER Firmware file type invalid
DRV_E_FW_FILENAME_INVALID	0x8004C073	USER Firmware file name not valid
DRV_E_FW_DOWNLOAD_ERROR	0x8004C074	USER Firmware file download error
DRV_E_FW_FILENAME_NOT_FOUND	0x8004C075	USER Firmware file not found in the internal table
DRV_E_FW_BOOTLOADER_ACTIVE	0x8004C076	USER Firmware file BOOTLOADER active

cifX Driver Specific ODM Error Codes		
Error Code (Definition)	Value	Description
DRV_E_FW_NO_FILE_PATH	0x8004C077	USER Firmware file no file path
DRV_E_CF_FILE_OPEN_FAILED	0x8004C078	USER Configuration file could not be opened
DRV_E_CF_FILE_SIZE_ZERO	0x8004C079	USER Configuration file size zero
DRV_E_CF_FILE_NO_MEMORY	0x8004C07A	USER Not enough memory to load configuration file
DRV_E_CF_FILE_READ_FAILED	0x8004C07B	USER Configuration file read failed
DRV_E_CF_INVALID_FILETYPE	0x8004C07C	USER Configuration file type invalid
DRV_E_CF_FILENAME_INVALID	0x8004C07D	USER Configuration file name not valid
DRV_E_CF_DOWNLOAD_ERROR	0x8004C07E	USER Configuration file download error
DRV_E_CF_FILE_NO_SEGMENT	0x8004C07F	USER No flash segment in the configuration file
DRV_E_CF_DIFFERS_FROM_DBM	0x8004C080	USER Configuration file differs from database
DRV_E_DBM_SIZE_ZERO	0x8004C083	USER Database size zero
DRV_E_DBM_NO_MEMORY	0x8004C084	USER Not enough memory to upload database
DRV_E_DBM_READ_FAILED	0x8004C085	USER Database read failed
DRV_E_DBM_NO_FLASH_SEGMENT	0x8004C086	USER Database segment unknown
DEV_E_CF_INVALID_DESCRIPTOR_VERSION	0x8004C096	CONFIG Version of the descriptor table invalid
DEV_E_CF_INVALID_INPUT_OFFSET	0x8004C097	CONFIG Input offset is invalid
DEV_E_CF_NO_INPUT_SIZE	0x8004C098	CONFIG Input size is 0
DEV_E_CF_MISMATCH_INPUT_SIZE	0x8004C099	CONFIG Input size does not match configuration
DEV_E_CF_INVALID_OUTPUT_OFFSET	0x8004C09A	CONFIG Invalid output offset
DEV_E_CF_NO_OUTPUT_SIZE	0x8004C09B	CONFIG Output size is 0
DEV_E_CF_MISMATCH_OUTPUT_SIZE	0x8004C09C	CONFIG Output size does not match configuration
DEV_E_CF_STN_NOT_CONFIGURED	0x8004C09D	CONFIG Station not configured
DEV_E_CF_CANNOT_GET_STN_CONFIG	0x8004C09E	CONFIG Cannot get the Station configuration
DEV_E_CF_MODULE_DEF_MISSING	0x8004C09F	CONFIG Module definition is missing
DEV_E_CF_MISMATCH_EMPTY_SLOT	0x8004C0A0	CONFIG Empty slot mismatch
DEV_E_CF_MISMATCH_INPUT_OFFSET	0x8004C0A1	CONFIG Input offset mismatch
DEV_E_CF_MISMATCH_OUTPUT_OFFSET	0x8004C0A2	CONFIG Output offset mismatch
DEV_E_CF_MISMATCH_DATA_TYPE	0x8004C0A3	CONFIG Data type mismatch
DEV_E_CF_MODULE_DEF_MISSING_NO_SI	0x8004C0A4	CONFIG Module definition is missing,(no Slot/Idx)

Table 64: cifX Driver Specific ODM Error Codes

10.5 Error Codes cifX Device Driver and netX Driver

10.5.1 Generic Error Codes

Error Code (Definition)	Value	Description
CIFX_INVALID_POINTER	0x800A0001	Invalid pointer (NULL) passed to driver
CIFX_INVALID_BOARD	0x800A0002	No board with the given nameindex available
CIFX_INVALID_CHANNEL	0x800A0003	No channel with the given index available
CIFX_INVALID_HANDLE	0x800A0004	Invalid handle passed to driver
CIFX_INVALID_PARAMETER	0x800A0005	Invalid parameter
CIFX_INVALID_COMMAND	0x800A0006	Invalid command
CIFX_INVALID_BUFFERSIZE	0x800A0007	Invalid buffer size
CIFX_INVALID_ACCESS_SIZE	0x800A0008	Invalid access size
CIFX_FUNCTION_FAILED	0x800A0009	Function failed
CIFX_FILE_OPEN_FAILED	0x800A000A	File could not be opened
CIFX_FILE_SIZE_ZERO	0x800A000B	File size is zero
CIFX_FILE_LOAD_INSUFF_MEM	0x800A000C	Insufficient memory to load file
CIFX_FILE_CHECKSUM_ERROR	0x800A000D	File checksum compare failed
CIFX_FILE_READ_ERROR	0x800A000E	Error reading from file
CIFX_FILE_TYPE_INVALID	0x800A000F	Invalid file type
CIFX_FILE_NAME_INVALID	0x800A0010	Invalid file name
CIFX_FUNCTION_NOT_AVAILABLE	0x800A0011	Driver function not available
CIFX_BUFFER_TOO_SHORT	0x800A0012	Given buffer is too short
CIFX_MEMORY_MAPPING_FAILED	0x800A0013	Failed to map the memory
CIFX_NO_MORE_ENTRIES	0x800A0014	No more entries available
CIFX_CALLBACK_MODE_UNKNOWN	0x800A0015	Unknown callback handling mode
CIFX_CALLBACK_CREATE_EVENT_FAILED	0x800A0016	Failed to create callback events
CIFX_CALLBACK_CREATE_RECV_BUFFER	0x800A0017	Failed to create callback receive buffer

Table 65: Generic Error Codes

10.5.2 Generic Driver Error Codes

Error Code (Definition)	Value	Description
CIFX_DRV_NOT_INITIALIZED	0x800B0001	Driver not initialized
CIFX_DRV_INIT_STATE_ERROR	0x800B0002	Driver init state error
CIFX_DRV_READ_STATE_ERROR	0x800B0003	Driver read state error
CIFX_DRV_CMD_ACTIVE	0x800B0004	Command is active on device
CIFX_DRV_DOWNLOAD_FAILED	0x800B0005	General error during download
CIFX_DRV_WRONG_DRIVER_VERSION	0x800B0006	Wrong driver version
CIFX_DRV_DRIVER_NOT_LOADED	0x800B0030	CIFx driver is not running
CIFX_DRV_INIT_ERROR	0x800B0031	Failed to initialize the device
CIFX_DRV_CHANNEL_NOT_INITIALIZED	0x800B0032	Channel not initialized (xOpenChannel not called)
CIFX_DRV_IO_CONTROL_FAILED	0x800B0033	IOControl call failed
CIFX_DRV_NOT_OPENED(0x800B0034	Driver was not opened
CIFX_DRV_DOWNLOAD_STORAGE_UNKNOWN	0x800B0040	Unknown download storage type (RAMFLASH based) found
CIFX_DRV_DOWNLOAD_FW_WRONG_CHANNEL	0x800B0041	Channel number for a firmware download not supported
CIFX_DRV_DOWNLOAD_MODULE_NO_BASEOS	0x800B0042	Modules are not allowed without a Base OS firmware

Table 66: Generic Driver Error Codes

10.5.3 Generic Device Error Codes

Error Code (Definition)	Value	Description
CIFX_DEV_DPM_ACCESS_ERROR	0x800C0010	Dual port memory not accessible (board not found)
CIFX_DEV_NOT_READY	0x800C0011	Device not ready (ready flag failed)
CIFX_DEV_NOT_RUNNING	0x800C0012	Device not running (running flag failed)
CIFX_DEV_WATCHDOG_FAILED	0x800C0013	Watchdog test failed
CIFX_DEV_SYSERR	0x800C0015	Error in handshake flags
CIFX_DEV_MAILBOX_FULL	0x800C0016	Send mailbox is full
CIFX_DEV_PUT_TIMEOUT	0x800C0017	Send packet timeout
CIFX_DEV_GET_TIMEOUT	0x800C0018	Receive packet timeout
CIFX_DEV_GET_NO_PACKET	0x800C0019	No packet available
CIFX_DEV_MAILBOX_TOO_SHORT	0x800C001A	Mailbox too short
CIFX_DEV_RESET_TIMEOUT	0x800C0020	Reset command timeout
CIFX_DEV_NO_COM_FLAG	0x800C0021	COM-flag not set
CIFX_DEV_EXCHANGE_FAILED	0x800C0022	IO data exchange failed
CIFX_DEV_EXCHANGE_TIMEOUT	0x800C0023	IO data exchange timeout
CIFX_DEV_COM_MODE_UNKNOWN	0x800C0024	Unknown IO exchange mode
CIFX_DEV_FUNCTION_FAILED	0x800C0025	Device function failed
CIFX_DEV_DPMSIZE_MISMATCH	0x800C0026	DPM size differs from configuration
CIFX_DEV_STATE_MODE_UNKNOWN	0x800C0027	Unknown state mode
CIFX_DEV_HW_PORT_IS_USED	0x800C0028	Device is still accessed
CIFX_DEV_CONFIG_LOCK_TIMEOUT	0x800C0029	Configuration locking timeout
CIFX_DEV_CONFIG_UNLOCK_TIMEOUT	0x800C002A	Configuration unlocking timeout
CIFX_DEV_HOST_STATE_SET_TIMEOUT	0x800C002B	Set HOST state timeout
CIFX_DEV_HOST_STATE_CLEAR_TIMEOUT	0x800C002C	Clear HOST state timeout
CIFX_DEV_INITIALIZATION_TIMEOUT	0x800C002D	Timeout during channel initialization
CIFX_DEV_BUS_STATE_ON_TIMEOUT	0x800C002E	Set Bus ON Timeout
CIFX_DEV_BUS_STATE_OFF_TIMEOUT	0x800C002F	Set Bus OFF Timeout
CIFX_DEV_MODULE_ALREADY_RUNNING	0x800C0040	Module already running
CIFX_DEV_MODULE_ALREADY_EXISTS	0x800C0041	Module already exists

Table 67: Generic Device Error Codes

10.6 Error Codes netX Driver

10.6.1 CIFS API Transport Error Codes

Error Code (Definition)	Value	Description
CIFS_TRANSPORT_SEND_TIMEOUT	0x800D0001	Time out while sending data
CIFS_TRANSPORT_RECV_TIMEOUT	0x800D0002	Time out waiting for incoming data
CIFS_TRANSPORT_CONNECT	0x800D0003	Unable to communicate to the device no answer
CIFS_TRANSPORT_ABORTED	0x800D0004	Transfer has been aborted due to keep alive timeout or interface detachment
CIFS_CONNECTOR_FUNCTIONS_READ_ERROR	0x800D0010	Error reading the connector functions from the DLL
CIFS_CONNECTOR_IDENTIFIER_TOO_LONG	0x800D0011	Connector delivers an identifier longer than 6 characters
CIFS_CONNECTOR_IDENTIFIER_EMPTY	0x800D0012	Connector delivers an empty identifier
CIFS_CONNECTOR_DUPLICATE_IDENTIFIER	0x800D0013	Connector identifier already used

Table 68: CIFS API Transport Error Codes

10.6.2 CIFS API Transport Header State Error Codes

Error Code (Definition)	Value	Description
CIFS_TRANSPORT_ERROR_UNKNOWN	0x800E0001	Unknown error code in transport header
CIFS_TRANSPORT_CHECKSUM_ERROR	0x800E0002	CRC16 checksum failed
CIFS_TRANSPORT_LENGTH_INCOMPLETE	0x800E0003	Transaction with incomplete length detected
CIFS_TRANSPORT_DATA_TYPE_UNKNOWN	0x800E0004	Device does not support requested data type
CIFS_TRANSPORT_DEVICE_UNKNOWN	0x800E0005	Device not available unknown
CIFS_TRANSPORT_CHANNEL_UNKNOWN	0x800E0006	Channel not available unknown
CIFS_TRANSPORT_SEQUENCE	0x800E0007	Sequence error detected
CIFS_TRANSPORT_BUFFER_OVERFLOW	0x800E0008	Buffer overflow detected
CIFS_TRANSPORT_RESOURCE	0x800E0009	Device signals out of resources
CIFS_TRANSPORT_KEEPA_LIVE	0x800E000A	Device connection monitoring error (Keep alive)
CIFS_TRANSPORT_DATA_TOO_SHORT	0x800E000B	Received transaction data too short

Table 69: CIFS API Transport Header State Error Codes

10.7 ODM Error Codes DBM V4

ODM Error Codes DBM V4		
Error Code (Definition)	Value	Description
CDBM_E_MD5_INVALID	0XC004C810	Checksum invalid
CDBM_E_INTERNALERROR	0XC004C811	Internal Error
CDBM_W_WRITEREGISTRY	0X8004C812	Error writing to the registry
CDBM_E_UNEXPECTED_VALUE_IN_OLD_HEADER_FORMAT	0XC004C813	Error in a file containing the old DBM Header format.
CDBM_E_CHECKSUM_INVALID	0XC004C814	The Checksum of the old Header is invalid
CDBM_E_DB_ALREADY_LOADED_FORMAT	0XC004C815	A database is already loaded
CDBM_E_NO_VALID_TRANSACTION	0XC004C816	No valid transaction handle given
CDBM_E_STD_STRUCT_ERROR	0XC004C817	An error occurred during validation of data
CDBM_E_UNSUPPORTED_DATA_TYPE_FORMAT	0XC004C818	Unsupported DataType
CDBM_W_CLASS_DELETED_FORMAT	0X8004C819 (Warning)	Using an Object which is marked as deleted
CDBM_W_CLIENT_DISCONNECTED	0X8004C81A (Warning)	A Client has already an outstanding connection to a Table. The connection is now destroyed.
CDBM_E_STRUCTURE_DEFINITION_INVALID	0XC004C81B	A structure definition of an Element in a Table is invalid
CDBM_E_NO_DATA_AVAILABLE	0XC004C81C	No data available for this operation
CDBM_E_NO_VALID_STRUCTURE	0XC004C81D	No valid structure available for this operation
CDBM_E_NO_TOGGLE_STRING_FOUND	0XC004C81E	No Toggle string found for this number
CDBM_E_ELEMENT_OUT_OF_RANGE	0XC004C81F	An element wasn't found in the Record of a Table
CDBM_E_ELEMENT_NOT_IN_TABLE	0XC004C820	The element is not part of the Table
CDBM_E_CANNOT_CONVERT_INTO_CLIENT_TYPE	0XC004C821	The data can't be converted into the Client type
CDBM_E_TRANSACTION_ALREADY_OPEN	0XC004C822	A transaction is already open. Please close this one first before opening a new one.
CDBM_I_OLD_WITHOUT_HEADER	0X4004C823 (Informational)	Use of an old DBM file Format without Header
CDBM_E_HR_FROM	0XC004C824	An HRESULT was received from a Subroutine
CDBM_E_PARAMETER	0XC004C825	A Parameter is invalid
CDBM_E_NOTIMPL	0XC004C826	Method is currently not implemented
CDBM_E_OUTOFMEMORY	0XC004C827	Out of memory
CDBM_E_NO_OPEN_TRANSACTION	0XC004C828	No transaction open
CDBM_E_NO_CONTENTS	0XC004C829	No contents available
CDBM_REC_NO_NOT_FOUND	0XC004C82A	Record not found
CDBM_STRUCTURE_ELEMENT_NOT_FOUND	0XC004C82B	Element of the Structure not found
CDBM_E_NO_MORE_RECORDS_IN_TABTYPE	0XC004C82C	Table type 3 can contain only one record
CDBM_E_WRITE	0XC004C82D	The data in the VARIANT must be given in a SafeArray
CDBM_E_WRITE_NO_PARRAY	0XC004C82E	The VARIANT contains no valid [parray] element

ODM Error Codes DBM V4		
Error Code (Definition)	Value	Description
CDBM_E_WRITE_CANT_ACCESS_DATA	0XC004C82F	Unable to access SafeArray Data in the VARIANT
CDBM_E_WRITE_DATA	0XC004C830	To write the data of this Element it must be given as a BSTR, or as an Array of VT_UI1/VT_I1
CDBM_E_WRITE_BSTR_E1	0XC004C831	The BSTR string must have an even length.
CDBM_E_WRITE_BSTR_E2	0XC004C832	The BSTR string must contain only hex digits (0..9 and a/A..f/F).
CDBM_E_WRITE_CANT_INTERPRET_ARRAY	0XC004C833	Unable to interpret data in the SafeArray.
CDBM_E_WRITE_VT_ERROR	0XC004C834	Data type in the SafeArray is not VT_UI1 or VT_I1.
CDBM_E_WRITE_LENGTH	0XC004C835	Data length is invalid for write operation of this type.
CDBM_WRITE_ELEMENT	0XC004C836	Element not found in the Record of the Table
CDBM_MIN_MAX_ERROR	0XC004C837	Can't write data because of min underflow or max overflow
CDBM_TABLE_EXIST	0XC004C838	Table already exist in the database
CDBM_MIN_MAX_INVALID	0XC004C839	The Min value is greater than the Max Value
CDBM_DEF_MIN_MAX_INVALID	0XC004C83A	The Default Value is not in the range between the Min value and the Max Value
CDBM_CANT_CHANGE_STRUCTURE_WHILE_RECORDS_EXIST	0XC004C83B	It's not allowed to change the structure while Records exist in the Table
CDBM_NEW_STRUCT_NEEDS_TYPE	0XC004C83C	In a newly added structure the data type must be set also
CDBM_VALUE_ERROR	0XC004C83D	Range error while validating a value
CDBM_DATATYPE_UNSUPPORTED_IN_RCS	0XC004C83E	The data type is unsupported in the RCS file format
CDBM_I_COUNT_OF_TABLES_EXCEEDS_RCS_RANGE	0X4004C83F (Informational)	The count of Tables exceeds the RCS range of Tables. This can cause problems if the file is downloaded to RCS Systems
CDBM_I_COUNT_OF_TABLES_EXCEEDS_OLDDBM_RANGE	0X4004C840 (Informational)	The count of Tables exceeds the DBM32.DLL range of Tables. This can cause problems if the file is used with older Tools using the DBM32.DLL
CDBM_UNSUPPORTED_DATATYPE_IN_RCS_MODE	0XC004C841	The Data type is not compatible with the old database format
CDBM_WRITE_UNSTRUCTURED_1	0XC004C842	The data of an unstructured record can only be written with the 'Write' Method not with 'WriteElement'.
CDBM_READ_UNSTRUCTURED_1	0XC004C843	The data of an unstructured record can only be read with the 'Read' Method not with 'ReadElement'
CDBM_WRITE_DATA_LENGTH_INVALID	0XC004C844	The given data length doesn't correspond with the expected data length.
CDBM_UNKNOWN_VIEW_MODE	0XC004C845	The View Mode is unknown.
CDBM_E_DIAG_TABLE	0XC004C846	It doesn't make much sense to add or delete records from a diagnostic table because those changes are never saved.
CDBM_E_ADR_STRING_ERROR	0XC004C847	The given Address string doesn't fit the required format of this type where all address bytes must be in the range between 0 and FF

ODM Error Codes DBM V4		
Error Code (Definition)	Value	Description
CDBM_ERROR_FROM_VAR_CHANGE_TYPE	0XC004C848	Function VariantChangeType return an error when trying to convert the Parameter
CDBM_E_MINERROR	0XC004C849	Error while comparing the Value with the lower range
CDBM_E_MAXERROR	0XC004C84A	Error while comparing the Value with the upper range
CDBM_E_RANGE_ERROR	0XC004C84B	Value out of Range
CDBM_E_TABLE_TYPE1	0XC004C84C	Table type 1 doesn't have a unique record length over all records
CDBM_E_TABLE_TYPE3_ADDREC	0XC004C84D	Table type 3 doesn't allow to insert more than one Record
CDBM_E_TABTYPE1	0XC004C84E	It's not allowed to insert more Records than structure definitions in Table Type 1
CDBM_E_TOGGLE_NOT_FOUND	0XC004C84F	Could not find the string for this value in the list of valid toggle strings
CDBM_E_TOGGLE_VALUE_IS_EMPTY_STRING	0XC004C850	The toggle string for this value is empty.
CDBM_VARIANT2BYTEARRAY_ERROR	0XC004C851	Error during conversion of Variant to byte array
CDBM_E_SET_ELEM_PROP_DEPENDENCY	0XC004C852	The Toggle Type needs also the additional string and the additional number entries in the Method
CDBM_E_TABTYPE1_REC_DOESNT_CORRESPOND_WITH_ELEMENT	0XC004C853	When reading the records of Table type 1 elementwise the record number must correspond with the element number
CDBM_TABTYPE1_NO_DATA_FOUND_FOR_RECORD	0XC004C854	When reading the records of Table type 1 and structure definitions are present it's assumed that for each structure element a corresponding record must exist
CDBM_E_TABTYPE1_WRITE_ELEMENT_NE_RECORD	0XC004C855	When writing the records of Table type 1 elementwise and structure definitions are present it's only allowed to write the corresponding element number in each record
CDBM_E_TABTYPE1_WRITE_ELEMENT_NOT_FOUND	0XC004C856	When writing the records of Table type 1 with an array and structure definitions are present it's assumed that a corresponding element number of this record exist
CDBM_I_TABLE_NAME_EXCEEDS_RCS_RANGE	0X4004C857 (Informational)	The Table name exceeds the maximum length of RCS compatible Table names
CDBM_W_CUT_STRING	0X8004C858 (Warning)	The string exceeds the maximum length and will be limited to the maximum length
CDBM_I_STRING_TOO_SHORT	0X4004C859 (Informational)	The string is below the minimum length. The minimum length will be reduced.
CDBM_I_STRING_TOO_LONG	0X4004C85A (Informational)	The string is exceeding the maximum. The maximum length will be extended.
CDBM_E_STRING_TOO_SHORT	0XC004C85B (Error)	The string is below the minimum length.
CDBM_E_STRING_TOO_LONG	0XC004C85C (Error)	The string is exceeding the maximum length
CDBM_E_WRONG_TYPE_FOR_WRITE	0XC004C85D	Writing on the Element type with the given Data type is not implemented
CDBM_E_NO_APPEND_IN_STRUCTURED_RECORDS	0XC004C85E	Method IDbmRecord::AppendData is not allowed for structured records

ODM Error Codes DBM V4		
Error Code (Definition)	Value	Description
CDBM_E_DATA_UNAVAILABLE	0XC004C85F	No data available
CDBM_E_CANT_CONVERT_INT	0XC004C860	Unable to convert the value into the Element type
CDBM_E_DBM_FILE_OVERFLOW	0XC004C861	You try to write a RCS like database which needs too much bytes
CDBM_E_PW_ERROR	0XC004C862	Password not correct
CDBM_E_FILELENGTH_CORRUPT	0XC004C863	The file length doesn't correspond to the length given in the Header.
CDBM_E_STRUCT_TYPE	0XC004C864	Error in the file.
CDBM_E_MD5SUM_INVALID	0XC004C865	MD5 sum invalid
CDBM_E_STRUCT_LENGTH	0XC004C866	Error in the expected and given structure length at a specific offset in the file.
CDBM_E_APPEND	0XC004C867	Append of data is only allowed if the Record contains only one data field and the field type will support this
CDBM_APPEND_NOT_SUPPORTED	0XC004C868	Append of Data not supported by this filed type
CDBM_DATA_TYPE_APPEND_ERROR	0XC004C869	Can't append Data of this type.
CDBM_E_UNSTRUCTURED_TABLE_DOESNT_SUPPORT_LENGTH	0XC004C86A	A Table without structure information doesn't support a record length
CDBM_E_DISABLED_WHILE_TRANSACTION_IS_OPEN	0XC004C86B	The Method is disabled while a transaction is open. Please close this one first and call the Method again.
CDBM_E_UNABLE_TO_CALL_READ_ON_LINKED_LIST	0XC004C86C	The Method is disabled on a LinkedList type. Please use the IRecordCollection on this type.
CDBM_E_ELEMENT_HAS_NO_SUBSTRUCTURE	0XC004C86D	An Element from a Table has no substructure
CDBM_STRUCT_ERROR_FROM_VAR_CHANGE_TYPE	0XC004C86E	Error from calling VariantChangeType
CDBM_E_FOREIGNKEY_DEF	0XC004C86F	The definition of a FOREIGNKEY must contain the name of the related Table in the description and this Table must exist at this time
CDBM_E_FOREIGNKEY_REF_TAB	0XC004C870	The description of a FOREIGNKEY must refer to a Table of type 'eDbmTableTypeLinkedList'
CDBM_E_KEY	0XC004C871	To create a Record Collection with a KEY it's necessary to have the data type KEY at the first position in all Records of the searched Table
CDBM_E_KEY_TABLE_TYPE	0XC004C872	This Method needs a Table of type 'eDbmTableTypeLinkedList'
CDBM_DATATYPE_NOT_IMPLEMENTED	0XC004C873	This data type is currently not implemented
CDBM_INSERT_POS_NOT_FOUND	0XC004C874	The position of the Record where the new one should be inserted wasn't found
CDBM_E_INSERT_REC_QI	0XC004C875	Error during insertion of a Record
CDBM_E_TAB_PROP	0XC004C876	Invalid Property in Table
CDBM_E_KEY_NOT_FOUND	0XC004C877	The KEY wasn't found in the Table
CDBM_E_KEY_INVALID	0XC004C878	The KEY is invalid for this operation

Table 70: ODM Error Codes DBM V4

11 Appendix

11.1 User Rights

User-rights are set within the FDT-container. Depending on the level the configuration is accessible by the user or read-only.

To access the **Settings**, **Configuration** and **Diagnosis** panes of the CANopen Master DTM you do not need special user rights. Also all users can select the decimal or hexadecimal Display mode or sort table entries.



Note: To edit, set or configure the parameters of the **Settings** and **Configuration** panes, you need user rights for *Maintenance*, for *Planning Engineer* or for *Administrator*.

The following tables give an overview of the user right groups and which user rights you need to configure the single parameters.

11.1.1 Settings

	Observer	Operator	Maintenanc e	Planning Engineer	Adminis- trator
<i>Driver</i>	D	D	X	X	X
<i>Verify or adapt Driver Settings</i>	-	-	X	X	X
<i>Configuring netX Driver</i>	-	-	X	X	X
<i>Device Assignment</i>	D	D	X	X	X
<i>Scanning for Devices</i>	-	-	X	X	X
<i>Selecting the Device (with or without firmware)</i>	-	-	X	X	X
<i>Selecting the Device once more (with Firmware)</i>	-	-	X	X	X
<i>Firmware Download</i>	D	D	X	X	X

Table 71: Settings (D = Displaying, X = Editing, Configuring)

11.1.2 Configuration

	Observer	Operator	Maintenance	Planning Engineer	Administrator
<i>Master Settings</i>	D	D	X	X	X
<i>Start of Bus Communication</i>	D	D	X	X	X
<i>Application Monitoring</i>	D	D	X	X	X
<i>Process Image Storage Format</i>	D	D	X	X	X
<i>Module Alignment</i>	D	D	X	X	X
<i>Process Data Handshake</i>	D	D	X	X	X
<i>Advanced</i>	D	D	X	X	X
<i>Device Status Offset</i>	D	D	X	X	X
<i>Bus Parameters</i>	D	D	X	X	X
<i>Device Description</i>	D	D	X	X	X
<i>Node Settings</i>	D	D	X	X	X
<i>Sync Master Settings</i>	D	D	X	X	X
<i>29 Bit COB-ID</i>	D	D	X	X	X
<i>Process Data</i>	D	D	X	X	X
<i>Address Table</i>	D	D	X	X	X
<i>Node ID Table</i>	D	D	X	X	X
<i>SDO Table</i>	D	D	X	X	X
<i>CAN-ID Table</i>	D	D	X	X	X
<i>Node BootUp Mapping Sequence</i>	D	D	X	X	X
<i>Monitoring</i>	D	D	X	X	X
<i>Node Guarding Protocol</i>	D	D	X	X	X
<i>Heartbeat Protocol</i>	D	D	X	X	X

Table 72: Configuration (D = Displaying, X = Editing, Configuring)

11.2 References

- [1] Device Type Manager (DTM) Style Guide, Version 1.0 ; FDT-JIG - Order No. <0001-0008-000>
- [2] CAN in Automation e.V., Erlangen: CANopen Application Layer and Communication Profile, CiA Draft Standard 301, Version 4.2.0, February 2011
- [3] CANopen Master Protocol API Manual, Revision 14, Hilscher GmbH 2013
- [4] CANopen Slave Protocol API Manual (V3), Revision 4, Hilscher GmbH 2013
- [5] CANdictionary, 6th edition, June 2011, CAN in Automation international users' and manufacturer's group e. V.

11.3 List of Figures

Figure 1: Dialog Structure of the CANopen Master DTM	14
Figure 2: Navigation Area	15
Figure 3: Status Bar – Status Fields 1 to 6	18
Figure 4: Status Bar Display Examples	18
Figure 5: Default Driver ‚cifX Device Driver’ for PC cards cifX	31
Figure 6: Default Driver ‚cifX Device Driver’ for PC cards cifX (example)	31
Figure 7: Default Driver ‚netX Driver’ for Hilscher devices except for PC cards cifX (example)	31
Figure 8: Manual Selection of multiple drivers (Example)	32
Figure 9: netX Driver > USB/RS232 Connection	35
Figure 10: netX Driver > TCP Connection	38
Figure 11: Device Assignment - detected Devices (* The name of the device class is displayed.) – Example for a device without firmware	40
Figure 12: Device Assignment - detected Devices (* The name of the device class is displayed.) Example for Devices without Firmware	42
Figure 13: Device Assignment - Selecting the Device (* The name of the device class is displayed.) – Example for a device without firmware / one Device is selected	43
Figure 14: Device Assignment - Selecting the Device (* The name of the device class is displayed.) – Example for Devices with and without Firmware / one Device is selected	44
Figure 15: Device Assignment - Selecting the Device (* The name of the device class is displayed.) – Example for a device with firmware / one Device is selected	45
Figure 16: Firmware Download	46
Figure 17: Error Message ‘The device is not assigned to the hardware!’	47
Figure 18: Window ‘Select Firmware File’ (Example CIFX)	48
Figure 19: Request Select Firmware File - Example Invalid Firmware	50
Figure 20: Request - Do you really want to download the firmware?	51
Figure 21: Firmware Download - Progress Bar	51
Figure 22: Clock Symbol and Hook Symbol green	52
Figure 23: Firmware Download – Download	52
Figure 24: Licensing	53
Figure 25: Configuration > Master Settings	57
Figure 26: Master Settings > Start of Bus Communication	58
Figure 27: Master Settings > Application Monitoring	58
Figure 28: Master Settings > Process Image Storage Format	59
Figure 29: Master Settings > Module Alignment	59
Figure 30: Master Settings > Process Data Handshake	60
Figure 31: Master Settings > Advanced	60
Figure 32: Master Settings > Device Status Offset	61
Figure 33: Bus Parameters (Under Device description the symbolic name of the device is displayed.)	62
Figure 34: Bus Parameters > Device Description	63
Figure 35: Bus Parameters > Node Settings	63
Figure 36: Bus Parameters > Sync Object	64
Figure 37: Bus Parameters > 29 Bit COB-ID	64
Figure 38: Diagram Acceptance Code / Acceptance Mask	65
Figure 39: Process Data (*The name of the Slave device is displayed.)	66
Figure 40: Configuration > Address Table (In the Figure shown here, in the column Device or Name example devices are displayed.)	67
Figure 41: Configuration > Address Table - Display Mode	68
Figure 42: Node ID Table (In the Figure shown here, in the column Device or Name example devices are displayed.)	69
Figure 43: SDO Table (In the Figure shown here, in the column Device or Name example devices are displayed.)	70
Figure 44: SDO Table > Display Mode	70

Figure 45: CAN-ID Table (In the Figure shown here, in the column Device or Name example devices are displayed.)	72
Figure 46: COB-ID-Table> Display Mode	72
Figure 47: Node BootUp Mapping Sequence	73
Figure 48: "Compare the configured Profile and Type Object 1000H with real value" enabled	73
Figure 49: Monitoring	75
Figure 50: Monitoring – Consumer (In the Figure shown here, in the column Device or Name example devices are displayed.)	76
Figure 51: Example Debug Mode with pending Diagnosis, Network View of a Project with one Network (Master and Slave)	82
Figure 52: Starting 'Network Scan' (Example)	86
Figure 53: Scan Response dialog of the Master DTM (Example)	87
Figure 54: Scan Response dialog of the Master DTM (Example)	88
Figure 55: DTM device is created (Example)	91
Figure 56: netDevice Message: Download	93
Figure 57: License Pane	96
Figure 58: License Pane - License Type	97
Figure 59: License Pane – License Type / Master protocols	97
Figure 60: License Pane – License Type / Utilities	97
Figure 61: License Pane - Request Form, please fill out / Device Information	100
Figure 62: License Pane - Request Form, please fill out / License Type	101
Figure 63: License Pane - Request Form, please fill out / Mandatory data	101
Figure 64: License Pane – Selecting the Subsidiary / Ordering / Contacts	102
Figure 65: License Pane – placing the order by E-mail	103
Figure 66: Example: Order E-Mail License request	103
Figure 67: License Pane - placing the order by Fax or by Telephone	104
Figure 68: Example: Order Data Form PrintOrderRequest	104
Figure 69: License Pane – Fax Number of the selected Subsidiary	105
Figure 70: License Pane – Telephone Number of the selected Subsidiary	105
Figure 71: License Pane - Ordering by exported File and E-Mail	106
Figure 72: License Pane - Download License	107
Figure 73: General Diagnosis	109
Figure 74: Master Diagnosis	111
Figure 75: Bus Diagnosis - Bus Monitoring	112
Figure 76: Station Diagnosis - Station Status	113
Figure 77: Firmware Diagnosis (*The name of the Firmware is displayed.)	114
Figure 78: Extended Diagnosis > [Folder Name] > Task Information Example Display	116
Figure 79: Extended Diagnosis > [Folder Name] > IniBatch Status Example Display	117
Figure 80: Extended Diagnosis > CAN_DL > Application Commands	118
Figure 81: Extended Diagnosis > CAN_DL > CAN Driver Status	119
Figure 82: Extended Diagnosis > CANOPEN_MASTER > Common diagnostic	120
Figure 83: Extended Diagnosis > CANOPEN_MASTER > CMS domain services	121
Figure 84: Extended Diagnosis > CANOPEN_MASTER > Node Initializations	122
Figure 85: Extended Diagnosis > CANOPEN_MASTER > Node state	122
Figure 86: Extended Diagnosis > CANOPEN_MASTER > Commands	123
Figure 87: Extended Diagnosis > CANOPEN_APM > Master configuration	125
Figure 88: Extended Diagnosis > CANOPEN_APM > Commands	126
Figure 89: Packet Monitor	128
Figure 90: Send > Packet Header and Send Data	129
Figure 91: Packet Header and Receive Data	130
Figure 92: Packet Description CANOPEN_MASTER_PACKET_SDO_REQ_T – SDO Upload Request	132
Figure 93: Example - Reading data via SDO-Upload – Send > Packet header	133
Figure 94: Example - Reading data via SDO-Upload –Send > Send Data	134

Figure 95: Packet Description CANOPEN_MASTER_PACKET_SDO_CNF_T – SDO Upload Confirmation	135
Figure 96: Example - Reading data via SDO-Upload – Receive > Packet header	136
Figure 97: Example - Reading data via SDO-Upload – Receive > Receive data	137
Figure 98: Packet Description CANOPEN_MASTER_PACKET_SDO_REQ_T – SDO Download Request	139
Figure 99: Example - Writing data via SDO-Download – Send > Packet header	140
Figure 100: Example - Writing data via SDO-Download – Send > Send Data	141
Figure 101: Packet Description CANOPEN_MASTER_PACKET_SDO_CNF_T – SDO Download Confirmation	142
Figure 102: Example - Writing data via SDO-Download – Receive > Packet header	143
Figure 103: Example – Writing data via SDO-Download – Receive > Receive data	144
Figure 104: IO Monitor	145
Figure 105: Window 'Process Image Monitor'	146

11.4 List of Tables

Table 1: Descriptions Dialog Panes	6
Table 2: List of Revisions	7
Table 3: General Device Information	15
Table 4: Overview Dialog Panes	16
Table 5: OK, Cancel, Apply and Help	17
Table 6: Selecting, inserting, deleting Table Line	17
Table 7: Status Bar Icons [1]	18
Table 8: Signal Words	22
Table 9: Getting started - Configuration Steps	26
Table 10: Descriptions of the Dialog Panes Settings	28
Table 11: Driver Selection List Parameters	31
Table 12: Parameters netX Driver > USB/RS232 Connection	36
Table 13: Parameters netX Driver > TCP Connection	39
Table 14: Parameters of the Device Assignment	41
Table 15: Parameter Firmware Download	46
Table 16: Parameters Select Firmware File	49
Table 17: Descriptions of the Dialog Panes Configuration	54
Table 18: Range of Value / Value for the Watchdog time	58
Table 19: Master Settings Pane Parameters - Process Image Storage Format	59
Table 20: Parameters Master Settings > Module Alignment	59
Table 21: Option Master Settings > Device Status Offset	61
Table 22: Bus Parameters > Node Settings	63
Table 23: Bus Parameters > Sync Object	64
Table 24: Process Data	66
Table 25: Parameters Address Table Pane - Inputs / Outputs	68
Table 26: Node ID Table	69
Table 27: SDO Table	70
Table 28: Mapping Method according to Specification DS301 V4	71
Table 29: CAN-ID Table	72
Table 30: Node BootUp Mapping Sequence	74
Table 31: Consumer Table	76
Table 32: Debug Mode Busline Colors and Debug Icons	83
Table 33: Description on the Scan Response dialog of the Master DTM	90
Table 34: Descriptions of the Diagnosis Panes	108
Table 35: Indication General Diagnosis	110
Table 36: Parameter General Diagnosis	110
Table 37: Parameter Master Diagnosis	111

Table 38: Bus Diagnosis - Bus Monitoring	112
Table 39: Possible Values for the Status	113
Table 40: Description Table Task Information	114
Table 41: Descriptions of the Dialog Panes Extended Diagnosis	115
Table 42: Extended Diagnosis > [Folder Name] > Task Information	116
Table 43: Extended Diagnosis > [Folder Name] > IniBatch Status	117
Table 44: Extended Diagnosis > CAN_DL > Application Commands	118
Table 45: Extended Diagnosis > CAN_DL > CAN Driver Status	119
Table 46: Extended Diagnosis > CANOPEN_MASTER > Common diagnostic	120
Table 47: Extended Diagnosis > CANOPEN_MASTER > CMS domain services	121
Table 48: Extended Diagnosis > CANOPEN_MASTER > Node Initializations	122
Table 49: Extended Diagnosis > CANOPEN_MASTER > Node state	122
Table 50: Extended Diagnosis > CANOPEN_MASTER > Commands	124
Table 51: Extended Diagnosis > CANOPEN_APM > Master configuration	125
Table 52: Extended Diagnosis > CANOPEN_APM > Commands	126
Table 53: Descriptions of the Diagnosis Panes	127
Table 54: Descriptions Packet Header	129
Table 55: Descriptions Packet Header	130
Table 56: Notes to the Window 'Process Image Monitor'	147
Table 57: Overview Error Codes and Ranges	149
Table 58: RCX General Task Errors	150
Table 59: RCX Common Status & Errors Codes	151
Table 60: RCX Status & Error Codes	152
Table 61: RCX Status & Error Codes Slave State	152
Table 62: ODM Error Codes - General ODM Error Codes	153
Table 63: ODM Error Codes - General ODM Driver Error Codes	154
Table 64: cifX Driver Specific ODM Error Codes	157
Table 65: Generic Error Codes	158
Table 66: Generic Driver Error Codes	159
Table 67: Generic Device Error Codes	160
Table 68: CIFS API Transport Error Codes	161
Table 69: CIFS API Transport Header State Error Codes	161
Table 70: ODM Error Codes DBM V4	165
Table 71: Settings (D = Displaying, X = Editing, Configuring)	166
Table 72: Configuration (D = Displaying, X = Editing, Configuring)	167

11.5 Glossary

CAN-ID

The CAN identifier is the main part of the arbitration field of a CAN data frame or CAN remote frame. It comprises 11 bit (base frame format) or 29 bit (extended frame format). The CAN identifier value determines implicitly the priority for the bus arbitration.

COB-ID

Communication Object Identifier

The COB-ID contains the CAN identifier and additional parameters for the related communication object. According to the CANopen specification ([2] page 131, Table 73) these are the „exists/not exists bit“, the „remote frame support bit“ (RTR allowed) and the „frame format 11 /29 bit“.

Data Packets

Data packets, i. e. messages are self-contained blocks of defined data length. The packets are used to communicate with the firmware and they are exchanged between the application (configuration software) and the firmware in the device. Using the configuration software feature *Packet Monitor* Packets can be sent once or cyclically to the connected device controlled by the user and packets received can be displayed.

Data packets comprise from a **Packet Header** and the **Send Data** or from a **Packet Header** and the **Receive Data**. The packet header can be evaluated by the receiver of the packet and contain the sender and receiver address, the data length, an ID number, status and error messages and the command or response code. The minimum packet size amounts 40 Byte for the packet header. The sending and receiving data are added.

DTM

Device Type Manager

The Device Type Manager (DTM) is a software module with graphical user interface for the configuration and/or for diagnosis of devices.

EDS

An Electronic Data Sheet (EDS) provides information necessary to access and alter the configurable parameters of a device. An Electronic Data Sheet (EDS) is an external file that contains information about configurable attributes for the device, including object addresses of each parameter. The application objects in a device represent the destination addresses for configuration data. These addresses are encoded in the EDS.

FDT

Field Device Tool

FDT specifies an interface, in order to be able to use DTM (Device Type Manager) in different applications of different manufacturers.

Node ID

The Node ID is the network address of the device. The network address serves to distinguish itself from other devices on the network. Therefore an unique address must be assigned to each device.

Master

Master devices initiate the data exchange at the bus.

ODMV3

The Online-Data-Manager Version 3 (ODMV3) is an application interface. The ODMV3 works as a server, which can be run as an out-proc server or system service. Its task is to provide different applications (e. g. SYCON.net), access to multiple devices and even share one device amongst several applications.

Slave

Slave devices are configured by the Master and perform then the communication.

11.6 Contacts

Headquarters

Germany

Hilscher Gesellschaft für
Systemautomation mbH
Rheinstrasse 15
65795 Hattersheim
Phone: +49 (0) 6190 9907-0
Fax: +49 (0) 6190 9907-50
E-Mail: info@hilscher.com

Support

Phone: +49 (0) 6190 9907-99
E-Mail: de.support@hilscher.com

Subsidiaries

China

Hilscher Systemautomation (Shanghai) Co. Ltd.
200010 Shanghai
Phone: +86 (0) 21-6355-5161
E-Mail: info@hilscher.cn

Support

Phone: +86 (0) 21-6355-5161
E-Mail: cn.support@hilscher.com

France

Hilscher France S.a.r.l.
69500 Bron
Phone: +33 (0) 4 72 37 98 40
E-Mail: info@hilscher.fr

Support

Phone: +33 (0) 4 72 37 98 40
E-Mail: fr.support@hilscher.com

India

Hilscher India Pvt. Ltd.
Pune, Delhi, Mumbai
Phone: +91 8888 750 777
E-Mail: info@hilscher.in

Italy

Hilscher Italia S.r.l.
20090 Vimodrone (MI)
Phone: +39 02 25007068
E-Mail: info@hilscher.it

Support

Phone: +39 02 25007068
E-Mail: it.support@hilscher.com

Japan

Hilscher Japan KK
Tokyo, 160-0022
Phone: +81 (0) 3-5362-0521
E-Mail: info@hilscher.jp

Support

Phone: +81 (0) 3-5362-0521
E-Mail: jp.support@hilscher.com

Korea

Hilscher Korea Inc.
Seongnam, Gyeonggi, 463-400
Phone: +82 (0) 31-789-3715
E-Mail: info@hilscher.kr

Switzerland

Hilscher Swiss GmbH
4500 Solothurn
Phone: +41 (0) 32 623 6633
E-Mail: info@hilscher.ch

Support

Phone: +49 (0) 6190 9907-99
E-Mail: ch.support@hilscher.com

USA

Hilscher North America, Inc.
Lisle, IL 60532
Phone: +1 630-505-5301
E-Mail: info@hilscher.us

Support

Phone: +1 630-505-5301
E-Mail: us.support@hilscher.com