Operating Instruction Manual

netX Configuration Tool for cifX, comX and netJACK

Configuration of Real Time Ethernet and Fieldbus Slaves

Hilscher Gesellschaft für Systemautomation mbH

www.hilscher.com

DOC110205OI06EN | Revision 6 | English | 2013-09 | Released | Public

Table of Contents

1 Introduction

1.1 About this Manual

This operating instruction manual contains descriptions on the configuration and diagnosis of

- PC Cards cifX,
- Communication Modules comX and
- Communication Modules netJACK

as follows:

- The user interface of the configuration program **netX Configuration Tool**.
- How to configure the device parameters of a Real-Time Ethernet or of a fieldbus Slave using **netX Configuration Tool**.
- How to make a diagnosis of the device communication using **netX Configuration Tool**.

For information about the installation, operation and hardware description of your device, please refer to the corresponding user manual of your device, which is enclosed on the DVD delivered.

1.1.1 Help

The **netX Configuration Tool** contains an integrated online help facility.

➢ To open the online help in **netX Configuration Tool**, click on the **Help** button or press the **F1** key.

1.1.2 List of Revisions

Index	Date	Version	Component	Chapter	Revision
6	13-09-16	1.05.x.x	netXSetup.exe	4.4.5	Windows 8 added. Section *CompoNet Slave Parameters* updated.

Table 1: List of Revisions

1.1.3 Conventions in this Manual

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

Notes

Important: <important note>

Note: <note>

<note, where to find further information>

Operation Instructions

1. <instruction>
2. <instruction>

or

➢ <instruction>

Results

<result>

1.2 Legal Notes

1.2.1 Copyright

1.2.2 Important Notes

The user manual, accompanying texts and the documentation were created for the use of the products by qualified experts, however, errors cannot be ruled out. For this reason, no guarantee can be made and neither juristic responsibility for erroneous information nor any liability can be assumed. Descriptions, accompanying texts and documentation included in the user manual do not present a guarantee nor any information about proper use as stipulated in the contract or a warranted feature. It cannot be ruled out that the user manual, the accompanying texts and the documentation do not correspond exactly to the described features, standards or other data of the delivered product. No warranty or guarantee regarding the correctness or accuracy of the information is assumed.

We reserve the right to change our products and their specification as well as related user manuals, accompanying texts and documentation at all times and without advance notice, without obligation to report the change. Changes will be included in future manuals and do not constitute any obligations. There is no entitlement to revisions of delivered documents. The manual delivered with the product applies.

Hilscher Gesellschaft für Systemautomation mbH is not liable under any circumstances for direct, indirect, incidental or follow-on damage or loss of earnings resulting from the use of the information contained in this publication.

1.2.3 Exclusion of Liability

The software was produced and tested with utmost care by Hilscher Gesellschaft für Systemautomation mbH and is made available as is. No warranty can be assumed for the performance and flawlessness of the software for all usage conditions and cases and for the results produced when utilized by the user. Liability for any damages that may result from the use of the hardware or software or related documents, is limited to cases of intent or grossly negligent violation of significant contractual obligations. Indemnity claims for the violation of significant contractual obligations are limited to damages that are foreseeable and typical for this type of contract.

It is strictly prohibited to use the software in the following areas:

- for military purposes or in weapon systems;
- for the design, construction, maintenance or operation of nuclear facilities;
- in air traffic control systems, air traffic or air traffic communication systems;
- in life support systems;
- in systems in which failures in the software could lead to personal injury or injuries leading to death.

We inform you that the software was not developed for use in dangerous environments requiring fail-proof control mechanisms. Use of the software in such an environment occurs at your own risk. No liability is assumed for damages or losses due to unauthorized use.

1.2.4 Warranty

Although the hardware and software was developed with utmost care and tested intensively, Hilscher Gesellschaft für Systemautomation mbH does not guarantee its suitability for any purpose not confirmed in writing. It cannot be guaranteed that the hardware and software will meet your requirements, that the use of the software operates without interruption and that the software is free of errors. No guarantee is made regarding infringements, violations of patents, rights of ownership or the freedom from interference by third parties. No additional guarantees or assurances are made regarding marketability, freedom of defect of title, integration or usability for certain purposes unless they are required in accordance with the law and cannot be limited. Warranty claims are limited to the right to claim rectification.

1.2.5 Export Regulations

The delivered product (including the technical data) is subject to export or import laws as well as the associated regulations of different counters, in particular those of Germany and the USA. The software may not be exported to countries where this is prohibited by the United States Export Administration Act and its additional provisions. You are obligated to comply with the regulations at your personal responsibility. We wish to inform you that you may require permission from state authorities to export, re-export or import the product.

1.2.6 Registered Trademarks

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

Pentium® is a registered trademark of Intel Corporation in the United States of America and further countries.

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

CC-Link is a registered trademark of Mitsubishi Electric Corporation, Tokyo, Japan.

CompoNet™, DeviceNet™ and EtherNet/IP™ are trademarks of ODVA (Open DeviceNet Vendor Association, Inc).

EtherCAT® is a registered trademark and a patented technology of Beckhoff Automation GmbH, Verl, Germany, formerly Elektro Beckhoff GmbH.

Modbus is a registered trademark of Schneider Electric.

POWERLINK is a registered trademark of B&R, Bernecker + Rainer Industrie-Elektronik Ges.m.b.H, Eggelsberg, Austria

PROFIBUS® und PROFINET® are registered trademarks of PROFIBUS International, Karlsruhe.

sercos and sercos interface are registered trademarks of sercos international e. V., Suessen, Germany.

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

1.2.6.1 EtherCAT Disclaimer

EtherCAT® is registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.

To get details and restrictions regarding using the EtherCAT technology refer to the following documents:

- “EtherCAT Marking rules”
- “EtherCAT Conformance Test Policy”
- “EtherCAT Vendor ID Policy”

These documents are available at the ETG homepage www.ethercat.org or directly over info@ethercat.org.

A summary over Vendor ID, Conformance test, Membership and Network Logo can be found within the appendix section of this document under section *EtherCAT Summary over Vendor ID, Conformance* test, Membership and Network Logo on page 177.

2 Installing the netX Configuration Tool

2.1 System Requirements

The system requirements listed below are valid for the Slave configuration tool **netX Configuration Tool**:

- PC with 586-, Pentium® processor or higher
- Operating system: Windows® XP SP3, Windows® Vista (32 bit) SP2, Windows® 7 (32 bit) or Windows® 7 (64 bit), Windows® 8 (32-Bit) or Windows® 8 (64-Bit)
- Administrator privilege required for installation
- Free space on hard disk: 50 MByte
- DVD ROM drive
- RAM: min. 256 MByte
- Graphics resolution: min 1024 x 768 pixels
- Keyboard and mouse

2.2 Requirements for the Installation

For information on the requirements and preconditions for the software installation of the configuration program **netX Configuration Tool**, please refer to the corresponding user manual of your device, which is enclosed on the DVD delivered.

2.3 Installing the netX Configuration Tool

➢ Start the netX Configuration Tool setup program.

Therefore:

➢ Close all application programs on the system!

➢ Insert the CD delivered with your device to the local CD ROM drive of the PC.

➢ Start in the directory **Software** the netX Configuration Tool setup program and execute the installation steps according to the instructions at the screen.

Or:

➢ Select **netX Configuration Tool** of the autostart menu.

➢ The **netX Configuration Tool** will be installed.

2.4 Uninstalling the netX Configuration Tool

➢ Select **Start > Control Panel > Software**

➢ Click **Remove** in the list beside the entry **netX Configuration Tool**.

➢ Answer the following question with **Yes**.

➢ The **netX Configuration Tool** will be removed.

3 Overview netX Configuration Tool

3.1 Overview Configuration Steps

The following table describes the main steps how to configure a netX based device, if the following requirements are fulfilled:

- The device hardware must be installed and operational.
- The **netX Configuration Tool** including the device driver must be installed.

For a detailed description of the sequence, as you must start and configure your device, please refer to the corresponding user manual of your device, which is enclosed on the DVD delivered.

No.	Step	Short Description	For detailed information see section	Page
1	Starting the netX Configuration Tool	➢ Select **Start > Programs > Hilscher GmbH > netX Configuration Tool**	*Working with netX Configuration Tool*	22
2	Selecting the Language	➢ Select in the **Select Language Icon Bar** the language icon for the desired language of the graphical user interface.	*Working with netX Configuration Tool*	22
3	Select Device	PCI Interface: ➢ Click on **cifX Device Driver >** [*Device Family*] (*Serial Number*) in the navigation area. USB or serial (RS32) connection: ➢ Click on **USB/RS232** > [*Device Family*] (*Serial Number*) in the navigation area.	*Navigation Area*	17
4	Selecting the Firmware Protocol	➢ Select in the **Select Network Icon Bar** the firmware button for the firmware (Slave device) you intend to use with your device. If all firmware symbols are greyed out: ➢ Make sure once more, the device is operational. ➢ Right click to the navigation area. ➢ Select the context menu **Reload**, to reestablish a connection to the device.	*Working with netX Configuration Tool* Reload	22 18
5	Setting the Parameters	➢ Click to **Configuration** in the navigation area. ➢ Set the configuration parameters for the Slave to be used. If you are not sure about the meaning of a single configuration parameter, we recommend to read the respective documentation or to choose the default value.	*Real-Time Ethernet and Fieldbus*	38
6	Or open existing configuration	➢ Open configuration template with existing configuration.	*Details on Configuration*	39
7	Downloading and save the Firmware and the Configuration	➢ Click to **Apply**. The firmware and the configuration are downloaded to the device. The configuration is saved to the hard disk.	*Working with netX Configuration Tool*	22

No.	Step	Short Description	For detailed information see section	Page
8	Starting the Communication and checking the Diagnostic Data	➢ Click to **Diagnostic** in the navigation area. ➢ Click to **Start**. The communication to the Master is started. ➢ Check the device communication with help of the displayed diagnostic data. ➢ Open the extended **Diagnostic** pane: Click **Extended >>**.	*'General' Dialog* *'Extended' Diagnosis*	74 77
9	IO Monitor	➢ Click to **IO Monitor** in the navigation area. ➢ Check the input or output data, ➢ close the IO Monitor dialog via OK.	*IO Monitor*	*176*
10	How to quit the netX Configuration Tool	➢ Click **OK** or **Cancel** to quit the **netX Configuration Tool**.	*Working with netX Configuration Tool*	22

Table 2: netX Configuration Tool Configuration Steps

3.2 Starting netX Configuration Tool

1. Make sure the device is correctly supplied with power and is operational.
2. Start the **netX Configuration Tool.**

➢ Select **Start > Programs > Hilscher GmbH > netX Configuration Tool.**

3.3 Introduction to the Dialog Structure

The graphical user interface of the **netX Configuration Tool** is composed of different areas and elements listed hereafter:

1. A header area containing the **Select Network and Language Bar** and the **Device Identification,**
2. The **Navigation Area**
   - with the folder **cifX Device Driver** or/and the folder **netX Transport DLL** and the device(s) connected (upper side) or
   - with the configuration menu buttons **Configuration**, **Diagnostic** and **IO Monitor** and depending on the device additional menu buttons (lower side),
3. The **Dialog Pane**,
4. The general buttons **OK**, **Cancel**, **Apply**, **Help**,
5. The **Status Bar** containing information e. g. the online-state of the **netX Configuration Tool**.

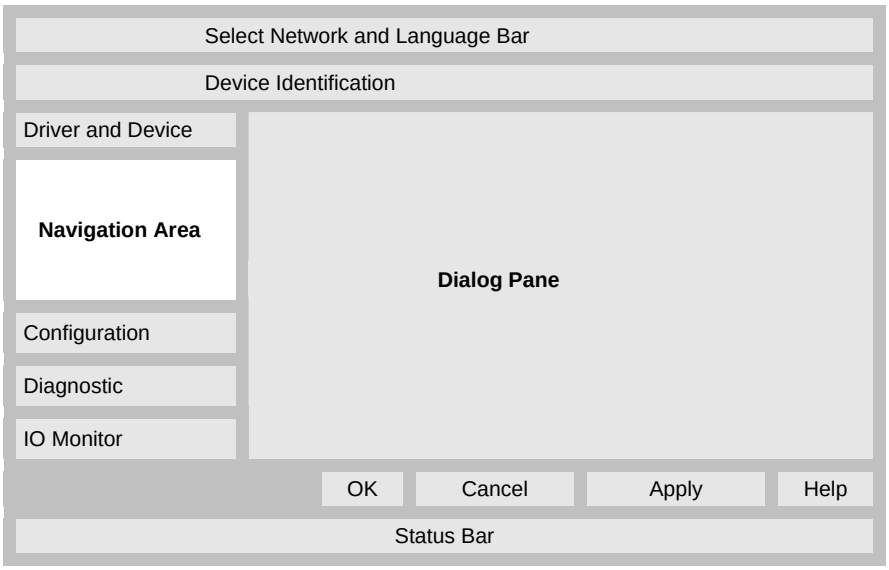


Figure 1: Dialog Structure of netX Configuration Tool

3.3.1 Select Network/Language Icon Bar and Device Identification

Select Network Icon Bar

Fieldbus device is connected (PROFIBUS MPI only for PC cards cifX)

Ethernet device is connected

Figure 2: Select Network Icon Bar (Example)

Language Icon Bar

Figure 3: Select Language Icon Bar (only English and German yet)

Device Identification

The **Device Identification** shows the information about the connected device.

Parameter	Meaning
IO Device	Name of the device
Vendor	Vendor name of the device
HW Device ID	Identification number of the hardware device
HW Vendor ID	Identification number of the hardware vendor
Firmware	Name of the currently loaded firmware
Version	Version of the currently loaded firmware

Table 3: Device Identification

3.3.2 Navigation Area

Drivers and Devices

At the upper side of the **Navigation Area** the connected devices are displayed depending on the used driver under the folder **cifX Device Driver** or **USB/RS232**:

- Under the **cifX Device Driver** folder the devices connected via PCI are displayed.

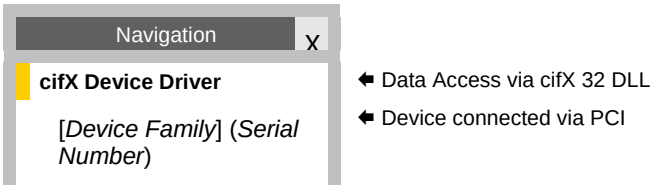


Figure 4: Drivers and Devices, Example PCI Interface

- Under the **USB/RS232** folder the devices connected via USB or serial (RS232) interface are displayed.

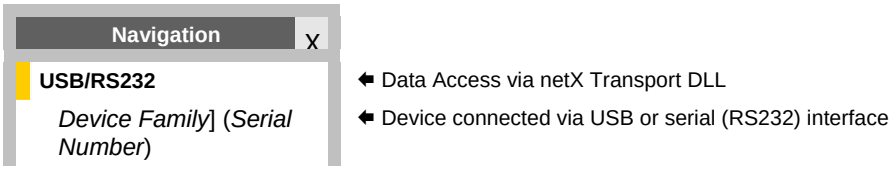


Figure 5: Drivers and Devices, Example USB or serial Connection

Configuration and Diagnosis

At the lower side of the **Navigation Area** the configuration menu buttons give access to the dialog panes **Configuration** and **Diagnostic** and depending on the device to additional panes.

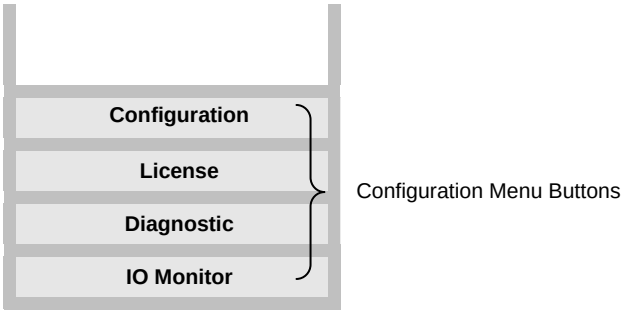


Figure 6: Configuration Menu Buttons

Note: The scope of functions of the **netX Configuration Tool** depends from the installation setup used. Therefore depending on the device the displayed navigation folders and menu buttons may differ from the example given here.

Reload

If the netX Configuration Tool is not connected to the device:

- Make sure once more, the device is operational.

Then reestablish a connection to the device:

- Right click to the navigation area.
- Select the context menu **Reload**.

Hide/display Navigation

The **Navigation Area** can be hidden or it can be displayed again.

Control	Meaning
	Window button to hide the navigation area, (at the right side of the navigation title bar).
	**Navigation** button to open the navigation area, (at the lower left corner of the dialog pane).

Table 4: Hide/display Navigation

3.3.3 Dialog Pane

At the dialog pane the panes **Configuration** and **Diagnostic** and depending on the device further panes can be displayed.

Configuration	
Real-Time Ethernet and Fieldbus	
	In the **Configuration** pane the parameters of the currently loaded firmware for Real-Time Ethernet systems and fieldbus systems are displayed and can be edited. For further information see section *Real-Time Ethernet and Fieldbus* on page 38.
License	
	In the **License** pane you can order and download licenses for Master protocols or utilities. For further information see section *License* on page 25.
Diagnostic, IO Monitor	
	In the **Diagnostic** pane diagnosis information can be displayed. Via **Start** and **Stop** the communication to the Master device can be started or stopped. For further information see section *'General' Dialog* on page *74* and section *'Extended' Diagnosis* on page 77.
	Also the **IO Monitor** is provided for test and diagnosis purposes. For further information, refer to section *IO Monitor* on page 176.

Table 5: Overview Dialog Panes

3.3.4 General Buttons

The table below explains the general buttons and controls in the **netX Configuration Tool** user interface.

Button	Meaning
OK	If the firmware or the configuration has not been changed and you click to the **OK** button, the **netX Configuration Tool** is closed.
	If the firmware or the configuration has been changed and you click to the **OK** button, the request is displayed **Do you want to download the new firmware and the new configuration on the device?** ➢ Select the **Yes** button, if you want to download the firmware/configuration directly to the device and then quit the **netX Configuration Tool.** The configuration is saved to the hard disk of the PC. Or ➢ Select the **No** button, if you want to quit the **netX Configuration Tool.** Or ➢ Select the **Cancel** button, if you want to cancel the procedure and to return to the main pane.
Cancel	If the firmware or the configuration has been changed and you click to the **Cancel** button, the request is displayed **The configuration was changed. Do you want to save it before closing the application?** ➢ Select the **Yes** button, if you want to save the configuration. The configuration will be saved to the hard disk of the PC but it is not downloaded to the device. Or ➢ Select the **No** button, if you want to quit the **netX Configuration Tool.**
Apply	If the firmware or the configuration has not been changed and the offline and the online configuration are identical the **Apply** button is grayed out.
	If the firmware or the configuration has been changed and you click to the **Apply** button, the request is displayed **Do you want to download the new firmware and the new configuration on the device?** ➢ Select the **Yes** button, if you want to download the firmware/configuration directly to the device. The configuration is saved to the hard disk of the PC. Or ➢ Select the **Cancel** button, if you want to cancel the procedure and to return to the main pane.
Help	To open the **netX Configuration Tool** online help, click on the **Help** button.
Default	To reset the parameters to the default state, click the **Default** button in the configuration pane.
Enable	If 'Enabled' is unchecked, the default value is used.

Table 6: General Buttons and Controls

3.3.5 Status Bar

The **Status Bar** displays information about the current state of the **netX Configuration Tool.** The current activity, e.g. download, is signaled graphically via icons in the status bar. Furthermore the status message and the tool version are displayed here.

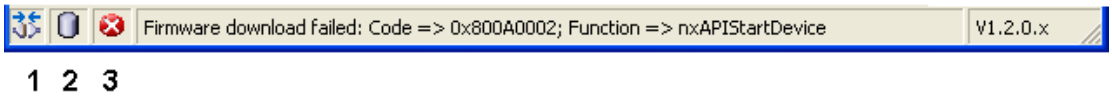


Figure 7: Status Bar: Status Fields 1 to 3, Status Messages, Version

Status Field	Icon/Meaning	
1	**DTM Connection States**	
		Connecting: Icon going closed = Device is going online
		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).
		Valid Modified: Parameter is changed (not equal to data source).
3	**Device Diagnosis Status**	
		Save operation succeeded: The save operation has been successful.
		Further messages due to successful handling of device data.
		Off-specification: The device is operating outside its specified range or internal diagnosis indicates deviations from measured or set values due to internal problems in the device or process characteristics.
		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

3.4 Working with netX Configuration Tool

To work with the **netX Configuration Tool** follow the steps described hereafter:

1. Make sure the device is correctly supplied with power and is operational.
2. Select **Start > Programs > Hilscher GmbH > netX Configuration Tool**.

➢ The **netX Configuration Tool** is displayed.

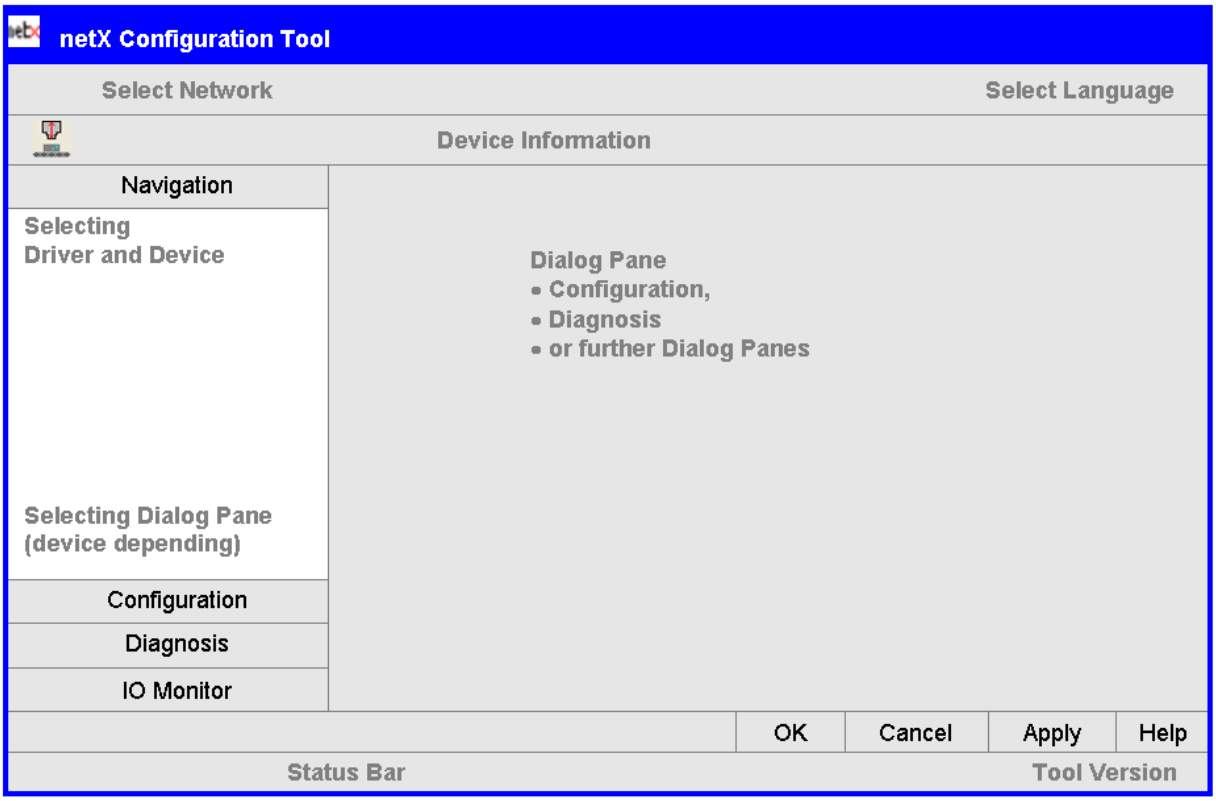


Figure 8: User Interface netX Configuration Tool

3. If necessary reestablish a connection to the device.

If the **Select Network** bar is disabled (all buttons greyed out), the **netX Configuration Tool** is not connected to the device.

➢ In this case make sure once more, the device is operational.

➢ Reestablish a connection to the device via **Navigation >** Right click > **Reload**.

4. Select the Language (currently English and German).

The **netX Configuration Tool** is displayed in the corresponding language.

5. Select Driver and Device.

Depending by the driver or by the device under **Navigation** the following folders are displayed:

- **cifX Device Driver** with the subfolder for the device connected via PCI Interface or/and
- **USB/RS232** with the subfolder for the device connected via USB or

serial interface.

➢ Click to **Navigation > cifX Device Driver** > or **USB/RS232** > [*Device Family*] (*Serial Number*).

➢ Depending by the device the network and the device are requested via the PCI, USB or serial interface.

➢ If a connected fieldbus device is selected, the **Select Network** (Fieldbus) is enabled.

➢ If a connected Real-Time Ethernet device is selected, the **Select Network** (Real-Time Ethernet) is enabled.

➢ The **Device Identification** shows the information about the connected device selected.

➢ **Navigation > Configuration**, **License**, **Diagnostic** and **IO Monitor** are enabled.

6. Select Network (Firmware Protocol).

➢ Select the Slave device firmware required for your device.

Network	Fieldbus
PROFIBUS	PROFIBUS DP-Slave
PROFIBUS MPI	PROFIBUS-MPI (only PC cards cifX)
CANopen	CANopen Slave
DeviceNet	DeviceNet Slave
CompoNet	CompoNet Slave
CC-Link	CC-Link Slave

Network	Real-Time Ethernet
PROFINET	PROFINET IO-Device
EtherCAT	EtherCAT-Slave
EtherNet/IP	EtherNet/IP-Adapter (Slave)
Modbus-IDA	Open Modbus/TCP
SERCOS interface	sercos Slave
POWERLINK	Powerlink Controlled Node/Slave
VARAN	VARAN Client (Slave)

Table 8: Select Network (Firmware Protocol)

7. Set Parameters.

➢ Click **Navigation >Configuration**.

➢ Set the parameters in the **Configuration** pane.

8. Download the Firmware and the Configuration.

➢ Click **Apply**.

➢ The selected firmware and the configuration are directly downloaded to the device. The configuration is saved to the hard disk of the PC. The download is signaled:

- In the status bar successively the messages are displayed: „Firmware download started“, „Configuration download started“, „Configuration download succeeded“.
- The corresponding progress bar is displayed.

9. Starting the Communication and checking the Diagnostic Data

- Click **Navigation > Diagnostic**.
- The **Diagnostic** is displayed.
- Click **Start** to start the communication to the Master device.
- The diagnostic information is updated online.
- Check the device communication with help of the displayed diagnostic data.
- Click **Extended >>**, to open the **Extended Diagnosis**.

10. Quit the **netX Configuration Tool**.

- Click **OK** or **Cancel**.

For further information refer to the chapter *Configuring Slave Devices using netX Configuration Tool* on page 38, to the section *General Buttons* on page 20 and to chapter Diagnostic on page 74 and to section *'Extended' Diagnosis* on page 77.

3.5 License

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

3.5.1 Open License Dialog

First open the **License** pane.

- Click in the navigation area on the **License** button to open the **License** pane.
- For further steps continue in the **License** pane.

3.5.2 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.

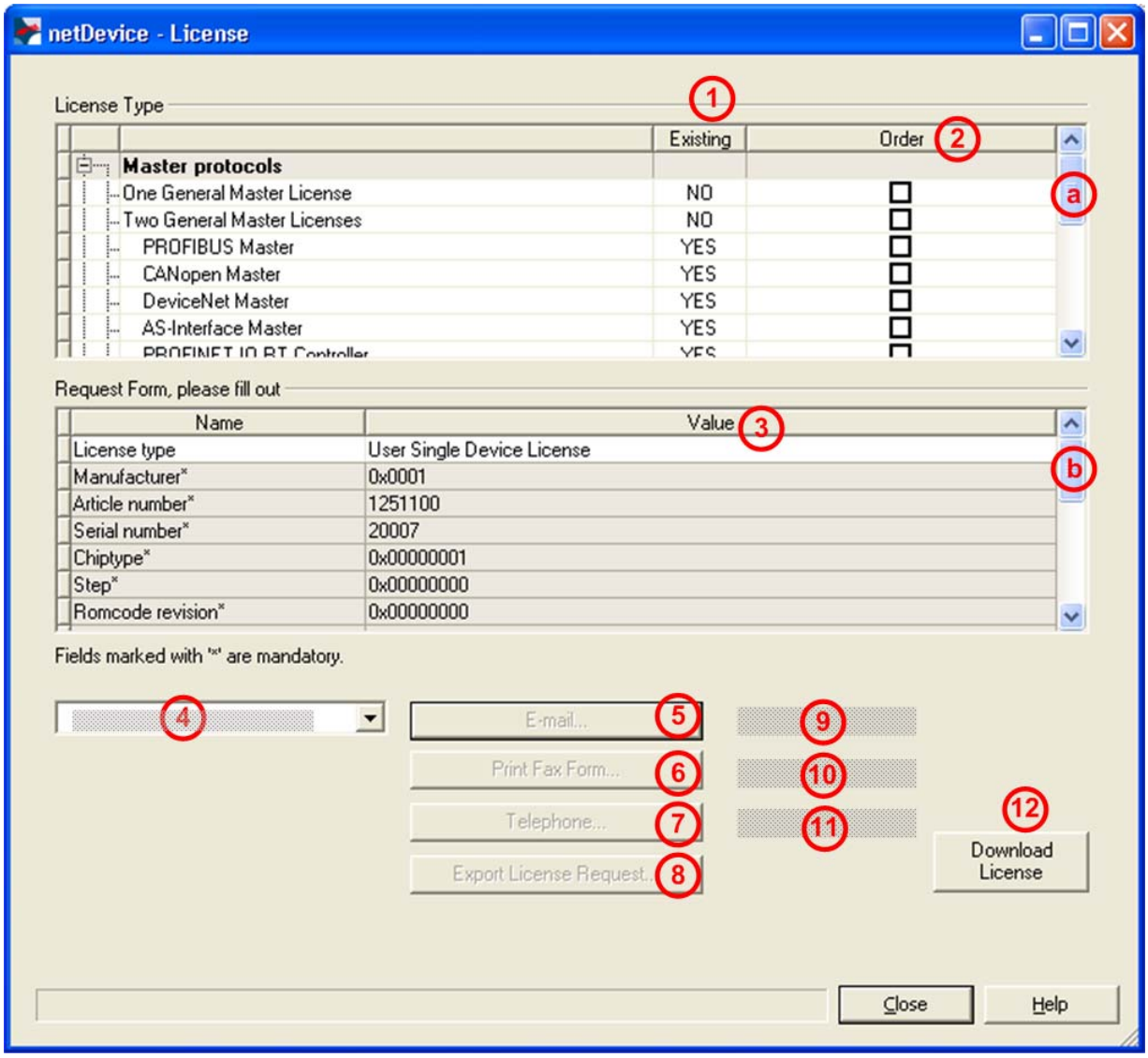


Figure 9: License Pane

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

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

3.5.3 Which Licenses are present in the Device?

Check, which licenses are present in the device.

How to proceed:

➢ Open the **License** pane as described under section *Open License Dialog* on page 25.

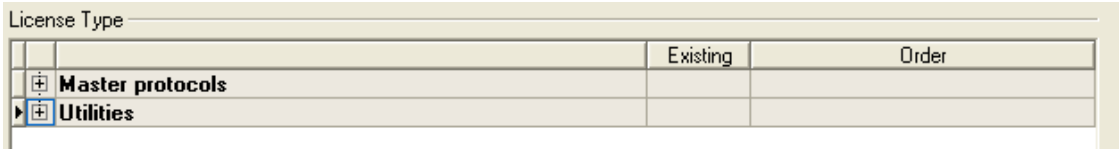


Figure 10: License Pane - License Type

➢ Under **License Type** click ⊞ at **Master protocols**.

➢ The **Master protocols** overview opens:

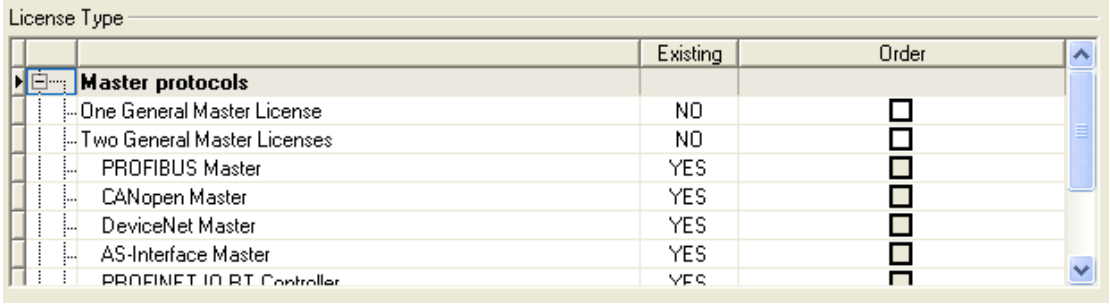


Figure 11: License Pane – License Type / Master protocols

➢ Or click ⊞ at **Utilities**.

➢ The **Utilities** overview opens:

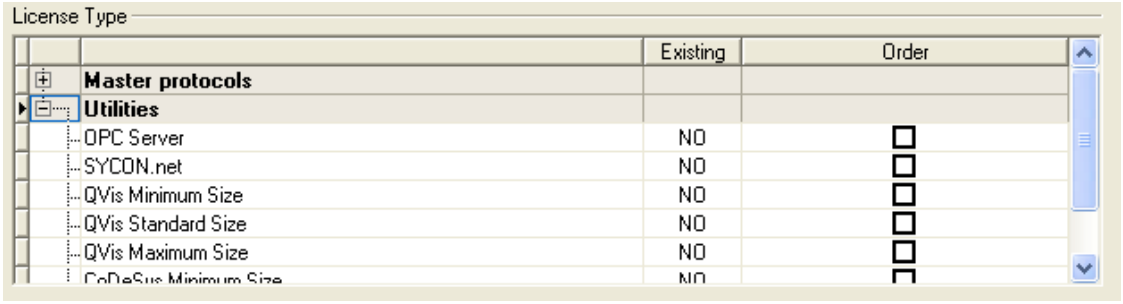


Figure 12: License Pane – License Type / Utilities

➢ 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.

3.5.3.1 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

3.5.3.2 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*.

3.5.4 How to order a License

To order a license, proceed as follows:

	Refer to Section:	*Page*
1. Open the license dialog.	*Open License Dialog*	25
2. Select the required licenses.	*Selecting License*	29
3. Enter the ordering data.	*Ordering Data*	30
4. Place your order.	*Ordering the License*	32

3.5.5 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.

3.5.6 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** (therefore refer to section *Data to manage the Order (License Information)*on page 31).

3.5.6.1 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:

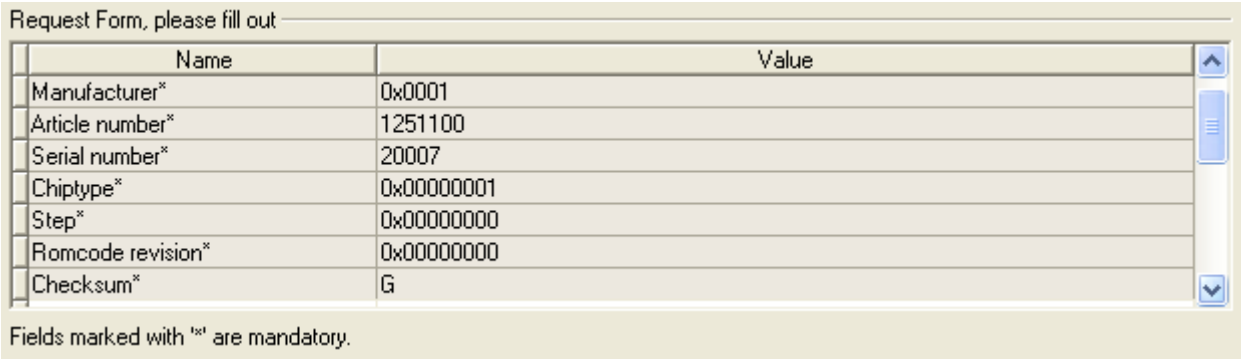


Figure 13: License Pane - Request Form, please fill out / Device Information

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

3.5.6.2 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).

Figure 14: 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

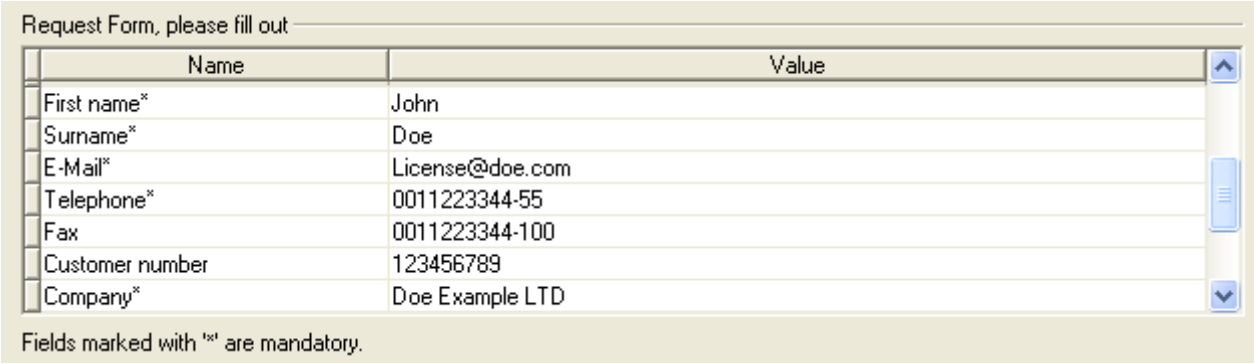


Figure 15: License Pane - Request Form, please fill out / Mandatory data

- 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.

3.5.7 Ordering the License

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

Figure 16: 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),	*Ordering the License by E Mail*	33
• or by **Fax** (6) or by **Telephone** (7),	*Ordering the License by Fax or by Telephone*	34
• or in a **File** (8).	*Exporting License Request to a File*	36

➪ The **Contact Data** of the selected subsidiary are displayed under Position (9), (10) and (11).

3.5.7.1 Ordering the License by E Mail

You can place your order by e-mail.

Figure 17: License Pane – placing the order by E-mail

➢ Click **E-mail…** (5).

➢ The order E-mail **License request** opens:

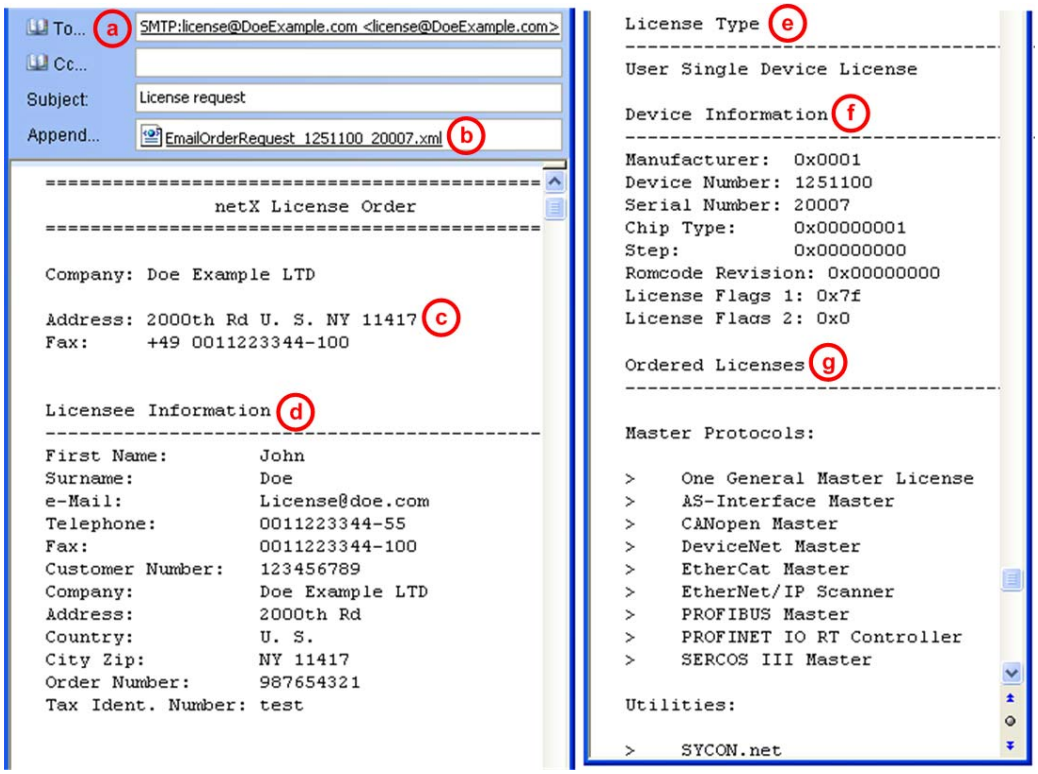


*Figure 18: Example: Order E-Mail **License request***

➢ The order e-mail **License request** contains:

- the **E-mail…** of the selected subsidiary (a),
- the automatically generated **XML file** (b) *EmailOrderRequest_-[Devicenumber]_[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.

3.5.7.2 Ordering the License by Fax or by Telephone

You can place your order by Fax or by Telephone.

Figure 19: 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_[Devicenumber]_[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 (c)
NY 11417
U. S.
fax: +11223344-100

Licensee Information (d)

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

License Type (e)

User Single Device License

Device Information (f)

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

Ordered Licenses (g)

Master Protocols
- One General Master License
- Sercos III Master

Utilities
- SYCON.net

Date: _____ Signature: _____

Figure 20: Example: Order Data Form PrintOrderRequest

- The order data form contains:
- the **Order Address** ⓒ,
- the **License Information** ⓓ,
- the **License Type** ⓔ,
- the **Device Data** ⓕ,
- the **ordered Licenses** ⓖ.

➢ Print the order data form, sign it and send it by Fax.

Figure 21: 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 22: 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.

3.5.7.3 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 23: 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_-[Devicenumber]_[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* on page 26).
- The order process is complete.

3.5.8 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** ⑫ in the **License** pane in the configuration software.

Figure 24: 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?* on page 27.

4 Configuring Slave Devices using netX Configuration Tool

4.1 Real-Time Ethernet and Fieldbus

The current parameters of the selected firmware protocol are displayed in the **Configuration** pane. The configuration parameters can be edited.

- Click in the navigation area to the **Configuration** button to open the **Configuration** pane.

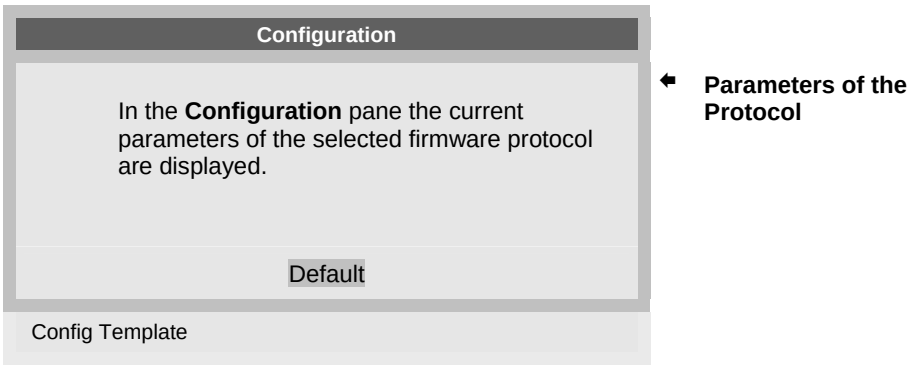


Figure 25: Configuration Pane

The window **Configuration** is described separately for each protocol. Therefore refer to the overview in the section *Overview Real-Time Ethernet and Fieldbus* on page 41.

4.1.1 Details on Configuration

Default

With the **Default** button the parameters can be reset to the default state.

Enable

If 'Enabled' is unchecked, the default value is used.

Error during Data Input

Error	Action
Parameter validation error! Input field cannot be empty.	Enter data.
Input validation error! Input field accepts only digits.	Enter only digits.
Input validation error! Input field accepts only digits and letters from A to F.	Enter only digits and letters from A to F.

Table 9: Error during Data Input

Configuration Template

Create new Configuration Template:

- In the field **Config Template** enter a name for the new configuration template.
- Select .

The settings in the **Configuration** window are saved in the new configuration template in a template XML file.

The XML files are *cifX_Templates.xml*, *netIC_Templates.xml* or *nxstk_Templates.xml* on the PC in the directory *Application Data* \ *Hilscher GmbH* \ *netX Configuration Tool* \ *Project*.

Open Configuration Template:

To access to the configuration settings stored in the configuration template the appropriate configuration template must be opened.

- Select the required configuration template via .

All settings stored in the configuration template are displayed in the window **Configuration** and can be applied.

Change Configuration Template:

- Open the required configuration template via  .
- Adjust the settings in the **configuration** window.
- Select  .

The changes of the configuration template are saved.

Delete Configuration Template:

- Open the configuration template to be deleted via  .
- Select  .

The configuration template is deleted.

Field / Button	Meaning
	Field to enter the name for the template configuration
	Saving the configuration template.
	Selecting a configuration template.
	Deleting a configuration template.

Table 10: Config Template

4.2 Overview Real-Time Ethernet and Fieldbus

Any **Parameters of the Protocol** are described in the subsections listed in the following table.

Section	Subsection	Page
Configuration Parameters Real-Time Ethernet Systems	*EtherCAT Slave Parameters*	*42*
	EtherNet/IP Adapter Parameters	*44*
	Open Modbus/TCP Parameters	*47*
	POWERLINK Controlled Node/Slave Parameters	*49*
	PROFINET IO-Device Parameters	*51*
	sercos Slave Parameters	*55*
	VARAN-Client (Slave) Parameters	59
Configuration Parameters Fieldbus Systems	*PROFIBUS DP Slave Parameters*	*62*
	PROFIBUS-MPI	64
	CANopen Slave Parameters	*66*
	DeviceNet Slave Parameters	*68*
	CompoNet Slave Parameters	*70*
	CC-Link Slave Parameters	*72*

Table 11: Descriptions Configuration Parameters Fieldbus- and Real-Time Fieldbus Systems

4.3 Configuration Parameters Real-Time Ethernet Systems

4.3.1 EtherCAT Slave Parameters

Parameter	Meaning	Range of Value/Value
Interface		
Bus Startup	Communication start application controlled or automatic	Application controlled, Automatic Default: Automatic
Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to 0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Off
I/O Data Status	Status of the input or the output data. For each input and output data the following status information (in Byte) is memorized in the dual-port memory: Status 0 = None (default) Status 1 = 1 Byte (for future use) Status 2 = 4 Byte (for future use)	None, (1 Byte, 4 Byte) Default: None
Ident		
Enabled	If 'Enabled' is unchecked, for the single Ident parameters each the respective default value is used.	
Vendor ID	Identification number of the manufacturer	0x00000000 ... 0xFFFFFFFF (hex) Primary Hilscher Vendor ID: 0x00000044 (hex) Default: Secondary Hilscher Vendor ID: 0xE0000044 (hex)
Product Code	Product code of the device as specified by the manufacturer	0x00000000 ... 0xFFFFFFFF (hex), Default: CIFX RE/ECS: 0x00000001 (hex), COMX 100XX-RE/ECS: 0x00000003 (hex), NJ 50X-RE/ECS: 0x00000021 (hex), NJ 100XX-RE/ECS: 0x00000022 (hex)
Revision Number	Revision number of the device as specified by the manufacturer	0x00000000 ... 0xFFFFFFFF (hex), Default: CIFX RE/ECS: 0x00000002 (hex), COMX 100XX-RE/ECS: 0x00000002 (hex), NJ 50X-RE/ECS: 0x00010000 (hex), NJ 100XX-RE/ECS: 0x00000002 (hex)
Serial Number	Serial number of the device	0x00000000 ... 0xFFFFFFFF (hex)
Data		

Parameter	Meaning	Range of Value/Value
Input Data Bytes	Length of the input data in Byte	CIFX RE/ECS, COMX 100XX-RE/ECS, NJ 100XX-RE/ECS: 0 … 256 (… 512*) Byte Default: 200 Byte
		NJ 50X-RE/ECS: 0 … 1024 Byte Default: 200 Byte
Output Data Bytes	Length of the output data in Byte	CIFX RE/ECS, COMX 100XX-RE/ECS, NJ 100XX-RE/ECS: 0 … 256 (… 512*) Byte Default: 200 Byte
		NJ 50X-RE/ECS: 0 … 1024 Byte Default: 200 Byte
* Sum: The maximum length of input and of output is equal 512 Bytes,		

Table 12: EtherCAT Slave Parameters

Note: To configure the Master, an XML file (device description file) is required. The settings in the used Master must comply with the settings in the Slave to establish communication. Important parameters are: Vendor ID, Product Code, Serial Number, Revision Number, Output and Input Data Bytes.

Note: If the XML file *Hilscher cifX RE ECS V2.**2**.x.xml* is used/updated, the firmware with the version *2.**2**.x* must be used/updated.

For CIFX RE/ECS, COMX 100XX-RE/ECS and NJ 100XX-RE/ECS please note also:

Note: *The loadable firmware supports for the number of cyclic input data and for cyclic output data in total up to 512 bytes. If more than 256 bytes for input data or for output data should be exchanged via EtharCAT, then a customer specific XML file is necessary. Additionally the following formular applies: (number of input bytes + 3)/4 + (number of output bytes + 3)/4 must be less or equal to 128.

4.3.2 EtherNet/IP Adapter Parameters

Parameter	Meaning	Range of Value/Value
Interface		
Bus Startup	Communication start application controlled or automatic	Application controlled, Automatic Default: Automatic
Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to 0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Off
I/O Data Status	Status of the input or the output data. For each input and output data the following status information (in Byte) is memorized in the dual-port memory: Status 0 = None (default) Status 1 = 1 Byte (for future use) Status 2 = 4 Byte (for future use)	None, (1 Byte, 4 Byte) Default: None
Ident		
Enabled	If ‘Enabled’ is unchecked, for the single Ident parameters each the respective default value is used.	
Vendor ID	Identification number of the manufacturer	0x00000000 ... 0x0000FFFF (hex), Hilscher: 0x00000011B (hex)
Product Code	Product code of the device as specified by the manufacturer	0x00000000 ... 0x0000FFFF (hex), Default CIFX RE/EIS: 0x00000101 (hex), COMX 100XX-RE/EIS: 0x00000103 (hex), NJ 50X-RE/EIS, 0x00000116 (hex), NJ 100XX-RE/EIS: 0x00000117 (hex)
Product Type	Communication Adapter	0x00000000 ... 0x0000FFFF (hex), Default: 0x00000000C (hex)
Major Rev	Major Revision	0 ... 255, Default: 1
Minor Rev	Minor Revision	0 ... 255, Default: 1
Device name	Device name of the device station as character string, e. g. EtherNet/IP Adapter (Slave).	0 - 31 ASCII characters, Default: Generic Device, Examples: CIFX RE/EIS, COMX 100XX-RE/EIS, NJ 50X-RE/EIS, NJ 100XX-RE/EIS
Bus		
IP Address	Valid IP Address for the device If ‘Enabled’ is unchecked (Default setting), the device obtains its IP Address from a DHCP server or also from a BOOTP server, if this one is checked. If ‘Enabled’ is checked, the device uses the manually entered value.	Valid IP address Default: unchecked

Parameter	Meaning	Range of Value/Value
Netmask	Valid Network mask for the device If 'Enabled' is unchecked (Default setting), the device obtains its Netmask from a DHCP server or also from a BOOTP server, if this one is checked. If 'Enabled' is checked, the device uses the manually entered value.	Valid network mask Default: unchecked
Gateway	Valid Gateway Address for the device If 'Enabled' is unchecked (Default setting), the device obtains its Gateway Address from a DHCP server or also from a BOOTP server, if this one is checked. If 'Enabled' is checked, the device uses the manually entered value.	Valid gateway address Default: unchecked
	There are three methods available, how the device can obtain its IP Address, Netmask and Gateway Address, one of which must be selected. These methods can also be combined. The device performs the following sequence in order to obtain the addresses: 1. from a DHCP server if DHCP is checked (if a DHCP server provides the requested addresses to the device, then the device uses these addresses) 2. from a BootP server if BootP is checked (if a BootP server provides the requested addresses to the device, then the device uses these addresses) 3. the addresses manually set are used. If the IP Address is set manually also the Network Mask must be set manually. The manually set Gateway Address is optional. If no DHCP server and no BootP server and no manually set addresses exist, then the protocol is not ready for initialization or for operation.	
Extras	BootP: If checked, the device obtains its IP Address, Netmask, Gateway Address from a BOOTP server.	Default: unchecked
	DHCP: If checked, the device obtains its IP Address, Netmask, Gateway Address from a DHCP server.	Default: checked
	100Mbit: Speed Selection, If checked, the device will operate at 100 MBit/s, else at 10 MBit/s. This parameter will not be in effect, when auto-negotiation is active.	Default: unchecked
	FullDuplex: Duplex Operation, If checked, full-duplex operation will be used. The device will operate in half-duplex mode, if this parameter is set to zero. This parameter will not be in effect, when auto-negotiation is active.	Default: unchecked
	Auto-neg.: Auto-Negotiation, If checked, the device will auto-negotiate link parameters with the remote hub or switch.	Default: checked
Data		
Prod. Data Length	Maximum allowed length of the input data in Byte. This parameter should be equal to or higher than the complete projected input data length, otherwise the EtherNet/IP device will reject the cyclic communication requests.	0 … 504 Byte Default: 32 Byte
Cons. Data Length	Maximum allowed length of the output data in Byte. This parameter should be equal to or higher than the complete projected output data length, otherwise the EtherNet/IP device will reject the cyclic communication requests.	0 … 504 Byte Default: 32 Byte

Table 13: EtherNet/IP Adapter Parameters

Note: To configure the Scanner/Master, an EDS file (device description file) is required. The settings in the used Scanner/Master must comply with the settings in the Adapter/Slave to establish communication. Important parameters are: Input, Output Data Bytes, Vendor ID, Product Type, Product Code, Major Rev, Minor Rev, IP Address and Netmask.

Note: To configure the connection type netX Configuration Tool uses as default the "32-bit Run/Idle header" setting for the "RT Transfer format" parameter.

4.3.3 Open Modbus/TCP Parameters

Parameter	Meaning	Range of Value/Value
Interface		
Bus Startup	Communication start application controlled or automatic	Application controlled, Automatic (Default)
Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to 0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Off
Protocol Mode	Mode of data exchange: 'Client' (Message Mode) or 'IO Server' For the **'Client'** mode the application program can use packets to read/write the process data of connected devices. For the **'IO Server'** mode, the communication partner has *read/write* access to the process data memory of the device from outside using of the function codes.	IO Server (default), Client
Data Swap	Data-storage mode: No: Data will not be swapped Yes: Data will be swapped.	Yes, No, Default: No
Map FC1 and FC3	If unchecked, data are read from the intput area using FC1, FC3 and FC23. If checked, data are read from the output area using FC1, FC3 and FC23. FC1 then can be used instead of FC2 and FC3 instead of FC4.	checked, unchecked, Default: unchecked
Bus		
Provided Server Connections	Server Connections Number of sockets to provide for server requests* *A value of 0 means that the Open Modbus/TCP task exclusively works as TCP Client. A value of 16 means that the Open Modbus/TCP task exclusively works as Server in the Message-Mode. With the default value 4 there are 4 Server connections provided. Then still up to 12 TCP client connections are available. The parameters 'Send Acknowledgement Timeout' (Send Timeout), 'Connect Acknowledgement Timeout' (Connect Timeout) and 'Close Acknowledgement Timeout' (Close Timeout) are for the Timeout between the Open Modbus/TCP Task and the TCP Task.	Sockets: 0 ... 16, Default: 4
Client Connection Watchdog Time	Connection remain open time Only for client jobs in Message Mode (packet mode). The connection to the destination-device stays open, until timeout is expired. **Note**: This timeout starts, after receiving the answer to a command For interface programming via Open Modbus/TCP, the value for Client Connection Watchdog Time (Omb Open Time) must be divided by the factor of 100. (Value range for the interface programming: 1 ... 60,000)	100 ... 6,000,000 ms, Default: 1000 ms
Response Timeout	Telegram Timeout Only for client jobs in Message Mode (packet mode). After expiration of this time, the job will be canceled and an error is send to the application. Note: This timeout starts after command is send to the destination device via TCP. For interface programming via Open Modbus/TCP, the value for Response Timeout must be divided by the factor of 100. (Value range for the interface programming: 1 ... 60,000)	100 ... 6,000,000 ms, Default: 2,000 ms
Send Acknowledge-ment Timeout	TCP Task SendTimeout Parameter Parameter for TCP task (in milliseconds) . Used OMB task internal. It specifies the timeout for trying to send messages via TCP/IP If the value 0 is selected, the default value of 31,000 milliseconds is used.	0 ... 2,000,000,000 ms, Default: 31,000 ms
Connect Acknowledge-ment Timeout	TCP Task Connect Timeout Parameter Parameter for TCP task (in milliseconds). Used OMB task internal. It specifies the timeout for trying to establish a connection with the TCP task. If the value 0 is selected, the default value of 31,000 milliseconds is used.	0 ... 2,000,000,000 ms, Default: 31,000 ms
Close Acknowledge-ment Timeout	TCP Task Close Timeout Parameter Parameter for TCP task (in milliseconds). Used OMB task internal. It specifies the timeout for trying to close a connection with the TCP task. If the value 0 is selected, the default value of 13,000 milliseconds is used.	0 ... 2,000,000,000 ms, Default: 13,000 ms

Parameter	Meaning	Range of Value/Value
IP Address	Valid IP address for the device If ‘Enabled’ is unchecked (Default setting), the device obtains its IP Address from a DHCP or BOOTP server. If ‘Enabled’ is checked, the device uses the manually entered value.	Valid IP address Default: unchecked
Net Mask	Valid Network mask for the device If ‘Enabled’ is unchecked (Default setting), the device obtains its Netmask from a DHCP or BOOTP server. If ‘Enabled’ is checked, the device uses the manually entered value.	Valid network mask Default: unchecked
Gateway	Valid Gateway address for the device If ‘Enabled’ is unchecked (Default setting), the device obtains its Gateway Address from a DHCP or BOOTP server. If ‘Enabled’ is checked, the device uses the manually entered value. There are three methods available, how the device can obtain its IP Address, Netmask and Gateway Address, one of which must be selected. These methods can also be combined. The device performs the following sequence in order to obtain the addresses: 1. from a DHCP server if DHCP is checked (if a DHCP server provides the requested addresses to the device, then the device uses these addresses) 2. from a BootP server if BootP is checked (if a BootP server provides the requested addresses to the device, then the device uses these addresses) 3. the addresses manually set are used. If the IP Address is set manually also the Network Mask must be set manually. The manually set Gateway Address is optional. If no DHCP server and no BootP server and no manually set addresses exist, then the protocol is not ready for initialization or for operation.	Valid gateway address Default: unchecked
Extras	BootP: If checked, the device obtains its IP Address, Netmask, Gateway Address from a BOOTP server.	Default: unchecked.
	DHCP: If checked, the device obtains its IP Address, Netmask, Gateway Address from a DHCP server.	Default: checked.

Table 14: Open Modbus/TCP Parameters

4.3.4 POWERLINK Controlled Node/Slave Parameters

Parameter	Meaning	Range of Value/Value
Interface		
Bus Startup	Communication start application controlled or automatic	Application controlled, Automatic Default: Automatic
Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to 0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Off
I/O Data Status	Status of the input or the output data. For each input and output data the following status information (in Byte) is memorized in the dual-port memory: Status 0 = None (default) Status 1 = 1 Byte (for future use) Status 2 = 4 Byte (for future use)	None, (1 Byte, 4 Byte) Default: None
Disable Host-Triggered Input Data Exchange	Decides, whether host-triggered update for input data exchange is enabled (hook checked) or disabled (hook unchecked).	Default: unchecked
Disable Host-Triggered Output Data Exchange	Decides, whether host-triggered update for output data exchange is enabled (hook checked) or disabled (hook unchecked).	Default: unchecked
Ident		
Enabled	If ‘Enabled’ is unchecked, for the single Ident parameters each the respective default value is used.	
Vendor ID	Identification number of the manufacturer	0x00000000 ... 0xFFFFFFFF (hex), Hilscher: 0x00000044 (hex)
Serial Number	Serial number of the device	0x00000000 ... 0xFFFFFFFF (hex), Default: 0x00000000 (hex)
Product Code	Product code of the device as specified by the manufacturer	0x00000000 ... 0xFFFFFFFF (hex), Default: CIFX RE/PLS: 0x00000001 (hex), COMX 100XX-RE/PLS: 0x00000003 (hex), NJ 50X-RE/PLS: 0x00000023 (hex), NJ 100XX-RE/PLS: 0x00000024 (hex)
Revision Number	Revision number of the device as specified by the manufacturer	0x00000000 ... 0xFFFFFFFF (hex), Default: 0x00000000 (hex)
Bus		
Node Id	EPL Node ID (EPL = Ethernet POWERLINK)	1...239, Default: 1
DNS Node Name	DNS-compatible name of the POWERLINK Controlled Node/Slave	
Gateway Address	Gateway address for IP stack	Default: 192.168.100.254
Data		
Input Data Bytes	Length of the input data in byte	1… 1490 Byte, Default: 4 Byte

Parameter	Meaning	Range of Value/Value
Output Data Bytes	Length of the output data in byte	1… 1490 Byte, Default: 4 Byte
Disable PDO Mapping Version Check	Decides, whether PDO mapping version is checked (hook checked) or not checked (hook unchecked).	Default: checked
Configure Default Objects	Decides, whether default objects are created (hook checked) or not (hook unchecked). If the objects will be created, the old set of previously existing objects will be cleared	Default: checked
Delete Application Objects	Decides, whether application objects are deleted (hook checked) or not (hook unchecked). When 'Configure Default Objects' is checked, then the firmware automatically does a 'Delete Application Objects'.	Default: checked

Table 15: POWERLINK Controlled Node/Slave Parameters

Note: To configure the Managing Node/Master, an XDD file (device description file) is required. The settings in the used Managing Node/Master must comply with the settings in the Controlled Node/Slave, to establish communication. Important parameters are: Vendor ID, Product Code, Serial Number, Revision Number, Node ID, Output and Input length.

4.3.5 PROFINET IO-Device Parameters

Parameter	Meaning	Range of Value/Value
Interface		
Bus Startup	Communication start application controlled or automatic	Application controlled, Automatic Default: Automatic
Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Off
I/O Data Status	Status of the input or the output data. For each input and output data the following status information (in Byte) is memorized in the dual-port memory: Status 0 = None (default) Status 1 = 1 Byte (for future use) Status 2 = 4 Byte (for future use)	None, (1 Byte, 4 Byte) Default: None
Ident		
Enabled	If ‘Enabled’ is unchecked, for the single Ident parameters each the respective default value is used.	
Vendor ID	Identification number of the manufacturer, assigned by PROFIBUS Nutzerorganisation e. V.	0x00000000 ... 0xFFFFFFFF (hex), Hilscher: 0x00000011E (hex)
Device ID	Identification number of the device, freely eligible by the manufacturer, fixed for every device.	0x00000000 ... 0x0000FFFF (hex), Default: CIFX RE/PNS: 0x00000103 (hex), COMX 50XX-REFO/PNS: 0x00000111 (hex), COMX 100XX-RE/PNS: 0x00000101 (hex), NJ 50X-RE/PNS: 0x00000112 (hex), NJ 100XX-RE/PNS: 0x00000113 (hex)
Device Type	Description of the device type, freely eligible	Character string, 0 ... 25 characters
Order ID	Hilscher device number or order description of the customer for its device	Character string, 0 ... 20 characters, e. g.: CIFX RE/PNS: 1250.100, COMX 50XX-REFO/PNS: 1551.110, COMX 100XX-RE/PNS: 153x.10x, NJ 50X-RE/PNS: 163x.100 NJ 100XX-RE/PNS: 162x.100
Name of Station	Station name of the PROFINET IO-Device station. It has to match the station name configured in the PROFINET IO-Controller for this device. Must be DNS compatible name, see GSDML file.	Character string, 1 ... 240 characters, Default: cifxrepns, comx50repns, comx100repns, nj50repns, nj100repns, nic50repns
Type of Station	Type name of the PROFINET station; name can be assigned freely.	Character string, 1 ... 240 characters Default: Station.Type

Parameter	Meaning	Range of Value/Value
Device Instance	The device instance must be selected according to the used firmware version and according the used device type. According to the version of the PROFINET IO Slave device firmware the device instance specifies, which features the device has. The device instance is a module of the GSDML to describe the device parameters device specific. Rule 1: Use a device instance (in the controller and in the device) that is supported by the used device firmware. Rule 2: The device instance that is configured in the controller for the device must match the device instance configured in the device. The list Device instances per device shows which device instance is possible per device and lists the reference to the firmware version and supported features.	See list Device instances per device on page 53.
Hardware Revision	The hardware revision can be read via PROFINET IO.	Hardware-Revision: 0 … 65535, Default: CIFX RE/PNS, COMX 50XX-REFO/PNS, COMX 100XX-RE/PNS, NJ 50X-RE/PNS, NJ 100XX-RE/PNS: 0
Software Revision 1,2,3 and Software Revision Prefix	The set of the **Software Revision 1,2,3** and **Software Revision Prefix** adds up to an audit software revision (version) that can be read via PROFINET IO. Example: v1.4.5 Software Revision 1 = 1, Software Revision 2 = 4, Software Revision 3 = 5, Software Revision Prefix = V These data serve for the versioning by the user, OEM. The version can be read via PROFINET IO. If the device is certified, these details appear on the certificate. The user has the responsibility for the information on the version. In particular, if a different firmware version is used, the user must adapt the user version. Use case 1: The user enters the firmware version corresponding to the used device's firmware and hardware revision of the used device. Use case 2: The user enters a custom version. The assignment of the firmware version to user version is performed by the user.	Software Revision 1: 0 … 65535, Default: CIFX RE/PNS, COMX 100XX-REFO/PNS, COMX 50XX-RE/PNS, NJ 50X-RE/PNS, NJ 100XX-RE/PNS: 0 _____ Software Revision 2: 0 … 65535, Default: CIFX RE/PNS, COMX 50XX-REFO/PNS, COMX 100XX-RE/PNS, NJ 50X-RE/PNS, NJ 100XX-RE/PNS: 0 _____ Software Revision 3: 0 … 65535, Default: CIFX RE/PNS, COMX 50XX-REFO/PNS, COMX 100XX-RE/PNS, NJ 50X-RE/PNS, NJ 100XX-RE/PNS: 0 _____ Software Revision Prefix: V: Released Version R: Revsion P: Prototype T: Test Device U: In field test Default: CIFX RE/PNS, COMX 50XX-REFO/PNS, COMX 100XX-RE/PNS, NJ 50X-RE/PNS, NJ 100XX-RE/PNS: V
Data		

Parameter	Meaning	Range of Value/Value
Output or Input	**Module:** for output modules for input modules **Note!** Output: If using the IO Controller, module 1 must be configured as Slot 1. Output modules = Slot 1 to 4 Input: If using the IO Controller, module 5 must be configured as Slot 5. Intput modules = Slot 5 to 8 Modules with a length of 0 (in the netX Configuration Tool) need not be configured in the IO controller. Example Default Settings: Modul 1: with 64 Byte output (must be configured in the IO Controller with Slot 1 = 64 Byte Output.) Modul 5: with 64 Byte input (must be configured in the IO Controller with Slot 5 = 64 Byte Input.)	1 … 4 5 … 8
	Type:	Byte, Integer8, Integer16, Integer32, Intege64, Unsigned8, Unsigned16, Unsigned32, Unsigned64, Float32, Float64 (each with consistence) Default: Byte
	Count: The number of Data Type in the module.	0, 1, 2, 3, 4, 8, 12, 16, 20, 32, 64 Possible selections for: Byte: 0, 1, 2, 3, 4, 8, 12, 16, 20, 32, 64 any other Data Types: 0, 1
Output Data Bytes	Total of the input identifier bytes of the modules 1 to 4	0 … 1024 Byte Default: 128 Byte
Input Data Bytes	Total of the output identifier bytes of the modules 5 to 8	0 … 1024 Byte Default: 128 Byte

Table 16: PROFINET IO-Device Parameters

Note: To configure the Controller, a GSDML file (device description file) is required. The settings in the used Controller must comply with the settings in the Device to establish communication. Important parameters are: Station Name, Vendor ID, Device ID, Input and Output Data Bytes.

Note: Under **Name of Station**, the name must be typed which was also used in the configuration file of the master of this device. If no name chosen freely is used in the configuration file, then the name from the GSDML file is used.

Device instances per device:

Device	Device Instance	Supported Features
CIFX RE/PNS	CIFX RE/PNS V3.1.x	Firmware version V3.1.x. Supports RT Communication.
	CIFX RE/PNS V3.2.x - V3.4.18	Firmware version V3.2.x - V3.4.18. Supports FastStartup, RT and IRT Communication.
	CIFX RE/PNS V3.4.19 - V3.4.x	Firmware version V3.4.19 - V3.4.x. Supports FastStartup, Identification & Maintenance 1-4, RT and IRT Communication.
COMX 50XX-REFO/PNS	COMX 50XX-REFO/PNS V3.4.19 - V3.4.x	Firmware version V3.4.19 - V3.4.x. Fiberoptics Hardware. Supports FastStartup, Identification & Maintenance 1-4, RT and IRT Communication.

Device	Device Instance	Supported Features
COMX 100XX-RE/PNS	COMX 100XX-RE/PNS V3.1.x	Firmware version V3.1.x. Supports RT Communication.
	COMX 100XX-RE/PNS V3.2.x - V3.4.18	Firmware version V3.2.x - V3.4.18. Supports FastStartup, RT and IRT Communication.
	COMX 100XX-RE/PNS V3.4.19 - V3.4.x	Firmware version V3.4.19 - V3.4.x. Supports FastStartup, Identification & Maintenance 1-4, RT and IRT Communication.
NJ 50X-RE/PNS	NJ 50X-RE/PNS V3.4.27 - V3.4.x	Firmware version V3.4.27 - V3.4.x. Supports FastStartup, Identification & Maintenance 1-4, RT and IRT Communication.
NJ 100XX-RE/PNS	NJ 100XX-RE/PNS V3.4.27 - V3.4.x	Firmware version V3.4.27 - V3.4.x. Supports FastStartup, Identification & Maintenance 1-4, RT and IRT Communication.

Table 17: PROFINET IO-Device Parameters Device Instances

4.3.6 sercos Slave Parameters

Parameter	Meaning	Range of Value/Value
Interface		
Bus Startup	Communication start application controlled or automatic	Application controlled, Automatic Default: Automatic
Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Off
Ident		
Device ID	Identification number of the device. This is a unique device ID managed by the vendor and identifies the component number. The device ID is fixed for every device. Device ID according to the sercos third generation specification as defined in IDN S-0-1300.x.05. The device ID can be changed by the user, if an other device ID than the default is useful for the used sercos network. **Note:** The **Device ID** is part of the SDDML device description file. If you use a sercos Master which is using SDDML files for configuration and the default device ID was changed, then you have to export via **Export SDDML** a new updated SDDML file from the Slave configuration software and import this SDDML file into the configuration software for the sercos Master.	Default: CIFX_RE_S3S_FIXCFG, COMX_100XX-RE_S3S_ FIXCFG, NJ_50X-RE_S3S_ FIXCFG, NJ_100XX-RE_S3S_ FIXCFG
Vendor Code	Identification number of the manufacturer, assigned by sercos International. Vendor Code according to the sercos third generation specification as defined in IDN S-0-1300.x.03. **Note:** The **Vendor Code** is part of the SDDML device description file. If you use a sercos Master which is using SDDML files for configuration and the default vendor code was changed, then you have to export via **Export SDDML** a new updated SDDML file from the Slave configuration software and import this SDDML file into the configuration software for the sercos Master.	0x00000000 ... 0xFFFFFFFF (hex), Hilscher: 0x0000003E8 (hex)
Version of SCP Sync	Version of the sercos Communication Profile SYNC 0 = SYNC deactivated. With this setting the configuration parameter for sercos Communication Profile Class SCP_Sync are not relevant and are displayed grayed out. 1 = SYNC activates version 1. If the sercos Master uses the telegram sequence MDT, then NRT and then AT, the device requires that the value is set to 1.	0 ... 255, Default: 0 Possible values: 0, 1
Version of SCP NRT	Version of the sercos Communication Profile NRT 0 = NRT deactivated. With this setting the configuration parameter for IP communication are not relevant and are displayed grayed out. 1 = NRT version 1 is activated. This function is not supported (value 0).	0 ... 255, Default: 0 Possible values: 0, 1
Bus		
	Configuration parameter for IP communication	
Enabled	If ‘Enabled’ is unchecked, for the single Ident parameters each the respective default value is used.	checked, unchecked
IP Address	Valid IP address for the device If ‘Enabled’ is unchecked (Default setting), the device obtains its IP Address from a DHCP or BOOTP server. If ‘Enabled’ is checked, the device uses the manually entered value. Only COMX 100XX-RE/S3S, NJ 50X-RE/S3S and NJ 100XX-RE/S3S support this function.	Valid IP address, Default for ‘Enabled’: unchecked

Parameter	Meaning	Range of Value/Value
Netmask	Valid Network mask for the device If 'Enabled' is unchecked (Default setting), the device obtains its Netmask from a DHCP or BOOTP server. If 'Enabled' is checked, the device uses the manually entered value. Only COMX 100XX-RE/S3S, NJ 50X-RE/S3S and NJ 100XX-RE/S3S support this function.	Valid network mask, Default for 'Enabled': unchecked
Gateway	Valid Gateway address for the device If 'Enabled' is unchecked (Default setting), the device obtains its Gateway Address from a DHCP or BOOTP server. If 'Enabled' is checked, the device uses the manually entered value. Only COMX 100XX-RE/S3S, NJ 50X-RE/S3S and NJ 100XX-RE/S3S support this function.	Valid gateway address, Default for 'Enabled': unchecked
	There are three methods available, how the device can obtain its IP Address, Netmask and Gateway Address, one of which must be selected. These methods can also be combined. The device performs the following sequence in order to obtain the addresses: 1. from a DHCP server if DHCP is checked (if a DHCP server provides the requested addresses to the device, then the device uses these addresses) 2. from a BootP server if BootP is checked (if a BootP server provides the requested addresses to the device, then the device uses these addresses) 3. the addresses manually set are used. If the IP Address is set manually also the Network Mask must be set manually. The manually set Gateway Address is optional. If no DHCP server and no BootP server and no manually set addresses exist, then the protocol is not ready for initialization or for operation.	
TCP Flags	BootP: If checked, the device obtains its IP Address, Netmask, Gateway Address from a BOOTP server. Only COMX 100XX-RE/S3S, NJ 50X-RE/S3S and NJ 100XX-RE/S3S support this function.	checked, unchecked, Default: unchecked
	DHCP: If checked, the device obtains its IP Address, Netmask, Gateway Address from a DHCP server. Only COMX 100XX-RE/S3S, NJ 50X-RE/S3S and NJ 100XX-RE/S3S support this function.	checked, unchecked, Default: unchecked
	Configuration Data Parameters of the Sercos Communication Profile Class SCP_Sync	
Control Clock Length*	This timing parameter defines the time how long the communication synchronized hardware output signal CON-CLK is set. The maximum valid value of the „Control Clock Length" depends on the configured cycle time. In general the signal CON_CLK must go to inactive again before the next cycle starts.	>= 1000 ns, Default: 1000 ns
Time Divided Control Clock*	This timing parameter defines the delay time from the start of the communicatin cycle to the first pulse of the communication synchronized hardware output signal DIV_CLK.	0 ... 16.777.210 ns, Default: 1000 ns
Delay Time of Divided Clock*	This timing parameter defines the delay time respectively the distance between two pulses of the communication synchronized hardware output signal DIV_CLK. In the „Divided Control Clock" Mode1 this parameter is ignored.	0 ... 6.710.860 ns, Default: 1000 ns
Divided Control Clock Length*	This timing parameter defines the pulse length, i. e. how long the communication synchronized hardware output signal DIV_CLK is set.	1000 ... 20.000 ns, Default: 1000 ns
	*All timing values are handled as multiple of 10 ns (e. g. Control Clock Length = 1005 ns is handled as 1000 ns).	

Parameter	Meaning	Range of Value/Value
Count Register	For „Divided Control Clock“ Mode0: Here the parameter indicates the number of pulses of the communication synchronized hardware output signal DIV_CLK within a communication cycle. For „Divided Control Clock“ Mode1: Here the parameter indicates the number of communication cycles.	0 ... 255, Default: 100
Divided Control Clock Output	This communication synchronization flag defines the output state of the communication synchronized hardware output signal DIV_CLK.	Disabled, High Activity, Low Activity, Default: High Activity
Divided Control Clock Mode	This communication synchronization flag defines the mode of the communication synchronized hardware output signal DIV_CLK. Mode 0: The Div_Clk signal becomes active several times within a communication cycle. The following condition must be fulfilled in mode 0: „Divided Control Clock Length“ + 100 <= „Delay Time of Divided Clock“. Mode 1: The Div_Clk signal becomes active once after N communication cycles.	Mode 0, Mode 1, Default: Mode 0
Controller Clock Output	This communication synchronization flag defines the output state of the communication synchronized hardware output signal CON_CLK.	Disabled, High Activity, Low Activity, Default: Disabled
Slave Configuration		
Number of Slaves	The number of used sercos addresses	1 … 8, Default:1
Slave 1 … 8		
sercos Address	Address for the sercos Slave. The address range is from 0 to 511.	1 … 511, Default: 1
FSP Type	Function Specific Profile Type according to IDN S-0-1302.x.01 FSP Type & Version: The FSP Type indicates the function specific type of the resource. Supported FSP profiles: FSP_IO, FSP_Drive FSP IO: A specification for IO Devices to be controlled by sercos, FSP Drive: A specification for mechanical drives to be controlled by sercos	IO V1, Drive V1, Default: Drive V1
SCP Configuration Type	SCP config type according to sercos specification third generation: 0x0101 - SCP_FixCFG Version 1.1.1, 0x0201 - SCP_VarCFG Version 1.1.1, 0x0202 - SCP_VarCFG Version 1.1.3 According to the sercos specification third generation the sercos Communication Profiles Classes SCP-FixCFG and SCP-VarCFG define the basic communication and are mutually exclusive. SCP-FixCFG: Exact two connections are supported, one as consumer and one as producer. The content of the connections is defined by the Slave and cannot be changed by the Master. SCP-VarCFG: A certain number of connections is supported. The Slave defines this number and provides it to the Master. The content of all connections has to be configured (e.g. by the Master). The Slave provides lists of IDNs, that can be cyclically produced and consumed, so the Master can find it out.	Fix. Version 1.1.1, Var. Version 1.1.1, Var. Version 1.1.3 Default Fix. Version 1.1.1
Output Data Size for Fixed Configuration	Output Data Size in Byte for „SCP Configuration Type“ / „Fix. Version 1.1.1“ **Note:** The output data size is part of the SDDML device description file. If you use a sercos Master which is using SDDML files for configuration and the default output data size was changed, then you have to export via **Export SDDML** a new updated SDDML file from the Slave configuration software and import this SDDML file into the configuration software for the sercos Master.	0 … 200 Byte, Default: 2 Byte

Parameter	Meaning	Range of Value/Value
Input Data Size for Fixed Configuration	Intput Data Size in Byte for „SCP Configuration Type“ / „Fix. Version 1.1.1“ **Note:** The input data size is part of the SDDML device description file. If you use a sercos Master which is using SDDML files for configuration and the default input data size was changed, then you have to export via **Export SDDML** a new updated SDDML file from the Slave configuration software and import this SDDML file into the configuration software for the sercos Master.	0 … 200 Byte, Default: 2 Byte
User SCP Types	Up to 20 User SCP Types are possible. The following User SCP types are already defined: 0x0401 - SCP_WD Version 1.1.1 for monitoring connections, 0x0501 - SCP_Diag Version 1.1.1 for bus-diagnosis, 0x0601 - SCP_RTB Version 1.1.1 for using Real time bits, 0x0901 - SCP_Mux Version 1.1.1 for multiplexed cyclic data, 0x0B01 - SCP_Sig Version 1.1.1 for using signal status/control words	SCP_WD Version 1.1.1, SCP_Diag Version 1.1.1, SCP_RTB Version 1.1.1, SCP_Mux Version 1.1.1, SCP_Sig Version 1.1.1
Slave Flags	Setup a Default OD	checked, unchecked, Default: unchecked
	Non Stack Objects Are Deleted (for Reset/Channel Init)	checked, unchecked, Default: unchecked
	sercos Address May not be Changed by the Master: enables (unchecked) or disables (checked), that the Master changes the sercos address.	checked, unchecked, Default: unchecked
Connection Control Offset	Connection Control Offset for the Slave connections 1, 2, 3 and 4	0 ... 5.758, Default: 0
Real Time Data Process Image Offset	Real Time Data Process Image Offset for the Slave connections 1, 2, 3 and 4	0 ... 5.760, Default: 0
Real Time Data Maximum Length 1, 2, 3 and 4	Real Time Data Maximum Length for the Slave connections 1, 2, 3 and 4	0 … 200 Byte, Default for Real Time Data Maximum Length 1, 2: 198 Byte, Real Time Data Maximum Length 3, 4: 0 Byte
Export SDDML	The parameters Device ID, Vendor Code, Input Data Size and Output Data Size are part of the SDDML device description file. If for the configuration of the sercos Master SDDML files are used and a default value of one of these parameters was changed, then a SDDML file must be created via **Export SDDML** and then used in the configuration of the SERCOS Master.	

Table 18: sercos Slave Parameter

Note: To configure the Master, an XML file (device description file) is required. The settings in the used Master must comply with the settings in the Slave to establish communication.

4.3.7 VARAN-Client (Slave) Parameters

Parameter	Meaning	Range of Value/Value
Interface		
Bus Startup	Communication start application controlled or automatic	Application controlled, Automatic; Default: Automatic
Client Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Off
I/O Data Status*	Status of the input or the output data. *(not supported)	
Ident		
Enable	If 'Enable' is unchecked, for the single Ident parameters each the respective default value is used.	Checked, unchecked, Default: unchecked
Vendor ID	Vendor Identifier: Identification number of the manufacturer. This value is specific for every single vendor.	0x00000000 ... 0xFFFFFFFF (hex), Hilscher: 0x0000001D(hex)
Vendor Name	Name of the manufacturer, specific for every single vendor.	Character string, 0 ... 64 characters, Default: Hilscher GmbH
Device ID	Device Identifier: Identification number of the device as indicated by the VARAN-BUS-NUTZERORGANISATION. This value is specific for every single device type and is fixed for every device. In case of an unknown device ID the manager rejects the client.	0x00000000 … 0xFFFFFFFF (hex), CIFX RE/VRS: 0x0000048C (hex); COMX 100XX-RE/VRS: 0x0000048A (hex); NJ 50X-RE/VRS: 0x00000488 (hex); NJ 100XX-RE/VRS: 0x00000489 (hex)
Device Name	Name of the device as indicated by the VARAN-BUS-NUTZERORGANISATION.	Character string, 0 ... 64 characters, Default: CIFX RE/VRS, COMX100xx-RE/VRS, NJ50x-RE/VRS, NJ100xx-RE/VRS
Product Version	Version of the device as specified by the manufacturer.	0x00000000 ... 0xFFFFFFFF (hex), Default: 0x00000000 (hex)
Order Number	Order number of the device as specified by the manufacturer.	0x00000000 ... 0xFFFFFFFF (hex) , Default: 0x00000000 (hex)
Serial Number	Serial number of the device as specified by the manufacturer.	0x00000000 ... 0xFFFFFFFF (hex) , Default: 0x00000000 (hex)
License Number	License number of the device as specified by the VARAN-BUS-NUTZERORGANISATION.	0x00000000 ... 0xFFFFFFFF (hex) , Default: 0x00000000 (hex)
Bus		

Parameter	Meaning	Range of Value/Value
Client Watchdog Time	Client watchdog time in ms. The Client watchdog time is the maximum cyclic communication timeout allowed, before Client signals a bus error. Example: Client watchdog time set to 50ms and Client is unplugged from the network (or the Manager is set into stop mode), then the Client goes in an error state on the 50th ms after this event.	[0 … 130] ms, Default = 130 ms, 0 = Off
Memory Area 2*	If ‘Memory Area 2’ is checked, the parameters for Memory Area 2 are enabled. (*currently not supported, set to default value)	Checked, unchecked, Default: unchecked
EMAC*	EMAC = Extended MAC-Address If ‘EMAC’ is checked, an integrated EMAC for IP data exchange with client application is used. (*currently not supported, set to default value)	Checked, unchecked, Default: unchecked
Memory Area 1 / Memory Area 2		
Read Size / Memory Area 1	Reading size of the memory area 1in Byte.	0 … 128 Byte, Default: 128 Byte
Read Size / Memory Area 2*	Reading size of the memory area 2 in Byte. *(not supported)	0 … 128 Byte, Default: 0 Byte
Write Size / Memory Area 1	Writing size of the memory area 1in Byte.	0 … 128 Byte, Default: 128 Byte
Write Size / Memory Area 2*	Writing size of the memory area 2 in Byte. *(not supported)	0 … 128 Byte, Default: 0 Byte
Read Offset / Memory Area 1	Reading offset of the memory area 1in Byte.	0x00000000 ... 0x0000FFFF (hex), Default: 0x00002000 (hex)
Read Offset / Memory Area 2*	Reading offset of the memory area 2 in Byte. *(not supported)	0x00000000 ... 0x0000FFFF (hex) , Default: 0x0000FFFF (hex)
Write Offset / Memory Area 1	Writing offset of the memory area 1 in Byte.	0x00000000 ... 0x0000FFFF (hex) , Default: 0x00002000 (hex)
Write Offset / Memory Area 2*	Writing offset of the memory area 2 in Byte. *(not supported)	0x00000000 ... 0x0000FFFF (hex) , Default: 0x0000FFFF (hex)
Sync OUT 0 / Sync OUT 1  **Note!** Only the following 3 combinations between SyncOut0Mode + SyncOut1Mode are reasonable: "Time for IN/OUT valid (TIO)" + "Disable", "Time for IN valid (TI)" + "Time for OUT valid (T0)", "Time for OUT valid (TO)" + "Time for IN valid (TI)"		
Mode / Sync OUT 0	"SyncOut 0 Mode“of the SYNC Out 0 signal. Using the setting - "Time for IN Valid (TI)“ for the output of the SYNC Out 0 signal the time for the data IN is valid, - "Time for OUT Valid (TO)“ for the output of the SYNC Out 0 signal the time for the data OUT is valid, - "Time for IN/OUT Valid (TIO)“ for the output of the SYNC Out 0 signal the time for the data IN/OUT is valid.	Time for IN/OUT Valid (TIO), Time for IN Valid (TI), Time for OUT Valid (TO), Default: Time for IN/OUT Valid (TIO)
Mode / Sync OUT 1	The "SyncOut 1 Mode “ applies the "SYNC OUT 1 ID” to the output of the SYNC Out 1 signal. Using the setting - "Disable“ means the output of the SYNC Out 1 signal is not available, - "Time for IN Valid (TI)“ for the output of the SYNC Out 1 signal the time for the data IN is valid, - "Time for OUT Valid (TO)“ for the output of the SYNC Out 1 signal the time for the data OUT is valid.	Disable, Time for IN Valid (TI), Time for OUT Valid (TO), Default: Disable

Parameter	Meaning	Range of Value/Value
Output / Sync OUT 0	The “Sync out 0 flag/Output” enables / disables the output of the Memory Area 1 SYNC Out 0 signal.	Enable, Disable, Default: Enable
Output / Sync OUT 1	The “Sync out 1 flag/Output” enables / disables the output of the Memory Area 2 SYNC Out 1 signal.	Enable, Disable, Default: Disable
Polarity / Sync OUT 0	The “Sync out 0 flag/Polarity” defines the active high/low behavior of the SYNC Out 0 signal.	Active Low, Active High, Default: Active Low
Polarity / Sync OUT 1	The “Sync out 1 flag/Polarity” defines the active high/low behavior of the Memory Area 2 SYNC Out 1 signal.	Active Low, Active High, Default: Active Low
Pulse Length (x10ns)	Pulse length of the SYNC Out signal in 10ns steps. (e.g. the value 100 results as 10ns*100=1000ns=1µs pulse)	0 … 2.147.483.647, Default: 100
Data		
Recv Data Length	Length of the received data in Byte.	Default: 128 Byte
Send Data Length	Length of the send data in Byte.	Default: 128 Byte

Table 19: VARAN Client (Slave) Parameters

Note: The settings in the used VARAN Manager (Master) must comply with the settings in the Client to establish communication. Important parameters are: Station Name, Vendor ID, Device ID, Recv Data Length and Send Data Length and Memory Area Read Offset / Write Offset.

4.4 Configuration Parameters Fieldbus Systems

4.4.1 PROFIBUS DP Slave Parameters

Parameter	Meaning	Range of Value/Value
Interface		
Bus Startup	Communication start application controlled or automatic	Application controlled, Automatic (Default)
Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to 0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Off
I/O Data Status	Status of the input or the output data. For each input and output data the following status information (in Byte) is memorized in the dual-port memory: Status 0 = None (default) Status 1 = 1 Byte (for future use) Status 2 = 4 Byte (for future use)	None, (1 Byte, 4 Byte) Default: None
Ident		
Ident Number	PROFIBUS Identification Number	0x00000000 ... 0x0000FFFF (hex), Default: CIFX DP/DPS: 0x000000B69 (hex), COMX 10XX-DPS/DPS: 0x000000D82 (hex), COMX 100XX-DP/DPS: 0x000000C0F (hex), NJ 100XX-DP/DPS: 0x00000D83 (hex)
Enable	If ‘Enabled’ is unchecked, the default value is used.	
Bus		
Station Address	PROFIBUS address of the device	0 … 126
Enable Address Switch	Defines, if the station address is configured in the configuration software or at the address switch. If checked, the station address is configured at the address switch. The parameter ‘Enable Address Switch’ with the setting ‘checked’ can only be used for COMX 10XX-DPS/DPS.	Default: CIFX DP/DPS, COMX 100XX-DP/DPS, NJ 100XX-DP/DPS: unchecked COMX 10XX-DPS/DPS: checked.
Baudrate	Network Baud Rate	9,6 kBit/s 19,2 kBit/s 93,75 kBit/s 187,5 kBit/s 500 kBit/s 1,5 MBit/s 3 MBit/s 6 MBit/s 12 MBit/s 31,25 kBit/s 45,45 kBit/s Auto-Detect Default: Auto-Detect
Extras	DPV1 Enable: If checked, DPV1 is supported or the DPV1 functions are activated.	Default: checked
	Sync supported: If checked, the Slave stack supports the SYNC command or the SYNC mode is activated.	

Parameter	Meaning	Range of Value/Value
	Freeze supported: If checked, the Slave stack supports the FREEZE command or the FREEZE mode is activated.	
	Fail safe supported: If checked, the FAILSAFE operation is supported or the FAILSAFE mode is activated.	
	Address change not allowed: If checked, the Slave stack supports the Set Slave Address command. The bus address can be changed via the Master.	
Data		
Output or Input	**Module:** for output modules for input modules	 1 … 4 5 … 8
	Type: Byte or Word	„Byte“ (Default), „Word“ each with consistency
	Size: The number of Bytes or Words in the module.	0, 1, 2, 3, 4, 8, 12, 16, 20, 32, 64 (Byte, Words)
Output Data Bytes	Total of the output identifier bytes of the modules 1 to 4	0 … 244, Default: 2
Input Data Bytes	Total of the input identifier bytes of the modules 5 to 8	0 … 244, Default: 2
Manual Input	Manual Input: If unchecked the field Configuration Data shows the output and input identifier bytes, which results from the settings of the output modules and input modules. If checked the field Configuration Data is editable. Output and input identifier bytes can be entered into the Configuration Data field to configure the device. Then, the settings of the output modules and input modules have no meaning.	Default: unchecked
Configuration Data	Configuration data for the output and input identifier bytes. The identifier bytes consists of the **Type** and the **Size**. The identifier bytes are the general identifier bytes according to the PROFIBUS standard.	Default: A1, 91 hex

Table 20: Parameters - PROFIBUS-DP Slave

Note: The input and output modules each work with ’consistency’.

Note: To configure the Master, a GSD file (device description file) is required. The settings in the used Master must comply with the settings in the Slave to establish communication. Important parameters are: Station Address, Ident Number, Baudrate and Config Data (the configuration data for the output and input length).

4.4.2 PROFIBUS-MPI

Parameter	Meaning	Range of Value/ Value
Interface		
Bus Startup	Communication start application controlled or automatic	Application controlled, Automatic (Default)
Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to 0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, Default = 1000 ms, 0 = Off
Bus		
Station Address	PROFIBUS MPI address of the device.	0 … 126, Default: 1
Baud Rate	PROFIBUS MPI Baud Rate Indicates the velocity of transmission: Number of bits per second. **Important!** - The **Baud Rate** setting is mandatory. - The option *Auto Detect* is only possible if the communication is initiated by another bus participant.	9,6 kBit/s, 19,2 kBit/s, 31,25 kBit/s, 45,45 kBit/s, 93,75 kBit/s, 187,5 kBit/s, 500 kBit/s, 1,5 MBit/s, 3 MBit/s, 6 MBit/s, 12 MBit/s, Auto Detect, Default: 187,5 kBit/s
Slot Time	Slot Time (T_{SL}) 'Wait for receipt' – monitoring time of the Senders (Requestor) of telegram for the acknowledgement of the recipient (Responder). After expiration, a retry occurs in accordance with the value of 'Max. telegram retries'.	37 ... 16383 tBit, Default: 415 tBit
Min. Station Delay Responder	Minimum Station Delay of Responders (min T_{SDR}) This is the shortest time period that must elapse before a remote recipient (Responder) may send an acknowledgement of a received query telegram. The shortest time period between receipt of the last Bit of a telegram to the sending of the first Bit of a following telegram.	1 ... 1023 tBit, Default: 60 tBit
Max. Station Delay Responder	Maximum Station Delay of Responders (max T_{SDR}) This is the longest time period that must elapse before a Sender (Requestor) may send a further query telegram. Greatest time period between receipt of the last Bit of a telegram to the sending of the first Bit of a following telegram. The Sender (Requestor, Master) must wait at least for this time period after the sending of an unacknowledged telegram (e.g. Broadcast only) before a new telegram is sent.	1 ... 1023 tBit, Default: 400 tBit
Quiet Time	Quiet Time (T_{QUI}) This is the time delay that occurs for modulators (Modulator-trip time) and Repeaters (Repeater-switch time) for the change over from sending to receiving.	0 ... 127 tBit, Default: 1 tBit
Setup Time	Setup Time (T_{SET}) Minimum period “reaction time” between the receipt of an acknowledgement to the sending of a new query telegram (Reaction) by the Sender (Requestor).	1 … 255 tBit, Default: 1 tBit
Target Rotation Time	Target Rotation Time (T_{TR}) Pre-set nominal Token cycling time within the Sender authorization (Token) will cycle around the ring. How much time still is available for sending data telegrams is dependent on the difference between the nominal and the actual token cycling time.	>= 255 tBit, Default: 10000 tBit
GAP Actualization Factor	GAP Actualization Factor (G) Factor for determining after how many Token cycles an added participant is accepted into the Token ring. After expiry of the time period G*TTR, the Station searches to see whether a further participant wishes to be accepted into the logical ring.	1 … 255, Default: 20

Parameter	Meaning	Range of Value/ Value
Highest Station Address	Highest Station Address (H_{SA}) Station address of the highest active (Master) station.	1 … 126, Default: 31
Max. Retry Limit	Max number of telegram retries (Max_Retry_Limit) Maximum number of repetitions allowed for reaching a station.	1 ... 8, Default: 1
Bit times: All times for the Bus parameters are given in Bit times. The Bit time tBit is the result of the reciprocal of the Baud rate: tBit = 1 / Baud rate (Baud rate in Bit/s) The conversion from milliseconds into a Bit time is shown in the following formula: Bit time = Time [milliseconds] * Baud rate		

Table 21: Parameters – PROFIBUS MPI

4.4.3 CANopen Slave Parameters

Parameter	Meaning	Range of Value/Value
Interface		
Bus Startup	Communication start application controlled or automatic	Application controlled, Automatic (Default)
Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to 0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Off
I/O Data Status	Status of the input or the output data. For each input and output data the following status information (in Byte) is memorized in the dual-port memory: Status 0 = None (default) Status 1 = 1 Byte (for future use) Status 2 = 4 Byte (for future use)	None, (1 Byte, 4 Byte) Default: None
Ident		
Enabled	If ‘Enabled’ is unchecked, for the single Ident parameters each the respective default value is used.	
Vendor ID	Identification number of the manufacturer	0x00000000 ... 0x0000FFFF (hex), Hilscher: 0x00000044 (hex)
Product Code	Product code of the device as specified by the manufacturer	0x00000000 ... 0xFFFFFFFF (hex), Default: CIFX CO/COS: 0x001314C4 (hex), COMX 10XX-COS/COS: 0x0017D3C4 (hex), COMX 100XX-CO/COS, 0x00175E94 (hex), NJ 100XX-CO/COS: 0x0018CD9C (hex)
Revision Number	Revision number of the device as specified by the manufacturer	0x00000000 ... 0xFFFFFFFF (hex), Default: 0x00020000 (hex)
Serial Number	Serial number of the device	0x00000000 ... 0xFFFFFFFF (hex)
Bus		
Node Address	Node ID of CANopen slave	1 … 127, Default: 2
Enable Address Switch	Defines, if the node address is configured in the configuration software or at the address switch. If checked, the node address is configured at the address switch. The parameter ‘Enable Address Switch’ with the setting ‘checked’ can only be used for COMX 10XX-COS/COS.	Default: CIFX CO/COS, COMX 100XX-CO/COS, NJ 100XX-CO/COS: unchecked COMX 10XX-COS/COS: checked
Baudrate	Baud rate of CANopen connection	Auto-Detect 1 Mbaud, 800 Kbaud, 500 Kbaud, 250 Kbaud, 125 Kbaud, 100 Kbaud, 50 Kbaud, 20 Kbaud, 10 Kbaud, Default: 1 MBaud

Parameter	Meaning	Range of Value/Value
Data		
Send Object/ Receive Object	**Send Object**: Send object index **Receive Object**: Receive object index	0x00002000 ... 0x00002003 (hex) 0x00002200 ... 0x00002203 (hex)
	Size: Number of data Bytes to send per send object or number of data Bytes to receive per send object.	CIFX CO/COS, COMX 100XX-CO/COS, NJ 100XX-CO/COS: 128 COMX 10XX-COS/COS: 16
Output Data Bytes	Total output data Bytes of all send objects	CIFX CO/COS, COMX 100XX-CO/COS, NJ 100XX-CO/COS: Default: 512 Bytes* COMX 10XX-COS/COS: Default: 64 Bytes*
Input Data Bytes	Total intput data Bytes of all receive objects	CIFX CO/COS, COMX 100XX-CO/COS, NJ 100XX-CO/COS: Default: 512 Bytes* COMX 10XX-COS/COS: Default: 64 Bytes*

Table 22: CANopen Slave Parameters

Note: To configure the Master, an EDS file (device description file) is required. The settings in the used Master must comply with the settings in the Slave to establish communication. Important parameters are: Node Address and Baudrate.

Note: *The maximum number of output data bytes and input data bytes is for the device 512 bytes respectively 64 bytes each. The number of output data bytes and input data bytes is not adjustable in the Slave and therefore greyed out in the dialog.

The number of output data bytes and input data bytes, which are to be transferred effectively between the CANopen Master and Slave, are configured in the used CANopen Master. The CANopen Master configures the device during establishing the communication and set the number of output data bytes and input data bytes at this same time.

4.4.4 DeviceNet Slave Parameters

Parameter	Meaning	Range of Value/Value
Interface		
Bus Startup	Communication start application controlled or automatic	Application controlled, Automatic (Default)
Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to 0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Off
I/O Data Status	Status of the input or the output data. For each input and output data the following status information (in Byte) is memorized in the dual-port memory: Status 0 = None (default) Status 1 = 1 Byte (for future use) Status 2 = 4 Byte (for future use)	None, (1 Byte, 4 Byte) Default: None
Ident		
Enabled	If ‘Enabled’ is unchecked, for the single Ident parameters each the respective default value is used.	
Vendor ID	Identification number of the manufacturer	0x00000000 ... 0x0000FFFF (hex), Hilscher: 0x00000011B (hex)
Product Code	Product code of the device	0x00000000 ... 0xFFFFFFFF (hex), Default CIFX DN/DNS: 0x0000001C (hex), COMX 10XX-DNS/DNS: 0x00000036, COMX 100XX-DN/DNS: 0x00000032, NJ 100XX-DN/DNS: 0x00000034
Serial Number	Serial number of the device	0x00000000 ... 0xFFFFFFFF (hex)
Product Type	Communication Adapter	0x00000000 ... 0x0000FFFF (hex), Default: 0x00000000C (hex)
Minor Rev	Minor Revision	1 ... 255, Default: 1
Major Rev	Major Revision	1 ... 255, Default: 1
Product Name	The variable Product Name is a text string that should represent a short description of the product/product family.	0 ... 31 ASCII Characters
Bus		
MAC ID	This parameter defines the DeviceNet address of the device within the network.	0 … 63, Default: 2
Enable Address Switch	Defines, if the MAC ID is configured in the configuration software or at the address switch. If checked, the MAC ID is configured at the address switch. The parameter ‘Enable Address Switch’ with the setting ‘checked’ can only be used for COMX 10XX-DNS/DNS.	Default: CIFX DN/DNS, COMX 100XX-DN/DNS, NJ 100XX-DN/DNS: unchecked COMX 10XX-DNS/DNS: checked.

Baudrate	Baud rate of DeviceNet connection	500 kBaud, 250 kBaud, 125 kBaud, Default: 500 kBaud
Extras	Ignore address switch: address switches are always ignored	Default: unchecked
	Continue on CAN bus off: Unchecked: A device reset by the user is necessary in case of a CAN bus off event (e. g. short circuit of the data lines) Checked: The device tries independently to continue operation in case of a CAN bus off event	Default: unchecked
	Continue On Loss of Network Power (NP): Function not supported	Default: unchecked
	Receive-Idle Clear Data: Unchecked: Received data keep their last state in case of idle state Checked: Received data were set to zero in case of idle state	Default: unchecked
	Receive Idle keeps Data: Function not supported	Default: unchecked
Data		
Prod. Data Length	Produced data length sets the number of send bytes.	0 … 255, Default: 8
Cons. Data Length	Consumed data length sets the number of receive bytes.	0 … 255, Default: 8

Table 23: DeviceNet Slave Parameters

Note: To configure the Master, an EDS file (device description file) is required. The settings in the used Master must comply with the settings in the Slave to establish communication. Important parameters are: MAC ID, Baudrate, Produced Size, Consumed Size, Vendor ID, Product Type, Product Code, Major Rev, Minor Rev.

4.4.5 CompoNet Slave Parameters

Parameter	Meaning	Range of Value/Value
Interface		
Bus Startup	Communication start application controlled or automatic	Application controlled, Automatic (Default)
Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to 0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Off
I/O Data Status	Status of the input or the output data. For each input and output data the following status information (in Byte) is memorized in the dual-port memory: Status 0 = None (default) Status 1 = 1 Byte (for future use) Status 2 = 4 Byte (for future use)	None, (1 Byte, 4 Byte) Default: None
Ident		
Enable	If 'Enable' is unchecked, for the single Ident parameters each the respective default value is used.	
Vendor ID	Identification number of the manufacturer *If the value 0x00000000 (hex) is entered, the firmware uses the internal Hilscher ID 0x0000011B (hex).*	0x00000000 ... 0x0000FFFF (hex), Hilscher: 0x0000011B (hex)
Product Code	Product code of the device *The firmware uses the entered value.*	0x00000000 ... 0xFFFFFFFF (hex), Default: CIFX CP/CPS: 0x00000201 (hex)
Serial Number	Serial number of the device *If the value 0x00000000 (hex) is entered, the firmware uses the internal device serial number.*	0x00000000 ... 0xFFFFFFFF (hex)
Product Type	Communication Adapter *If the value 0x00000000 (hex) is entered, the firmware uses the value 0x0000000C (hex).*	0x00000000 ... 0x0000FFFF (hex), Default: 0x0000000C (hex)
Major Rev	Major Revision *If the value 0 is entered, the firmware uses the value 1.*	0 ... 255, Default: 1
Minor Rev	Minor Revision *If the value 0 is entered, the firmware uses the value 1.*	0 ... 255, Default: 1
Product Name	The variable Product Name is a text string that should represent a short description of the product/product family. *If no product name is entered, the device uses an internal default name.*	0 ... 31 ASCII Characters
Bus		
Node Mode	Node Mode of the CompoNet Slave	Word MIX Slave, Word IN Slave, Word OUT Slave, Bit MIX Slave, Bit IN Slave, Bit OUT Slave, Default: Word MIX Slave
Node Address	Node Address of the CompoNet Slave	Default: 2

<table>
<tr><th>Parameter</th><th>Meaning</th><th>Range of Value/Value</th></tr>
<tr><td>MAC ID</td><td>The MAC ID defines the CompoNet address of the device within the network.
The MAC ID is calculated from the selected Node Type and from the selected Node Address according to the following table. In this field the MAC ID is only displayed.
<table>
<tr><th>Node Type</th><th>Node Address</th><th>MAC ID</th></tr>
<tr><td>Word IN</td><td>0-0x3F</td><td>0x0-0x3F (0-63)</td></tr>
<tr><td>Word OUT</td><td>0-0x3F</td><td>0x40-0x7F (64-127)</td></tr>
<tr><td>Word MIX</td><td>0-0x3F</td><td>0x0-0x3F (0-63)</td></tr>
<tr><td>Bit IN</td><td>0-0x7F</td><td>0x80-0xFF (128-255)</td></tr>
<tr><td>Bit OUT</td><td>0-0x7F</td><td>0x100-0x17F (256-383)</td></tr>
<tr><td>Bit MIX</td><td>0-0x7F</td><td>0x80-0xFF (128-255)</td></tr>
</table></td><td>0 … 383,
Default: 2 (when using the default values for node type and node address)</td></tr>
<tr><td>Baudrate</td><td>Baud rate of the CompoNet connection</td><td>Auto-Detect,
93,75 kbps,
1,5 Mbps,
3 Mbps,
4 Mbps
Default: Auto-Detect</td></tr>
<tr><td colspan="3">Data</td></tr>
<tr><td>Produced Data</td><td>Produced Data sets the number of send points (Bits).
Produced data can be selected for the Node Type „IN“ and „MIXED“, for „OUT“ it is grayed out.</td><td>Node Type „Bit“:
2, 4 (Points), Default: 2
Node Type „Word“:
8, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 170, 192, 208, 224, 240, 256 (Points), Default:16</td></tr>
<tr><td>Size (Produced Data)</td><td>The number of bytes in the module for the produced data.</td><td>1 … 32 (Bytes)*
[*= 8 … 256 Points]
Default
Node Type „Bit“: 1
Node Type „Word“: 2</td></tr>
<tr><td>Consumed Data</td><td>Consumed Data sets the number of receive points (Bits).
Consumed data can be selected for the Node Type „OUT“ and „MIXED“, for „IN“ it is grayed out.</td><td>Node Type „Bit“:
2, 4 (Points), Default: 2
Node Type „Word“:
8, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 170, 192, 208, 224, 240, 256 (Points), Default:16</td></tr>
<tr><td>Size (Consumed Data)</td><td>The number of bytes in the module for the consumed data.</td><td>1 … 32 (Bytes)*
[*= 8 … 256 Points]
Default
Node Type „Bit“: 1
Node Type „Word“: 2</td></tr>
</table>

Table 24: CompoNet Slave Parameters

Note: To configure the Master, an EDS file (device description file) is required. The settings in the used Master must comply with the settings in the Slave to establish communication. Important parameters are: Node Mode, MAC ID, Baudrate, Produced Data, Consumed Data, Vendor ID, Product Type, Product Code, Major Rev, Minor Rev.

4.4.6 CC-Link Slave Parameters

Parameter	Meaning	Range of Value/Value
Interface		
Bus Startup	Communication start application controlled or automatic	Application controlled, Automatic (Default)
Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to 0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Off
I/O Data Status	Status of the input or the output data. For each input and output data the following status information (in Byte) is memorized in the dual-port memory: Status 0 = None (default) Status 1 = 1 Byte (for future use) Status 2 = 4 Byte (for future use)	None, (1 Byte, 4 Byte) Default: None
Ident		
Enabled	If ‘Enabled’ is unchecked, for the single Ident parameters each the respective default value is used.	
Vendor Code	Code for the Vendor	0 ... 65535 bzw. 0x00000000 ... 0x0000FFFF (hex), Hilscher: 0x00000352 (hex)
Model Type	Model type	0 ... 255, Default CIFX CC/CCS: 1, COMX 10XX-CCS/CCS: 3
Software Version	Software version	0 ... 63, Default: 2
Bus		
Station Address	Station address of CC-Link Slave Note: The number of occupied stations plus station address must not exceed the parameter range	1 ... 64, Default: 1
Enable Address Switch	Defines, if the station address is configured in the configuration software or at the address switch. If checked, the station address is configured at the address switch. The parameter ‘Enable Address Switch’ with the setting ‘checked’ can only be used for COMX 10XX-CCS/CCS.	Default: CIFX CC/CCS: unchecked COMX 10XX-CCS/CCS: checked
Baud Rate	Network transmission rate	156 kBaud (Default) 625 kBaud 2500 kBaud 5 MBaud 10 MBaud
Enable Baudrate Switch	Defines, if the baudrate is configured in the configuration software or at the baudrate switch. If checked, the baudrate is configured at the baudrate switch. The parameter ‘Enable Address Switch’ with the setting ‘checked’ can only be used for COMX 10XX-CCS/CCS.	Default: CIFX CC/CCS: unchecked COMX 10XX-CCS/CCS: checked
Data		
CC-Link Version	CC-Link Version 1 CC-Link Version 2	1 (Default) 2
Hold last received Output Data	Hold Clear Mode; Behavior in case of bus error Clear output data (unchecked) Hold last received output data (checked)	Default: unchecked
Station Type	Type of CC-Link station Remote I/O Station: Remote Device Station	 0 (Default) 1

Parameter	Meaning	Range of Value/Value
Number of Stations	Number of occupied stations Remote I/O Station: Remote Device Station:	 1 (Default) 1 ... 4
Extension Cycle	Number of extension cycles Allowed numbers for CC-Link version 1: Single/One cycle Allowed numbers for CC-Link version 2: Single/One cycle Double/Two cycles Quadruple/Four cycles Octuple/Eight cycles	 1 1 (Default) 2 4 8
IO-Data Bytes	The number of IO-Data bytes depends on the following settings: station type, number of stations and number extension cycles. The number of stations can only be configured with station type Remote Device Station version 1 and version 2 and the number of extension cycles can only be configured with version 2. Firmware/stack works according to CC-Link Version 2.0 input data output data Firmware/stack works according to CC-Link Version 1.11 input data output data	 12 … 368 Bytes 12 … 368 Bytes 4 … 48 Bytes 4 … 48 Bytes Default: 4

Table 25: CC-Link Slave Parameters

Note: To configure the Master, a CSP file (device description file) is required. The settings in the used Master must comply with the settings in the Slave to establish communication.

Important parameters are: Slave Station Address, Baudrate, Station Type and Vendor Code. For CC-Link Version 2.00 are important: number of cycles as well as number of extension cycles.

5 Diagnostic

5.1 ‘General’ Dialog

In the general **Diagnostic** dialog information regarding the device state and other general diagnosis parameters are displayed.

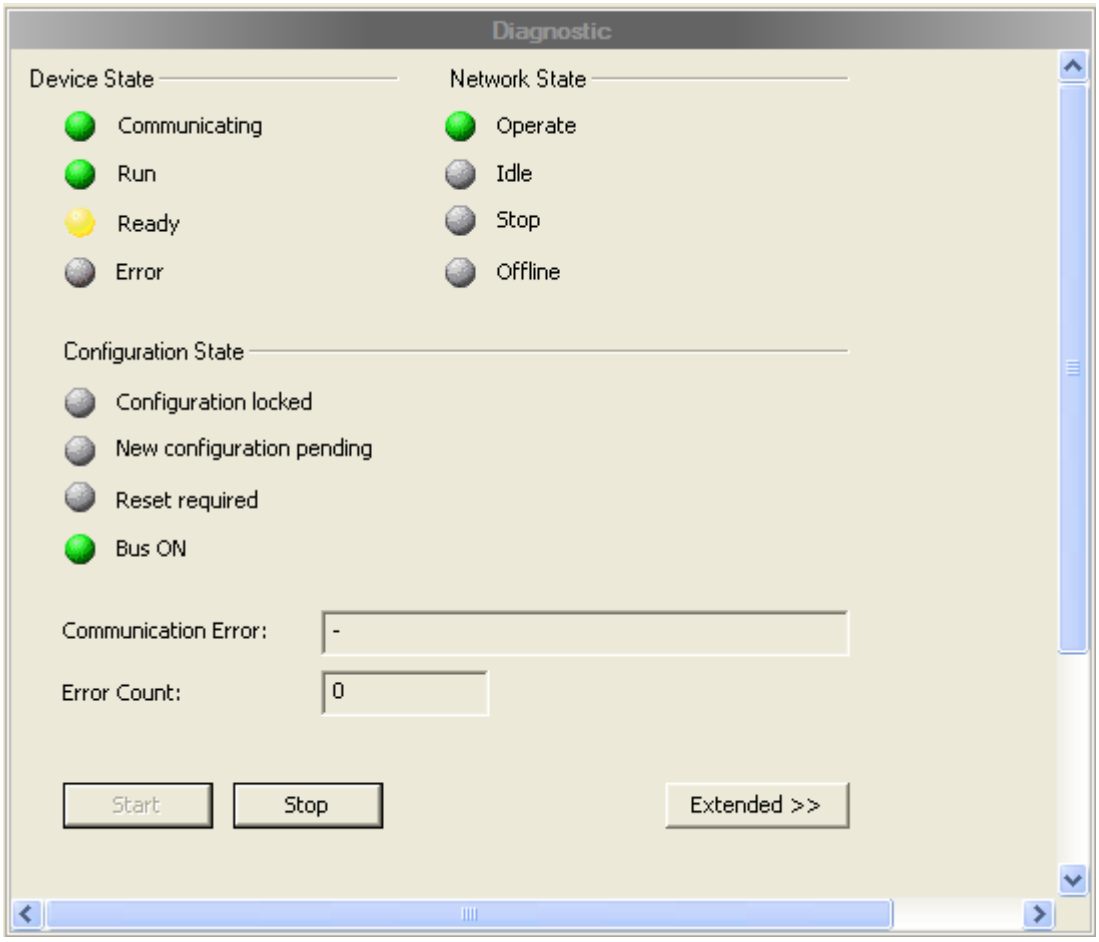


Figure 26: General Diagnostic

LED	Meaning	Color	State
Device State			
Communicating	Shows whether the Slave device executes the network communication.	(green)	COMMUNICATING
		(gray)	Not COMMUNICATING
Run	Shows whether the Slave device has been configured correctly.	(green)	Configuration OK
		(gray)	Configuration not OK
Ready	Shows whether the Slave device has been started correctly. The Slave device waits for a configuration.	(yellow)	Device READY
		(gray)	Device not READY
Error	Shows whether the Slave 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 Slave device is in data exchange. In a cyclic data exchange the input data or the output data of the Master are transmitted to the Slave.	(green)	In OPERATION state
		(gray)	Not in OPERATION state
Idle	Shows whether the Slave device is in idle state.	(yellow)	In IDLE state
		(gray)	Not in IDLE state
Stop	Shows whether the Slave device is in Stop state: There is no cyclic data exchange at the Slave network. The Slave 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 Slave 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 Slave 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 Slave 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 Slave 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 26: Diagnostic (General)

Parameter	Meaning
Communication Error	Indicates the name of the communication error. If the cause of error is resolved, the value is set to zero again.
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 if they were network related or caused internally.

Table 27: Parameter Diagnostic (General)

Start/Stop Communication	
Start	**Start** is enabled, if the communication has been stopped before or if the configuration requires this (for **Bus startup** ‚Application Controlled'). Start the the communication to the Master: ➢ Click **Start**. ➪ The device communicates at the bus.
Stop	**Stop** is enabled, if the communication has been started. Stop the the communication to the Master: ➢ Click **Stop**. ➪ Die Kommunikation des Gerätes am Bus wird gestoppt.
Extended >>	
Extended >>	Open the extended **Diagnostic**: ➢ Click **Extended >>**. ➪ The window extended **Diagnostic** is displayed.

Table 28: Start/Stop Communication, Extended >>

6 ‘Extended’ Diagnosis

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

6.1 ‘Extended’ Dialog

In the extended **Diagnostic** dialog under **Tasks Information** a task can be seleced to have access to the exteded diagnosis information of this task.

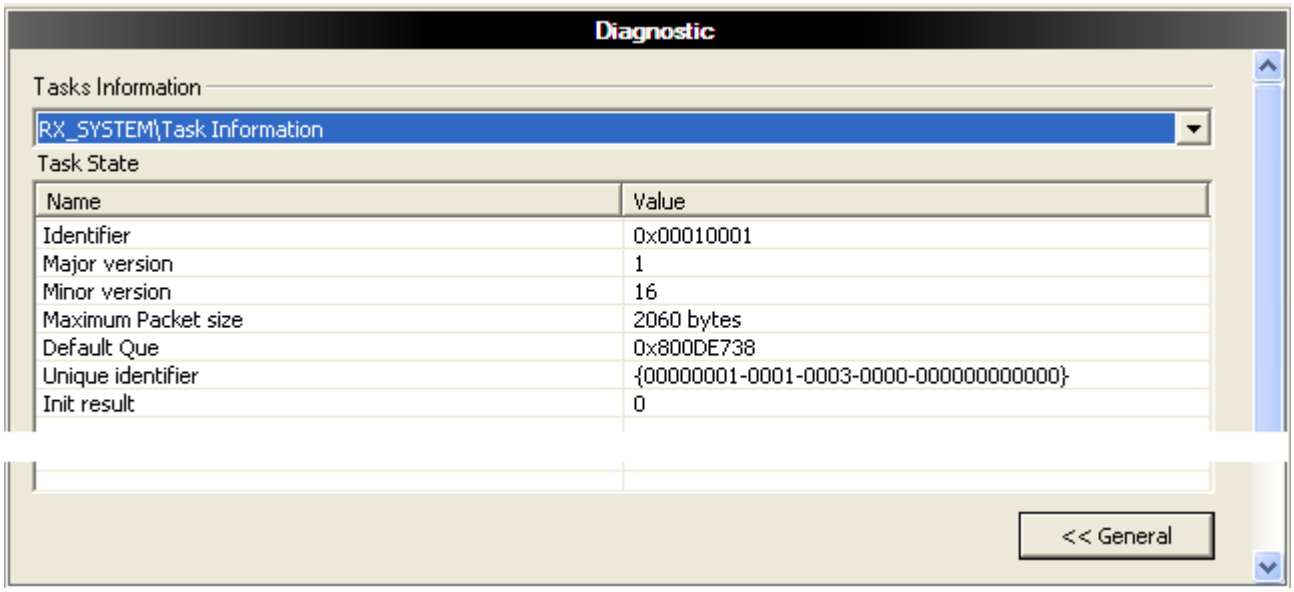


Figure 27: Example general Diagnostic

Tasks Information / Task State	
	Select Task: ➢ Click **Tasks Information**  . ➢ Select a task. ➧ The extended diagnosis information of this task are displayed in the **Task State** window. For the descriptions of the exteded diagnosis refer to the section *‘Extended’* ***Diagnosis*** on page 77.
<< General	
	Open the general **Diagnostic:** ➢ Click **<< General.** ➧ The window general **Diagnostic** is displayed.

Table 29: Tasks Information / Task State, << General

6.2 Overview

For an overview of the descriptions of the dialogue windows under **Extended Diagnosis** for every communication system refer to the following subsections.

Type of System	Subsection/ Communication System	Manual Page
Real-Time-Ethernet System	*EtherCAT Slave*	79
	EtherNet/IP Adapter	83
	Open Modbus/TCP	91
	PROFINET IO Device	96
	POWERLINK Controlled Node	101
	sercos	102
	VARAN Client (Slave)	103
Fieldbus System	*PROFIBUS DP Slave*	110
	PROFIBUS MPI	117
	CC-Link Slave	122
	CompoNet Slave	140
	CANopen Slave	144
	DeviceNet Slave	160

Table 30: Overview Extended Diagnosis

For more information about the parameters in the dialogue windows **Extended Diagnosis** refer to the descriptions of the configuration parameters in section *Configuration Parameters Real-Time Ethernet Systems* beginning from page 42, in section *Configuration Parameters Fieldbus Systems* beginning from page 62, as well as to the corresponding user manual of your device, in the section *Technical Data of the Communication Protocols*. This manual is located on the included DVD to your device.

6.3 EtherCAT Slave

6.3.1 Overview EtherCAT Slave

Here you find an overview of the **Extended Diagnosis** for *EtherCAT Slave*.

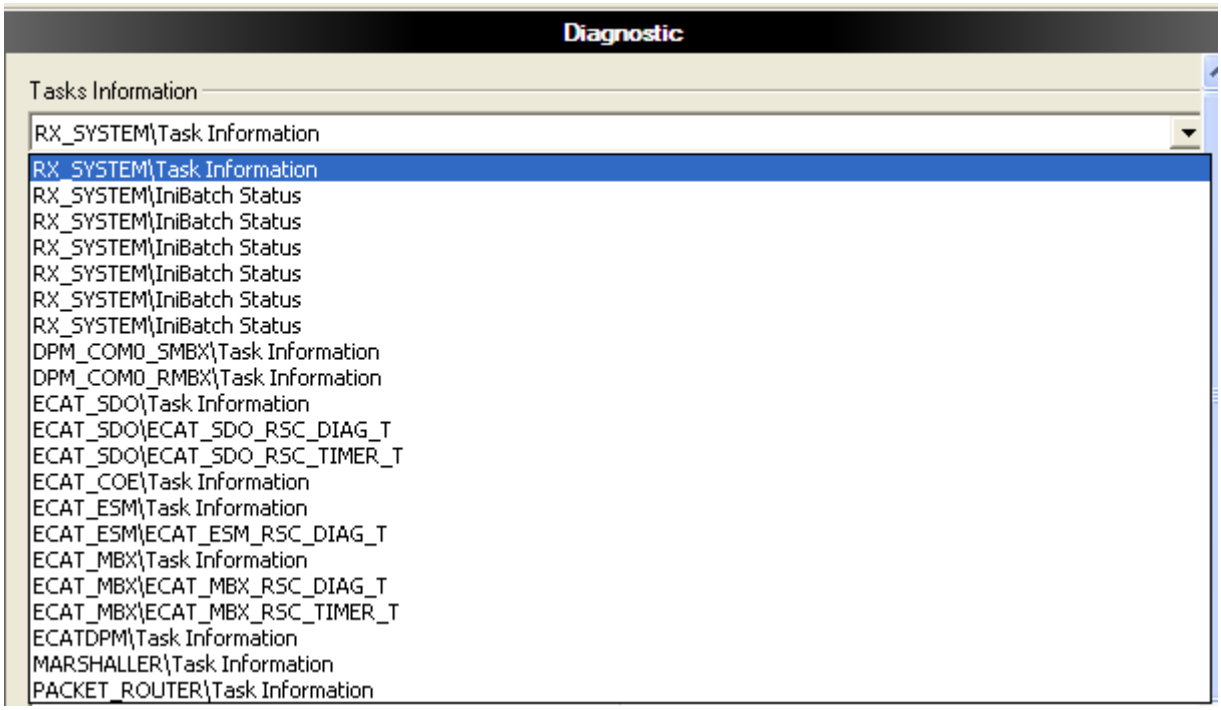


Figure 28: Extended Diagnosis EtherCAT Slave (cifX, comX and netJACK, Example cifX)

Task Information Group	Taks Information	Manual Page
ECAT_SDO	*ECAT_SDO\ECAT_SDO_RSC_DIAG_T*	*80*
	ECAT_SDO\ECAT_SDO_RSC_TIMER_T	*80*
ECAT_ESM	*ECAT_ESM\ECAT_ESM_RSC_DIAG_T*	*81*
ECAT_MBX	*ECAT_MBX\ECAT_MBX_RSC_DIAG_T*	*81*
	ECAT_MBX\ECAT_MBX_RSC_TIMER_T	*82*

Table 31: Descriptions of the Dialog Panes Extended Diagnosis EtherCAT Slave

For the description of further tasks refer to section *Descriptions for Tasks with similar Functions* on page 166.

6.3.2 ECAT_SDO\ECAT_SDO_RSC_DIAG_T

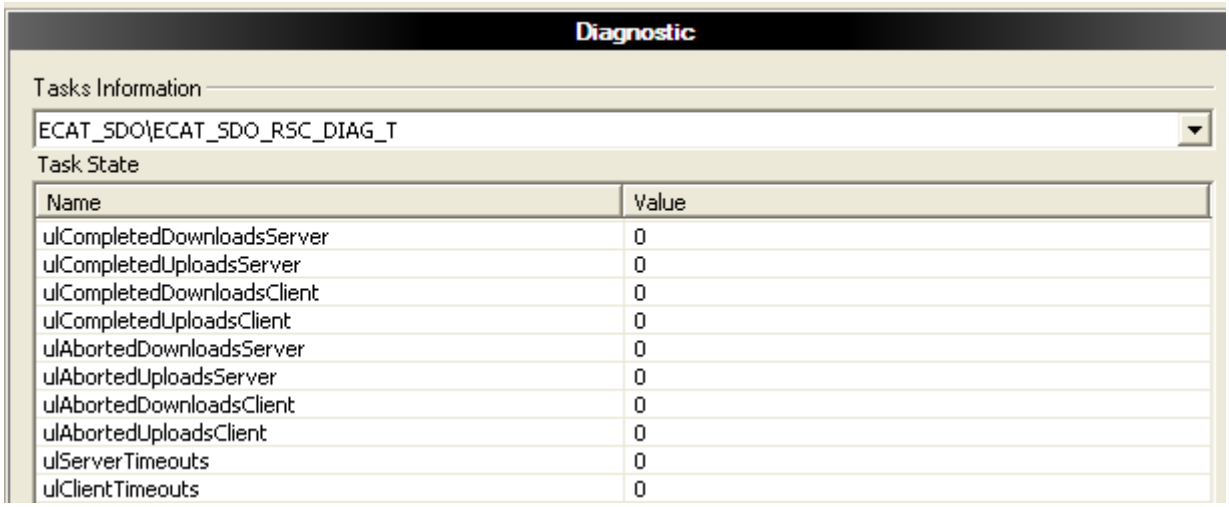


Figure 29: Extended Diagnosis > ECAT_SDO > ECAT_SDO_RSC_DIAG_T

Name	Description
[*Service*]	Diagnosis counter of the ECAT_SDO_RSC_DIAG_T layer. Indicates the services processed. (The services of the single packets are described in the EtherCAT Slave Protocol API manual [9].)

Table 32: Extended Diagnosis > ECAT_SDO > ECAT_SDO_RSC_DIAG_T

6.3.3 ECAT_SDO\ECAT_SDO_RSC_TIMER_T

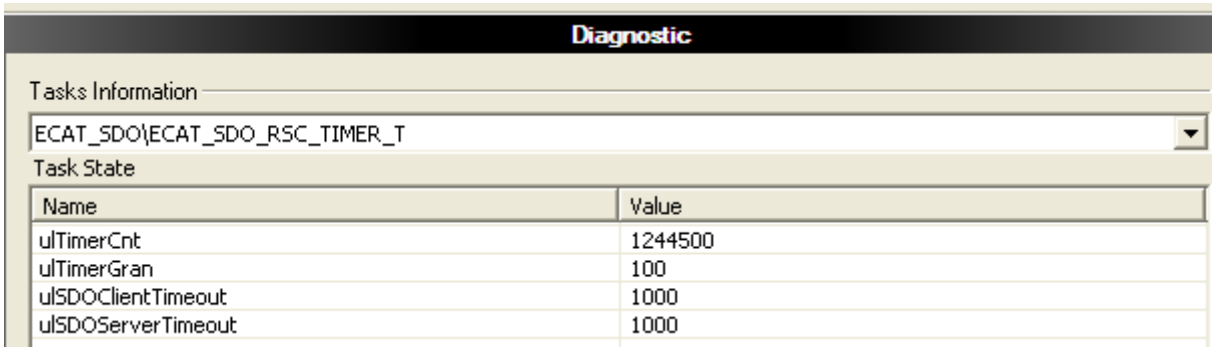


Figure 30: Extended Diagnosis > ECAT_SDO > ECAT_SDO_RSC_TIMER_T

Name	Description
ulTimerCnt	Timer Count
ultimerGran	Granularity of the timer
ulSDOClientTimeout	Clients
ulSDOServerTimeout	Timeout value of the SDO Server

Table 33: Extended Diagnosis > ECAT_SDO > ECAT_SDO_RSC_TIMER_T

6.3.4 ECAT_ESM\ECAT_ESM_RSC_DIAG_T

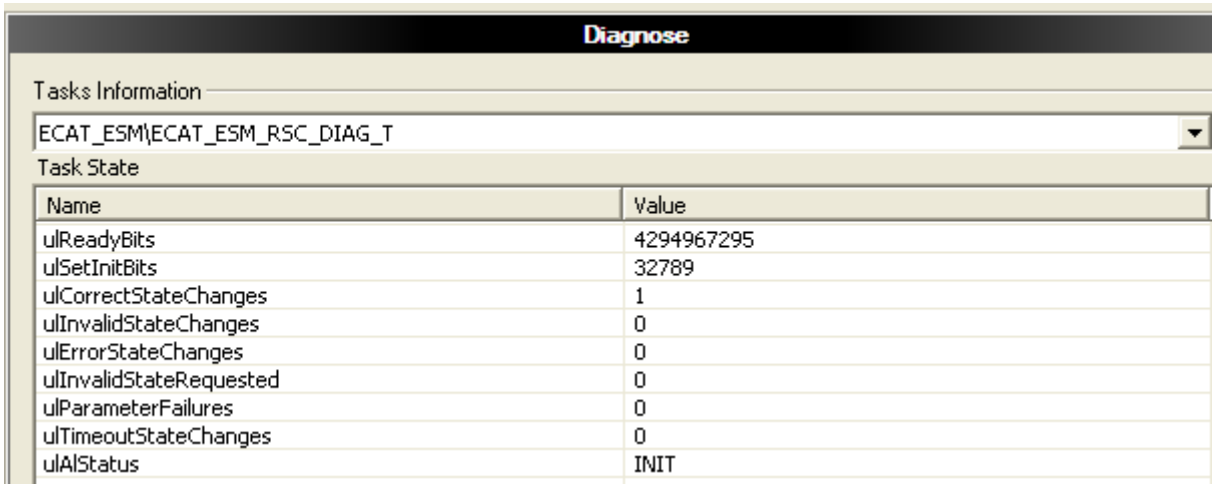


Figure 31: Extended Diagnosis > ECAT_ESM > ECAT_ESM_RSC_DIAG_T

Name	Description
ulReadyBits	Number of Ready Bits
ulSetInitBits	Number of SetInit Bits
[*Service*]	Diagnosis counter of the ECAT_ESM_RSC_DIAG_T layer. Indicates the services processed. (The services of the single packets are described in the EtherCAT Slave Protocol API manual [9].)
ulAlStatus	AL status, for more detailed information, see the EtherCAT Specification, Part 5 and 6

Table 34: Extended Diagnosis > ECAT_ESM > ECAT_ESM_RSC_DIAG_T

6.3.5 ECAT_MBX\ECAT_MBX_RSC_DIAG_T

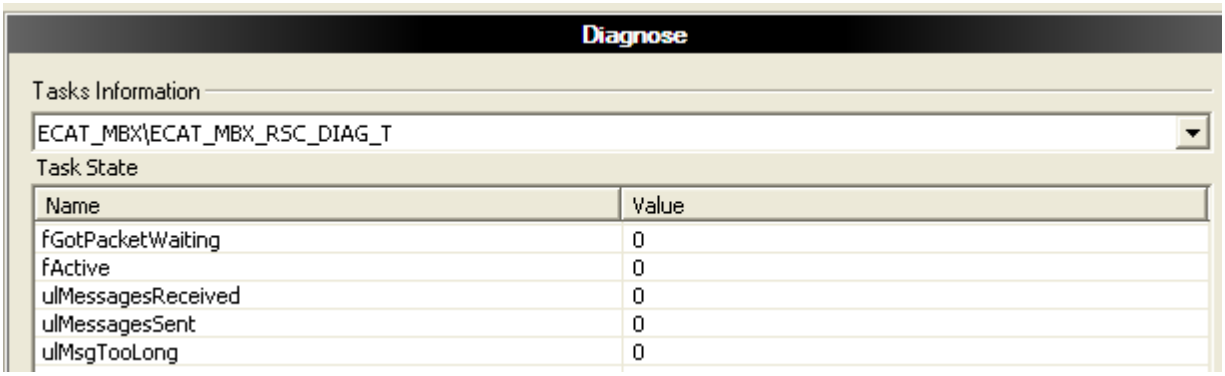


Figure 32: Extended Diagnosis > ECAT_MBX > ECAT_MBX_RSC_DIAG_T

Name	Description
[*Service*]	Diagnosis counter of the ECAT_MBX_RSC_DIAG_T layer. Indicates the services processed. (The services of the single packets are described in the EtherCAT Slave Protocol API manual [9].)

Table 35: Extended Diagnosis > ECAT_MBX > ECAT_MBX_RSC_DIAG_T

6.3.6 ECAT_MBX\ECAT_MBX_RSC_TIMER_T

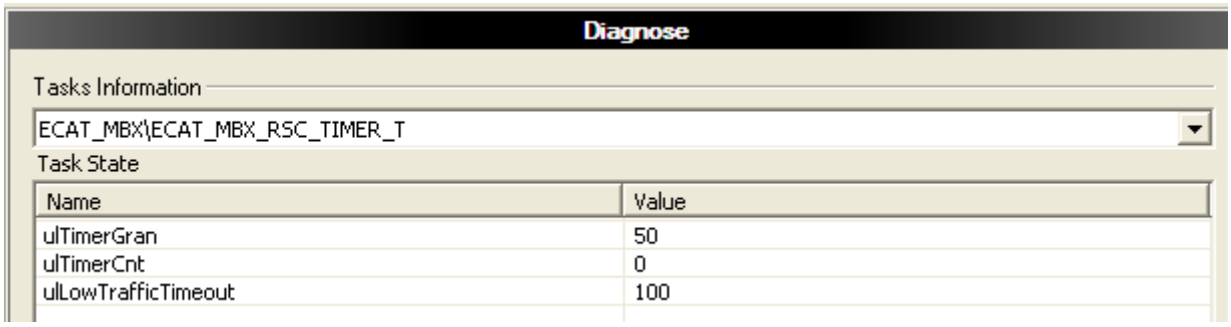


Figure 33: Extended Diagnosis > ECAT_MBX > ECAT_MBX_RSC_TIMER_T

Name	Description
ulTimerCnt	Timer Count
ulTimerGran	Granularity of the timer
ulLowTrafficTimeout	Timeout value for low network data traffic

Table 36: Extended Diagnosis > ECAT_MBX > ECAT_MBX_RSC_TIMER_T

6.4 EtherNet/IP Adapter

6.4.1 Overview EtherNet/IP Adapter

Here you find an overview of the **Extended Diagnosis** for *EtherNet/IP Adapter.*.

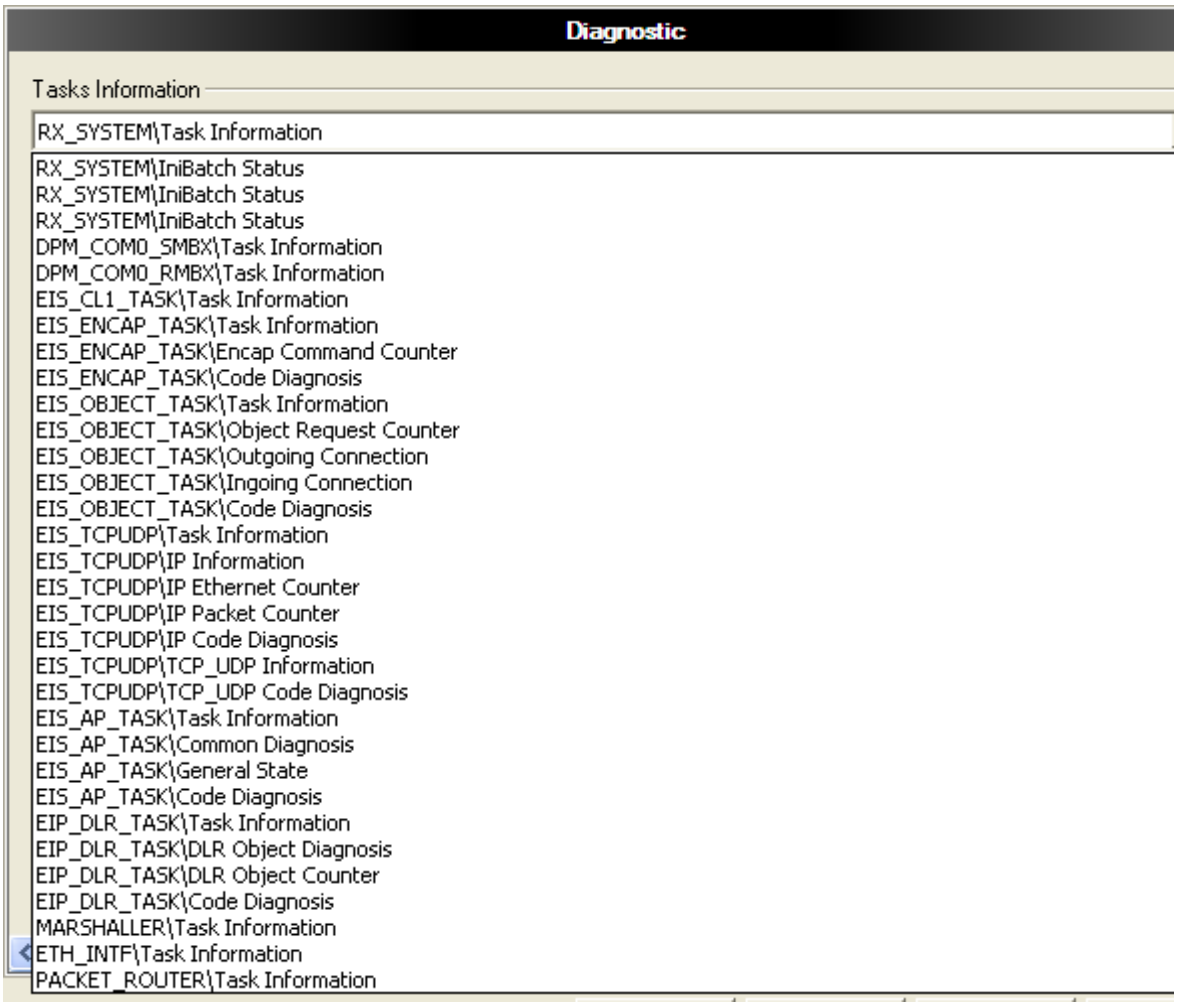


Figure 34: Extended Diagnosis EtherNet/IP Adapter (cifX, comX and netJACK, Example cifX)

Task Information Group	Taks Information	Manual Page
APS_TASK	*APS_TASK/Common Diagnostic*	84
	APS_TASK/General State	84
OBJECT_TASK	*OBJECT_TASK/Object Request Counter*	85
	OBJECT_TASK/Outgoing Connection	86
	OBJECT_TASK/Ingoing Connection	86
ENCAP_TASK	*ENCAP_TASK/Encap Command Counter*	87
EIS_DLR-TASK	*EIS_DLR_TASK/DLR Object Diagnosis*	88
	EIS_DLR_TASK/DLR Object Counter	90

Table 37: Descriptions of the Dialog Panes Extended Diagnosis EtherNet/IP Adapter.

For the description of further tasks refer to section *Descriptions for Tasks with similar Functions* on page 166.

6.4.2 APS_TASK/Common Diagnostic

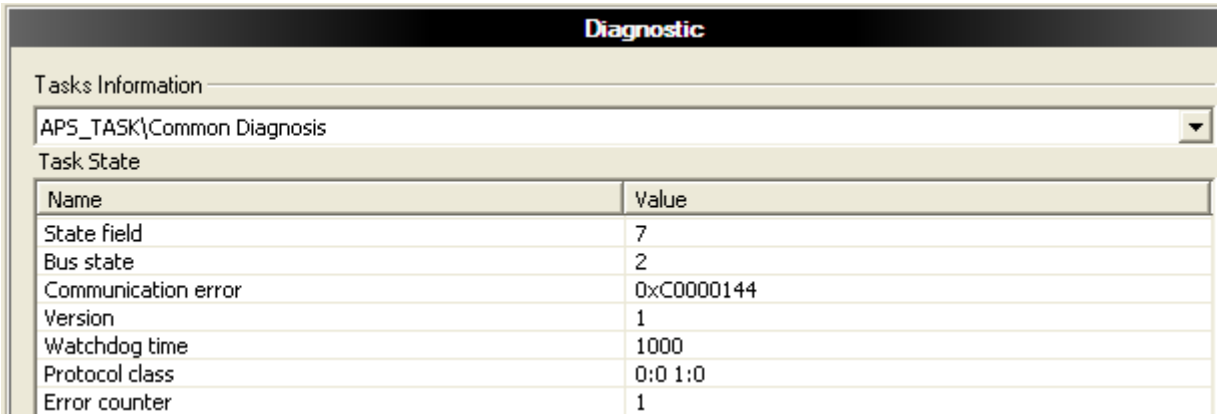


Figure 35: Extended Diagnosis > APS_TASK > Common Diagnostic

Name	Description
Status field	Communication change of state
Bus state	Communication state
Communication Error	Unique error number according to protocol stack (not supported yet)
Version	Version number of this diagnosis structure
Watchdog time	Configured watchdog time
Protocol Class	Protocol class: MASTER, SLAVE, CLIENT, SERVER, GATEWAY
Error Counter	Total number of detected error since power-up or reset

Table 38: Extended Diagnosis > APS_TASK > Common Diagnostic

6.4.3 APS_TASK/General State

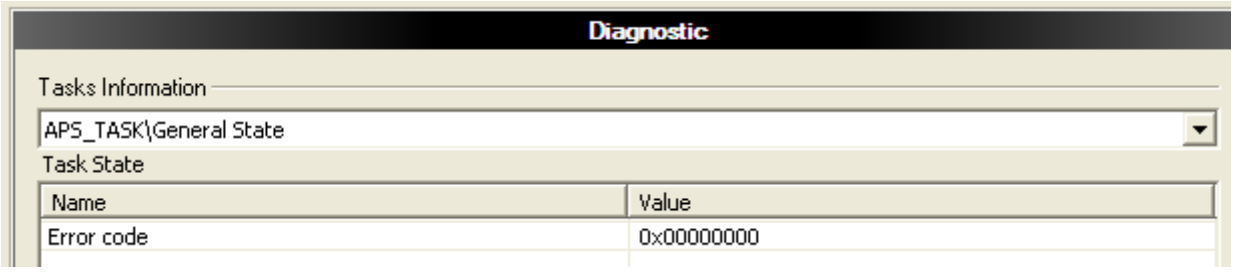
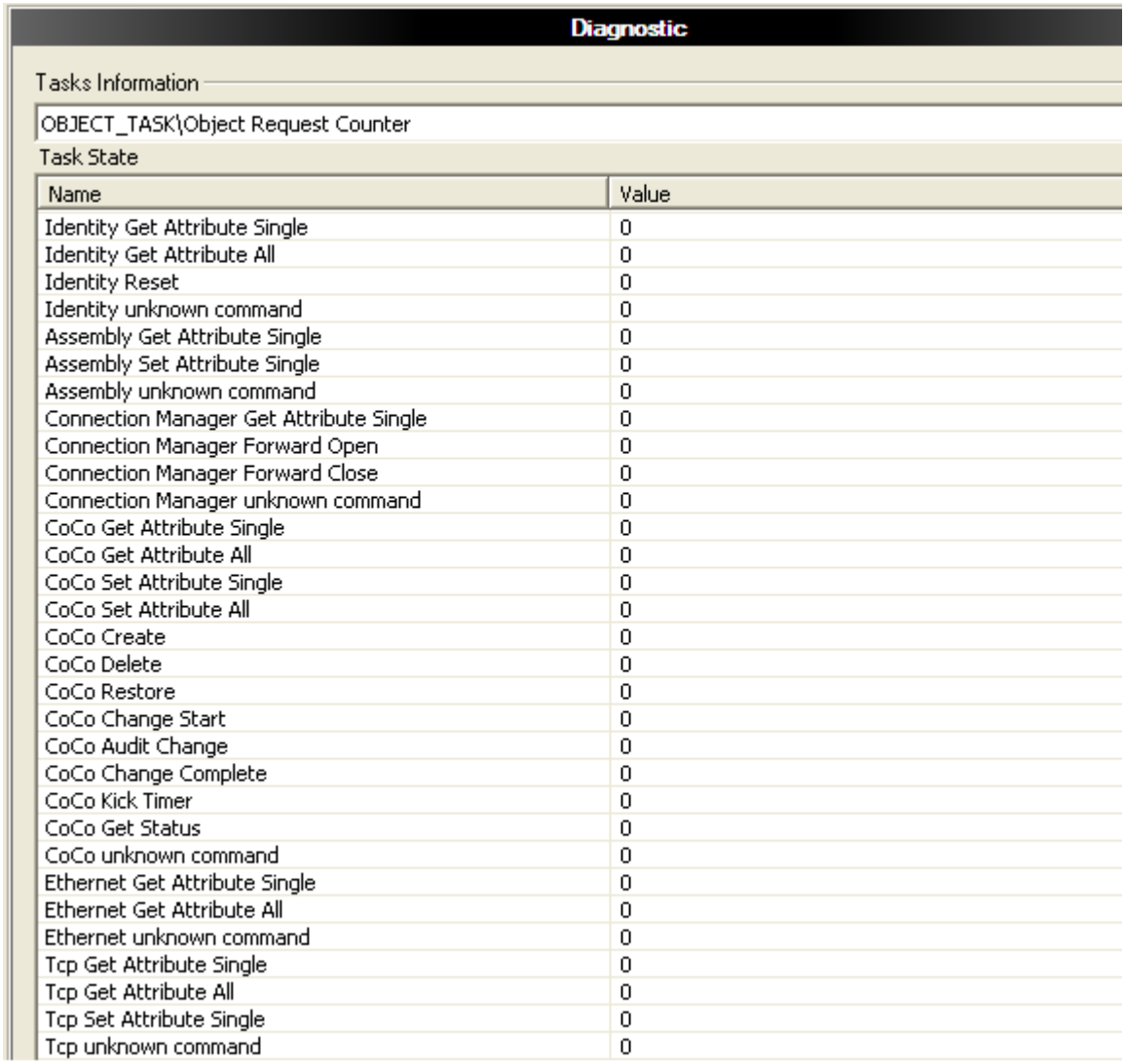


Figure 36: Extended Diagnosis > APS_TASK > General State

Name	Description
Error Code	Code of the last occurred error

Table 39: Extended Diagnosis > APS_TASK > General State

6.4.4 OBJECT_TASK/Object Request Counter



Name	Value
Identity Get Attribute Single	0
Identity Get Attribute All	0
Identity Reset	0
Identity unknown command	0
Assembly Get Attribute Single	0
Assembly Set Attribute Single	0
Assembly unknown command	0
Connection Manager Get Attribute Single	0
Connection Manager Forward Open	0
Connection Manager Forward Close	0
Connection Manager unknown command	0
CoCo Get Attribute Single	0
CoCo Get Attribute All	0
CoCo Set Attribute Single	0
CoCo Set Attribute All	0
CoCo Create	0
CoCo Delete	0
CoCo Restore	0
CoCo Change Start	0
CoCo Audit Change	0
CoCo Change Complete	0
CoCo Kick Timer	0
CoCo Get Status	0
CoCo unknown command	0
Ethernet Get Attribute Single	0
Ethernet Get Attribute All	0
Ethernet unknown command	0
Tcp Get Attribute Single	0
Tcp Get Attribute All	0
Tcp Set Attribute Single	0
Tcp unknown command	0

Figure 37: Extended Diagnosis > OBJECT_TASK > Object Request Counter

The **Object Request Counter** table indicates the respective number of services sent to different objects.

Name	Description
Identity Get Attribute Single	Counter for the *Get Attribute Single* services to the *identity* object
Identity Get Attribute All	Counter for the *Get Attribute All* services to the *identity* object
Identity Reset	Counter for the *Reset* services to the *identity* object
Identity unknown Command	Counter for the *Unknown Command* services to the *identity* object
Assembly Get Attribute Single	Counter for the *Get Attribute Single* services to the *assembly* object
[*Object*] [Service]	Counter for the [*Object*] services to the [*Service*] object

Table 40: Extended Diagnosis > OBJECT_TASK > Object Request Counter

6.4.5 OBJECT_TASK/Outgoing Connection

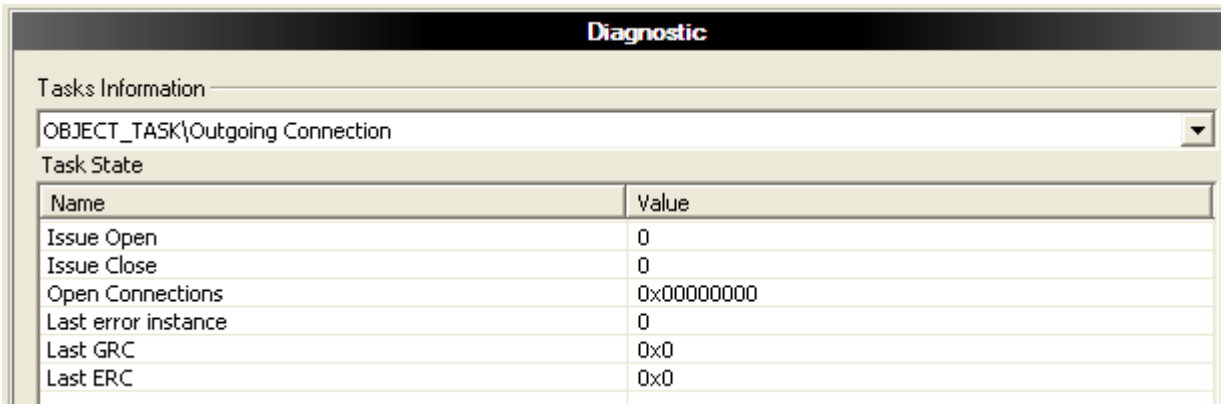


Figure 38: Extended Diagnosis > OBJECT_TASK > Outgoing Connection

Name	Description
Issue Open	Counter for open issues
Issue Close	Counter for closed issues
Open Connections	Actual open connection
Last error instance	Last error instance
Last GRC	Last General Status Code
Last ERC	Last Extended Status Code

Table 41: Extended Diagnosis > OBJECT_TASK > Outgoing Connection

6.4.6 OBJECT_TASK/Ingoing Connection

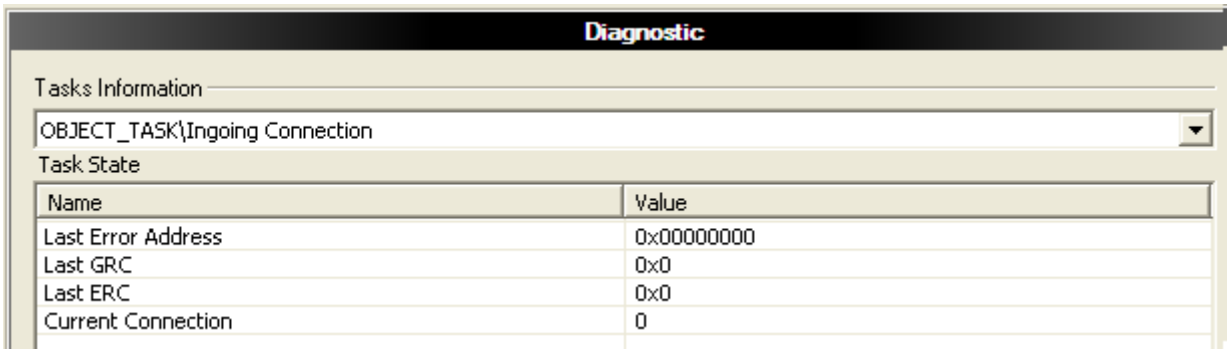


Figure 39: Extended Diagnosis > OBJECT_TASK > Ingoing Connection

Name	Description
Last Errors Address	IP Address
Last GRC	Last General Status Code
Last ERC	Last Extended Status Code
Actual Connection	Actual open connection

Table 42: Extended Diagnosis > OBJECT_TASK > Ingoing Connection

6.4.7 ENCAP_TASK/Encap Command Counter

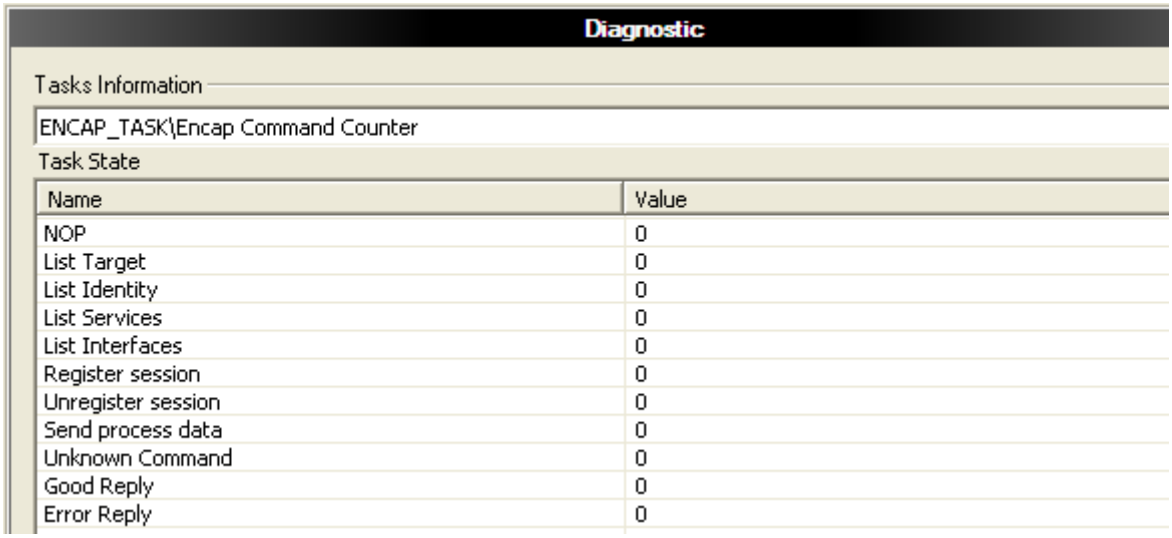
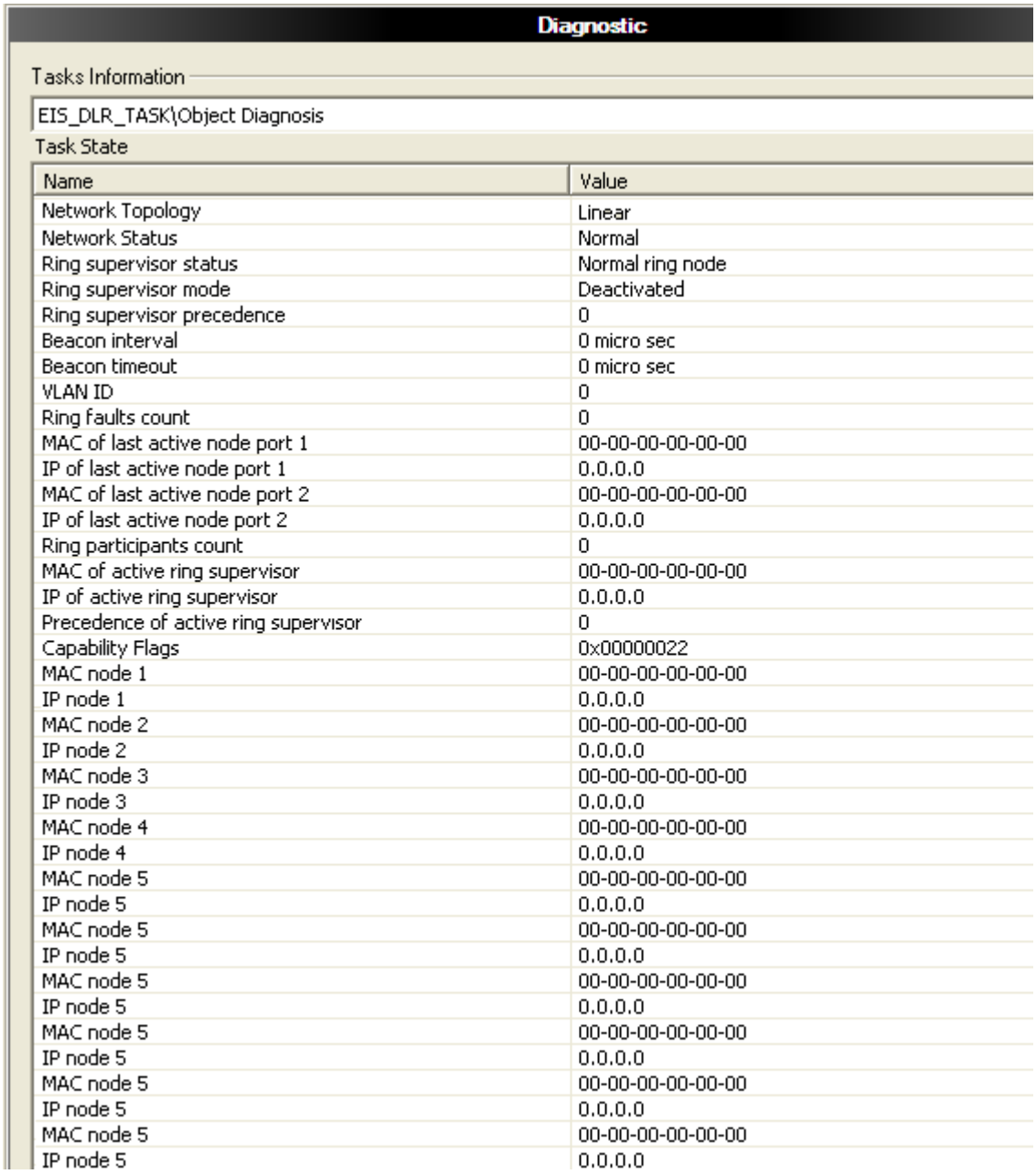


Figure 40: Extended Diagnosis > ENCAP_TASK > Encap Command Counter

Name	Description
NOP	Counter incoming encapsulation telegrams
List Target	Counter incoming encapsulation telegrams
List Identity	Counter incoming encapsulation telegrams
List Services	Counter incoming encapsulation telegrams
List Interfaces	Counter incoming encapsulation telegrams
Register Session	Counter incoming encapsulation telegrams
Unregister Session	Counter incoming encapsulation telegrams
Process data	Unit Date RR Data telegrams
Unknown Command	Counter incoming encapsulation telegrams
Good Reply	Counter incoming encapsulation telegrams
Error Reply	Counter incoming encapsulation telegrams

Table 43: Extended Diagnosis > ENCAP_TASK > Encap Command Counter

6.4.8 EIS_DLR_TASK/DLR Object Diagnosis



Name	Value
Network Topology	Linear
Network Status	Normal
Ring supervisor status	Normal ring node
Ring supervisor mode	Deactivated
Ring supervisor precedence	0
Beacon interval	0 micro sec
Beacon timeout	0 micro sec
VLAN ID	0
Ring faults count	0
MAC of last active node port 1	00-00-00-00-00-00
IP of last active node port 1	0.0.0.0
MAC of last active node port 2	00-00-00-00-00-00
IP of last active node port 2	0.0.0.0
Ring participants count	0
MAC of active ring supervisor	00-00-00-00-00-00
IP of active ring supervisor	0.0.0.0
Precedence of active ring supervisor	0
Capability Flags	0x00000022
MAC node 1	00-00-00-00-00-00
IP node 1	0.0.0.0
MAC node 2	00-00-00-00-00-00
IP node 2	0.0.0.0
MAC node 3	00-00-00-00-00-00
IP node 3	0.0.0.0
MAC node 4	00-00-00-00-00-00
IP node 4	0.0.0.0
MAC node 5	00-00-00-00-00-00
IP node 5	0.0.0.0
MAC node 5	00-00-00-00-00-00
IP node 5	0.0.0.0
MAC node 5	00-00-00-00-00-00
IP node 5	0.0.0.0
MAC node 5	00-00-00-00-00-00
IP node 5	0.0.0.0
MAC node 5	00-00-00-00-00-00
IP node 5	0.0.0.0
MAC node 5	00-00-00-00-00-00
IP node 5	0.0.0.0

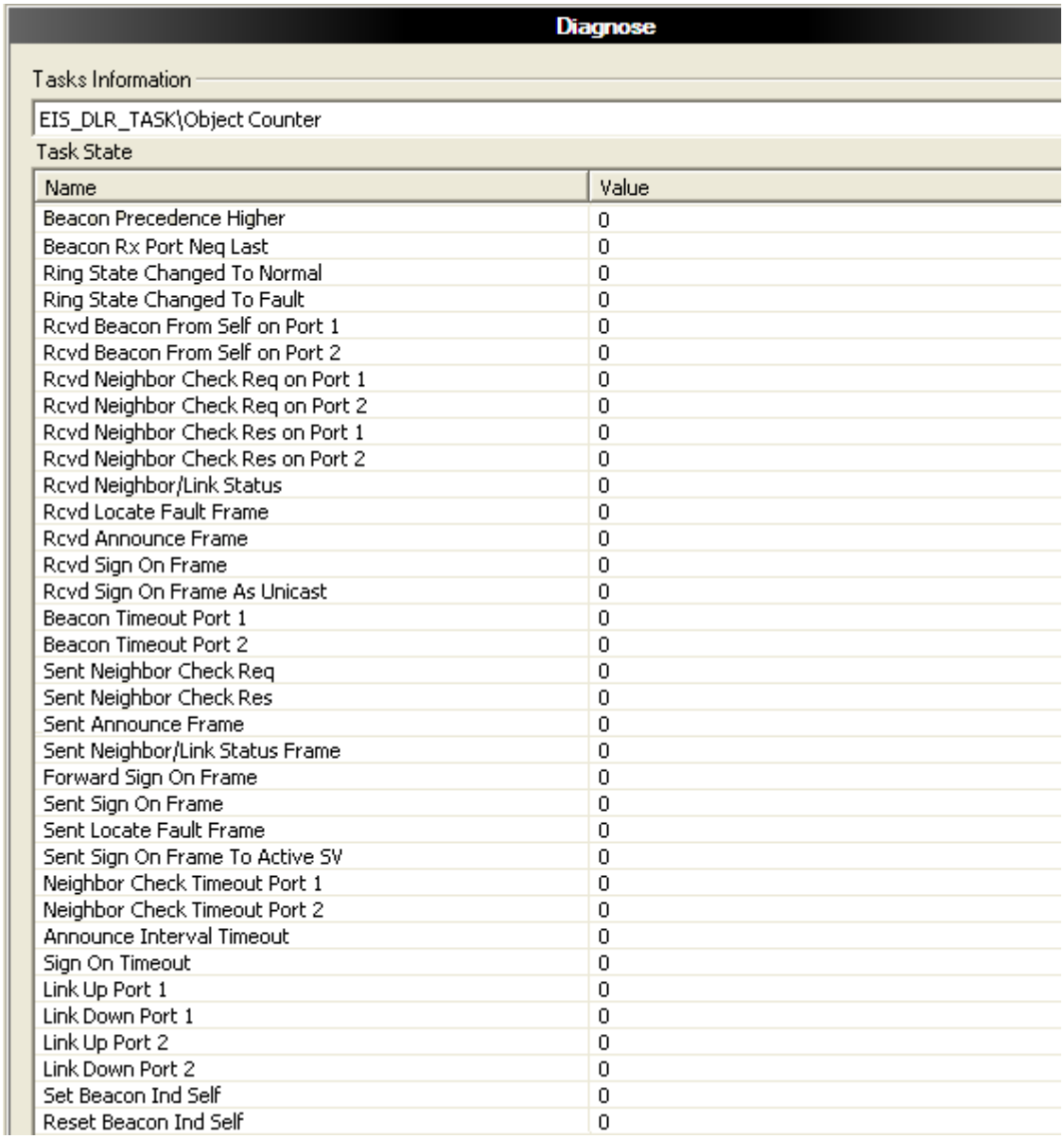
Figure 41: Extended Diagnosis > EIS_DLR_TASK > DLR Object Diagnosis

Name	Description
Network Topology	Possible Values: Linear or Ring
Network Status	Possible Values: Normal, Ring fault, Unexpected loop detected, Partial network fault, Rapid fault/Restore cycle
Ring Supervisor Status	Possible Values: Backup, Active ring supervisor, Normal ring node, None DLR device, Current supervisor parameters not supported
Ring Supervisor Mode	Possible Values: Deactivated, Activated
Ring Supervisor Precedence	Precedence value assigned to a ring supervisor, and transmitted in Beacon frames (further see www.odva.org)
Beacon-Interval (Mikro-Sek)	Ring interval at which the supervisor sends beacon frames (further see www.odva.org)
Beacon-Timeout (Mikro-Sek)	Amount of time nodes will wait before timing out reception of Beacon frames and taking the appropriate action (depending on weather supervisor or normal ring node). (further see www.odva.org)

Name	Description
VLAN-ID	VLAN ID used when sending DLR protocol frames.
Ring faults count	Counts Ring faults
MAC of last active node port 1	MAC Address of last active node at port 1
IP of last active node port 1	IP Address of last active node at port 1
MAC of last active node port 2	MAC Address of last active node at port 2
IP of last active node port 2	IP Address of last active node at port 2
Ring participants count	Ring participants count
MAC of active ring supervisor	MAC Address of active ring supervisor
IP of active ring supervisor	IP Address of active ring supervisor
Precedence of active ring supervisor	The precedence the supervisor has in the ring.
Capability Flags	Indicates which capabilities the supervisor has.
MAC Node 1	MAC Address Node 1
IP Node 1	IP Address Node 1
to	
MAC Node 10	MAC Address Node 2
IP Node 10	IP Address Node 2

Table 44: Extended Diagnosis > EIS_DLR_TASK > DLR Object Diagnosis

6.4.9 EIS_DLR_TASK/DLR Object Counter



Name	Value
Beacon Precedence Higher	0
Beacon Rx Port Neq Last	0
Ring State Changed To Normal	0
Ring State Changed To Fault	0
Rcvd Beacon From Self on Port 1	0
Rcvd Beacon From Self on Port 2	0
Rcvd Neighbor Check Req on Port 1	0
Rcvd Neighbor Check Req on Port 2	0
Rcvd Neighbor Check Res on Port 1	0
Rcvd Neighbor Check Res on Port 2	0
Rcvd Neighbor/Link Status	0
Rcvd Locate Fault Frame	0
Rcvd Announce Frame	0
Rcvd Sign On Frame	0
Rcvd Sign On Frame As Unicast	0
Beacon Timeout Port 1	0
Beacon Timeout Port 2	0
Sent Neighbor Check Req	0
Sent Neighbor Check Res	0
Sent Announce Frame	0
Sent Neighbor/Link Status Frame	0
Forward Sign On Frame	0
Sent Sign On Frame	0
Sent Locate Fault Frame	0
Sent Sign On Frame To Active SV	0
Neighbor Check Timeout Port 1	0
Neighbor Check Timeout Port 2	0
Announce Interval Timeout	0
Sign On Timeout	0
Link Up Port 1	0
Link Down Port 1	0
Link Up Port 2	0
Link Down Port 2	0
Set Beacon Ind Self	0
Reset Beacon Ind Self	0

Figure 42: Extended Diagnosis > EIS_DLR_TASK > DLR Object Counter

Name	Description
[*Service*]	Diagnosis counter: Indicates the services processed. (The services of the single packets are described in the EtherNet/IP Slave Protocol API manual [2] [2].)

Table 45: Extended Diagnosis > EIS_DLR_TASK > DLR Object Counter

6.5 Open Modbus/TCP

6.5.1 Overview Open Modbus/TCP

Here you find an overview of the **Extended Diagnosis** for *Open Modbus/TCP*.

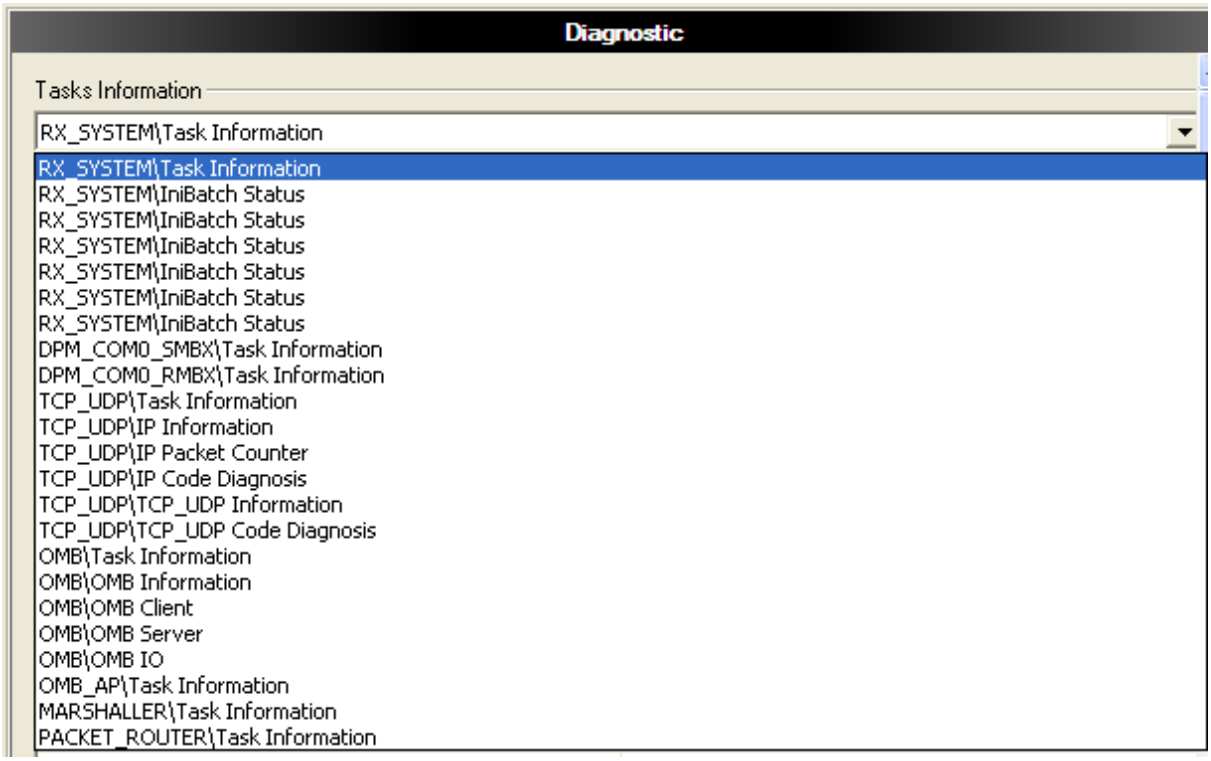


Figure 43: Extended Diagnosis Open Modbus/TCP (cifX, comX and netJACK, Example cifX)

Task Information Group	Taks Information	Manual Page
OMB	*OMB/OMB Information*	**92**
	OMB/OMB Client	93
	OMB/OMB Server	94
	OMB/OMB IO	95

Table 46: Descriptions of the Dialog Panes Extended Diagnosis Open Modbus/TCP (cifX, comX and netJACK, Example cifX)

For the description of further tasks refer to section *Descriptions for Tasks with similar Functions* on page 166.

6.5.2 OMB/OMB Information

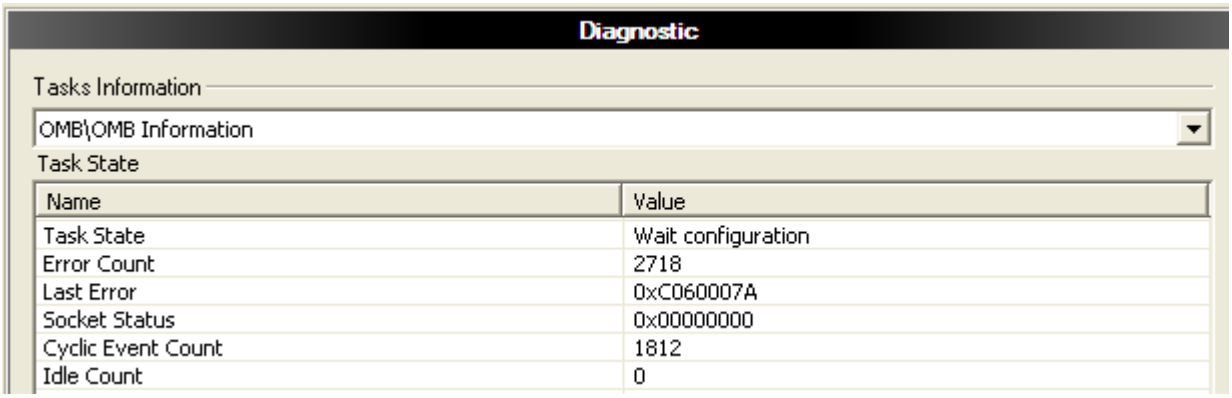


Figure 44: Extended Diagnosis > OMB > OMB Information

Name	Description
Task State	Actual state of the protocol process: 0 = Task not initialized 1 = Task is running 2 = Task initialized 3 = Initialization error 4 = Waiting for configuration
Error Count	Counter for errors
Last Error	Last occurred error (Description see appropriate manual)
Socket Status	The socket status informs about the TCP sockets. More exactly, it contains the information whether sockets are open or closed in a bit-coded manner. The coding is as follows: The socket number corresponds to the position of the bit within the variable, i.e. bit 0 represents socket # 0 and bit 15 represents socket # 15. The coding is in that way that a bit value of 1 means the respective socket is open and a value of 0 indicates it is not open.
Cyclic Event Count	The cyclic event counter represents the number of cyclic events that have occurred.
Idle Count	The idle count is currently not used.

Table 47: Extended Diagnosis > OMB > OMB Information

6.5.3 OMB/OMB Client

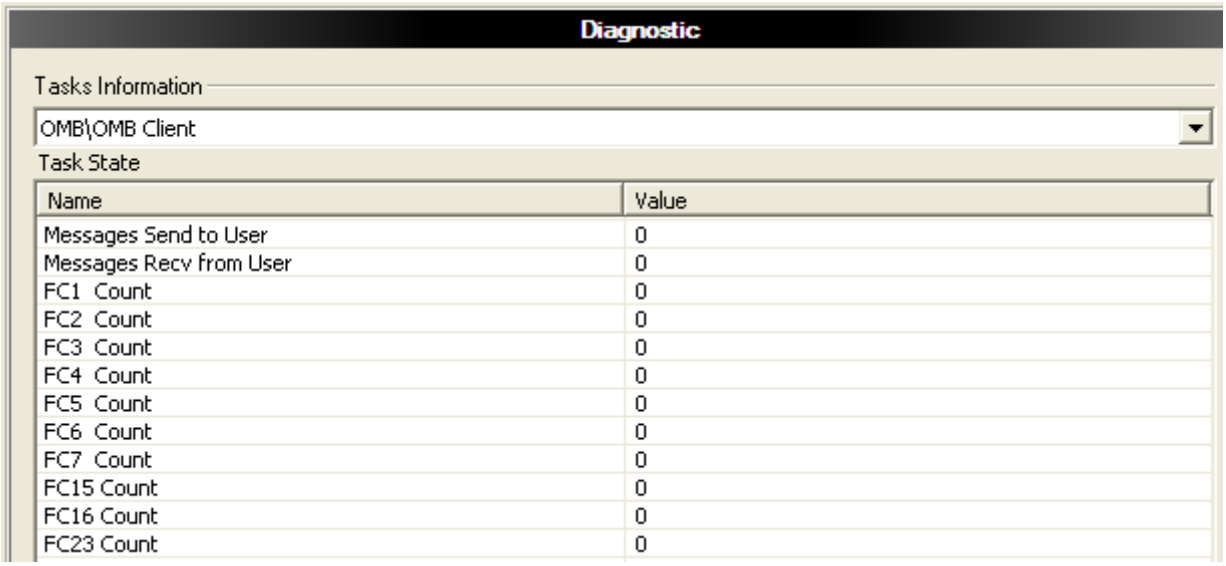


Figure 45: Extended Diagnosis > OMB > OMB Client

Name	Description
Messages Send to User	Number of messages send from the OMB client to the user
Messages Recv from User	Number of messages received from the user
FC[*N*] Count	Number of access events to the respective function code FC = Function code Possible values for *N* = 0 1, 2, 3, 4, 5, 6, 7, 15, 16, 23

Table 48: Extended Diagnosis > OMB > OMB Client

6.5.4 OMB/OMB Server

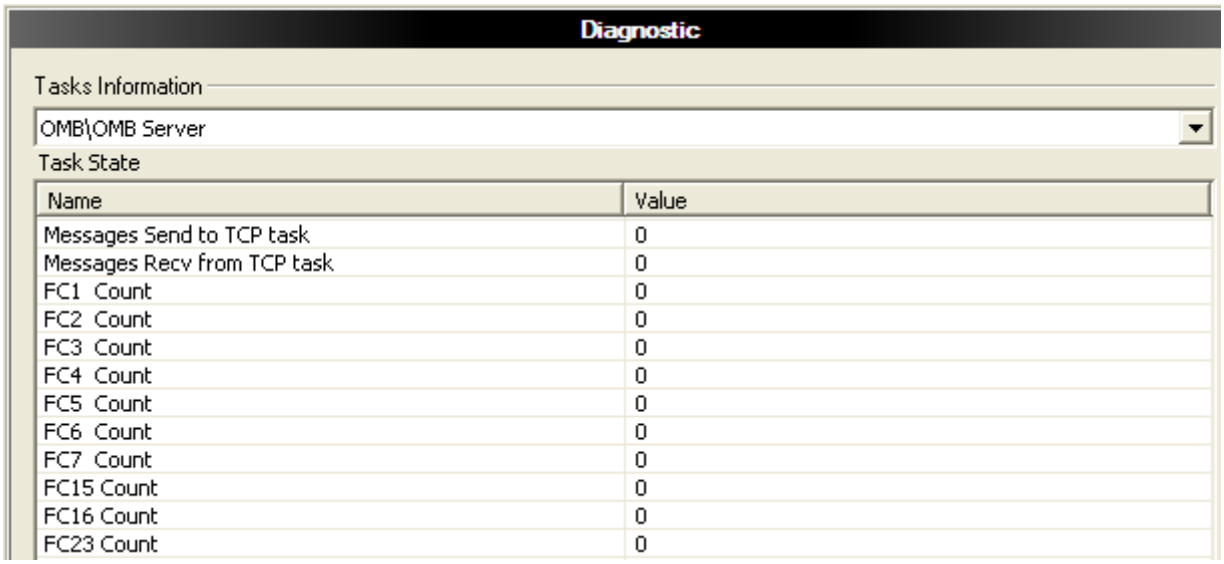


Figure 46: Extended Diagnosis > OMB > OMB Server

Name	Description
Messages Send to TCP task	Number of messages send from the OMB server to the TCP task
Messages Recv from TCP task	Number of messages received from the TCP task
FC[*N*] Count	Number of access events to the respective function code FC = Function code Possible values for *N* = 0 1, 2, 3, 4, 5, 6, 7, 15, 16, 23

Table 49: Extended Diagnosis > OMB > OMB Server

6.5.5 OMB/OMB IO

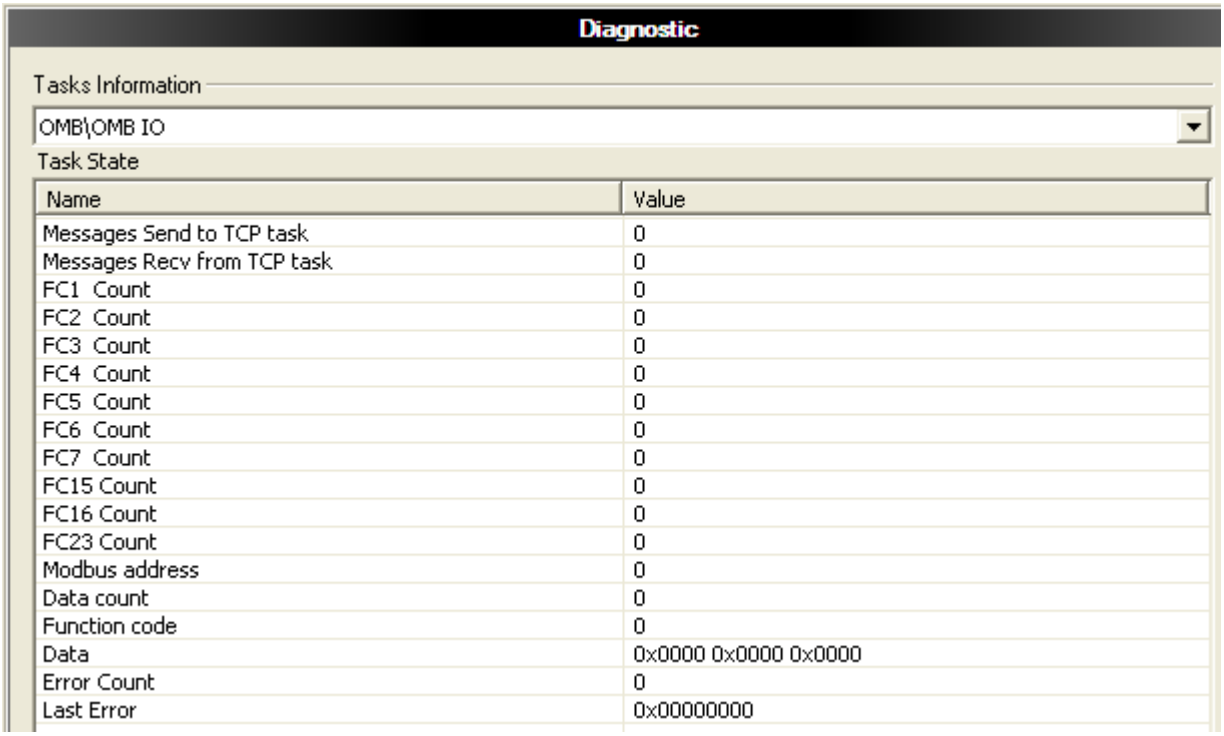


Figure 47: Extended Diagnosis > OMB > OMB IO

Name	Description
Messages Send to TCP task	Number of messages send from the OMB server to the TCP task
Messages Recv from TCP task	Number of messages received from the TCP task
FC[*N*] Count	Number of access events to the respective function code FC = Function code Possible values for *N* = 0 1, 2, 3, 4, 5, 6, 7, 15, 16, 23
Modbus address	Address within the Modbus Data model or memory map (Range 0 ... 65535).
Data count	Data counter
Function code	The Modbus standard function codes for send and receive operations: FC1: Read coils FC2: Read discrete inputs (input bits) FC3: Read multiple registers FC4: Read input registers FC5: Write coil FC6: Write single register FC15 : Force multiple coils FC16: Write multiple registers FC23: Read/Write multiple registers
Data	The first data of the last sent function code (telegram) are displayed.
Error Count	Counter for errors
Last Error	Last occurred error (Description see appropriate manual)

Table 50: Extended Diagnosis > OMB > OMB IO

6.6 PROFINET IO Device

6.6.1 Overview PROFINET IO Device

Here you find an overview of the **Extended Diagnosis** for *PROFINET IO-Device*.

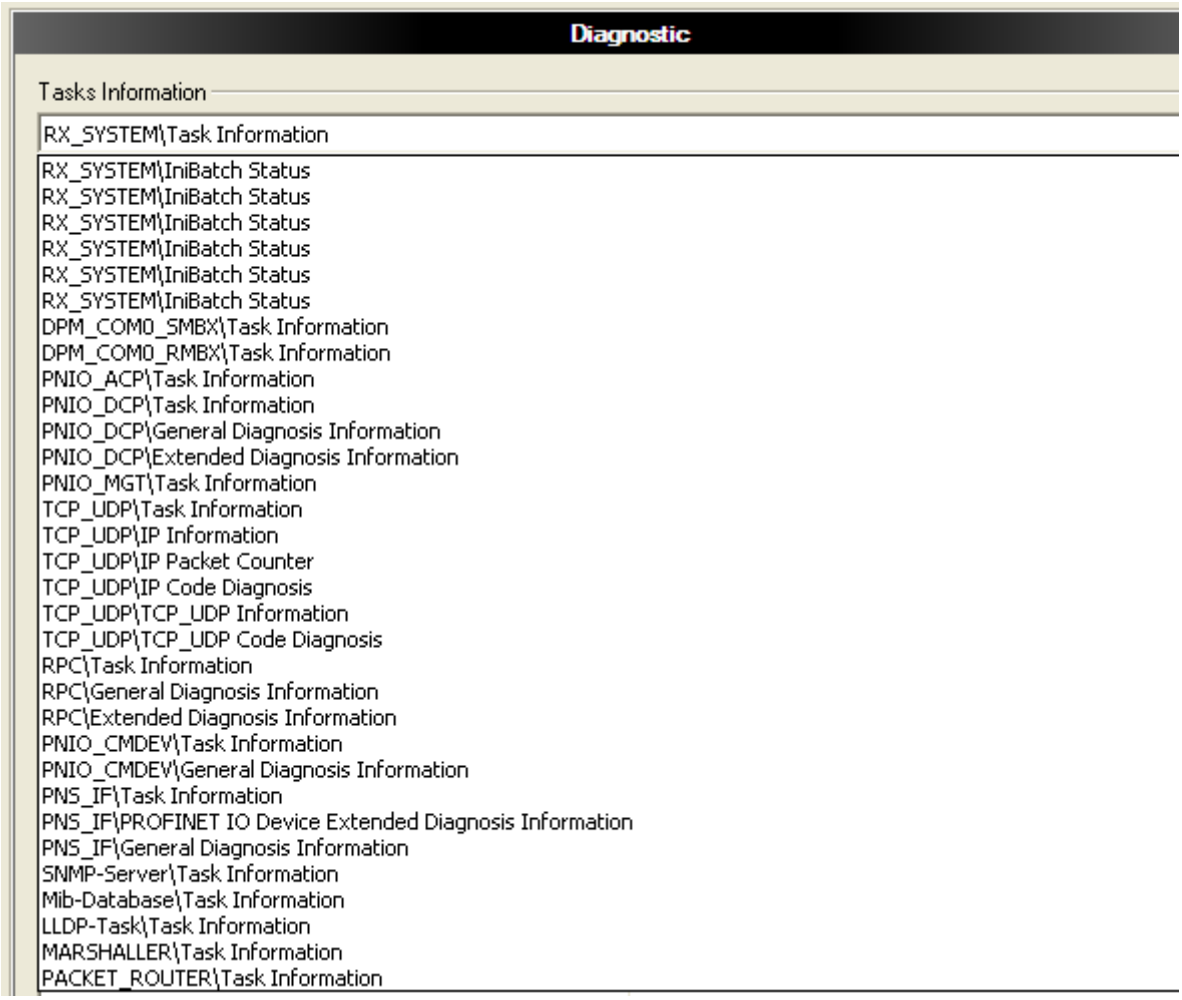


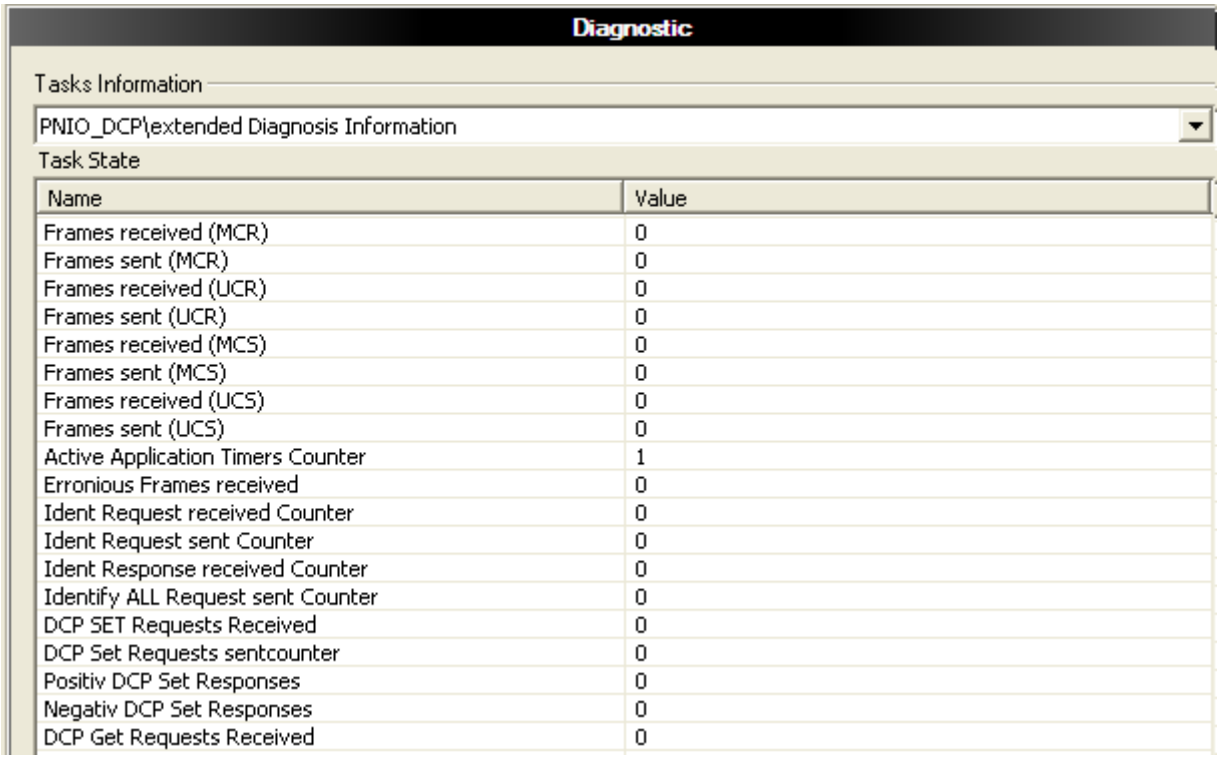
Figure 48: Extended Diagnosis PROFINET IO-Device (cifX, comX and netJACK, Example cifX)

Task Information Group	Taks Information	Manual Page
PNIO_DCP	*PNIO_DCP/Extended Diagnosis Information*	97
RPC	*RPC/Extended Diagnosis Information*	99
PNS_IF	*PNS_IF/PROFINET IO Device Extended Diagnosis Info*	100

Table 51: Descriptions of the Dialog Panes Extended Diagnosis PROFINET IO Device

For the description of further tasks refer to section *Descriptions for Tasks with similar Functions* on page 166.

6.6.2 PNIO_DCP/Extended Diagnosis Information



Name	Value
Frames received (MCR)	0
Frames sent (MCR)	0
Frames received (UCR)	0
Frames sent (UCR)	0
Frames received (MCS)	0
Frames sent (MCS)	0
Frames received (UCS)	0
Frames sent (UCS)	0
Active Application Timers Counter	1
Erronious Frames received	0
Ident Request received Counter	0
Ident Request sent Counter	0
Ident Response received Counter	0
Identify ALL Request sent Counter	0
DCP SET Requests Received	0
DCP Set Requests sentcounter	0
Positiv DCP Set Responses	0
Negativ DCP Set Responses	0
DCP Get Requests Received	0

Figure 49: Extended Diagnosis > PNIO_DCP > Extended Diagnosis Information

The extended Diagnosis Information PNIO_DCP displays the counter reading of the four state machines from the PROFINET IO DCP protocol.

MCR: Multi Cast Receiver

UCR: Uni Cast Receiver

MCS: Multi Cast Sender

UCS: Uni Cast Sender

Name	Description
Frames received (MCR)	Multi Cast Receiver Frames received
Frames sent (MCR)	Multi Cast Receiver Frames sent
Frames received (UCR)	Uni Cast Receiver Frames received
Frames sent (UCR)	Uni Cast Receiver Frames sent
Frames received (MCS)	Multi Cast Sender Frames received
Frames sent (MCS)	Multi Cast Sender Frames sent
Frames received (UCS)	Uni Cast Sender Frames received
Frames sent (UCS)	Uni Cast Sender Frames sent
Active Application Timers Counter	Software timer actually running in the task
Erroneous Frames received	Counter for erroneous frames received
Ident Request received Counter	Counter for Ident Requests received
Ident Request sent Counter	Counter for Ident Requests sent
Ident Response received Counter	Counter for Ident Responses received
Identify ALL Request sent Counter	Counter for Identify ALL Requests sent
DCP SET Requests Received	Counter for DCP SET Requests received
DCP Set Requests sentcounter	Counter for DCP Set Requests sent
Positive DCP Set Responses	Counter for Positive DCP Set Responses
Negative DCP Set Responses	Counter for Negative DCP Set Responses
DCP Get Request Received	Counter for DCP Get Requests received

Table 52: Extended Diagnosis > PNIO_DCP > Extended Diagnosis Information

6.6.3 RPC/Extended Diagnosis Information

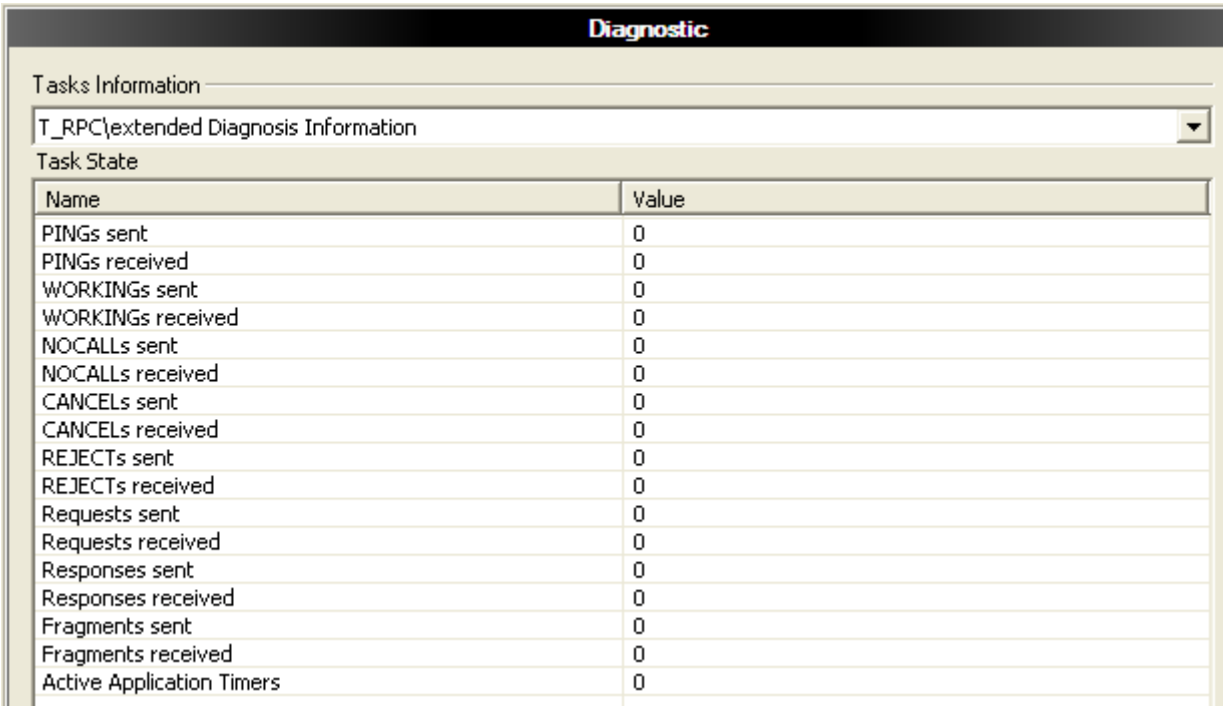


Figure 50: Extended Diagnosis > RPC > Extended Diagnosis Information

The parameters under *RPC > Extended Diagnosis Information* represent PROFINET specific counters.

For further information refer to the PROFINET IO specification. [2], [3]

6.6.4 PNS_IF/PROFINET IO Device Extended Diagnosis Info

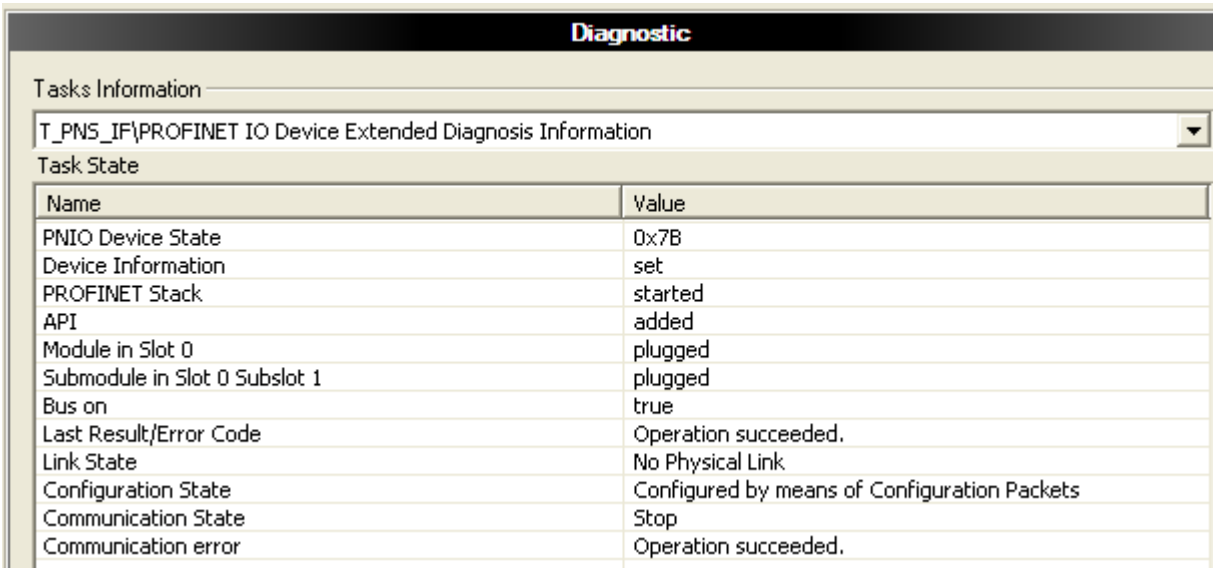


Figure 51: Extended Diagnosis > PNS_IF/PROFINET IO Device Extended Diagnosis Information

ID	Description
PNIO Device Status	Summary of the PROFINET IO stack status
Device information	Manufacturer information about the device, which is defined in the GSDML file.
PROFINET stack	Status of the PROFINET IO stack: started
API	API of the PROFINET IO stack: opened
Module in Slot 0	Module in Slot 0 of the PROFINET IO stack: plugged
Submodule in Slot 0 Subslot 1	Submodule in Slot 0 Subslot 1 of the PROFINET IO stack: plugged
Bus on	Indicates whether the bus is on (active) or off (offline)
Last Result/Error Code	Last occurred error of the PROFINET IO stack
Link State	State of the physical network connection of the PROFINET IO stack
Configuration State	Configuration of the PROFINET IO stack
Communication State	Communication State of the PROFINET IO stack
Communication error	Indicates communication errors of the PROFINET IO stack

Table 53: Extended Diagnosis > PNS_IF > PROFINET Stack State

6.7 POWERLINK Controlled Node

6.7.1 Overview POWERLINK Controlled Node

Here you find an overview of the **Extended Diagnosis** for *POWERLINK Controlled Node*.

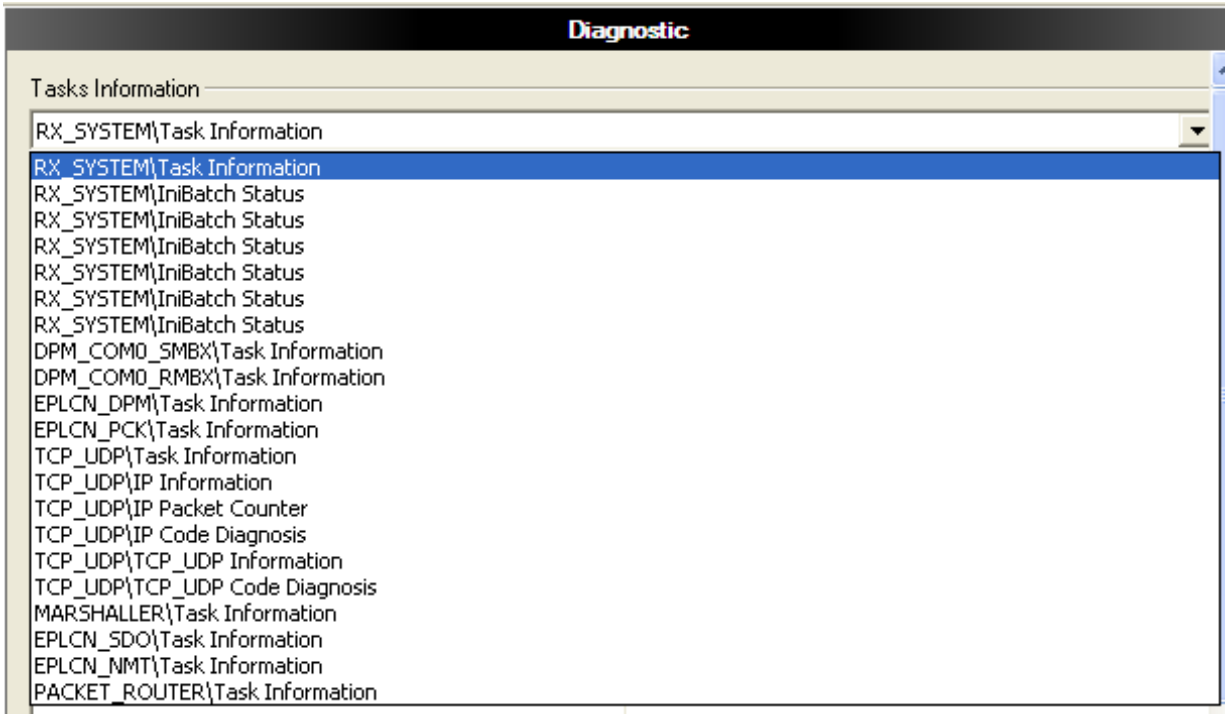


Figure 52: Extended Diagnosis POWERLINK Controlled Node (cifX, comX and netJACK, Example cifX)

For the description of further tasks refer to section *Descriptions for Tasks with similar Functions* on page 166.

6.8 sercos

6.8.1 Overview sercos

Here you find an overview of the **Extended Diagnosis** for *sercos*.

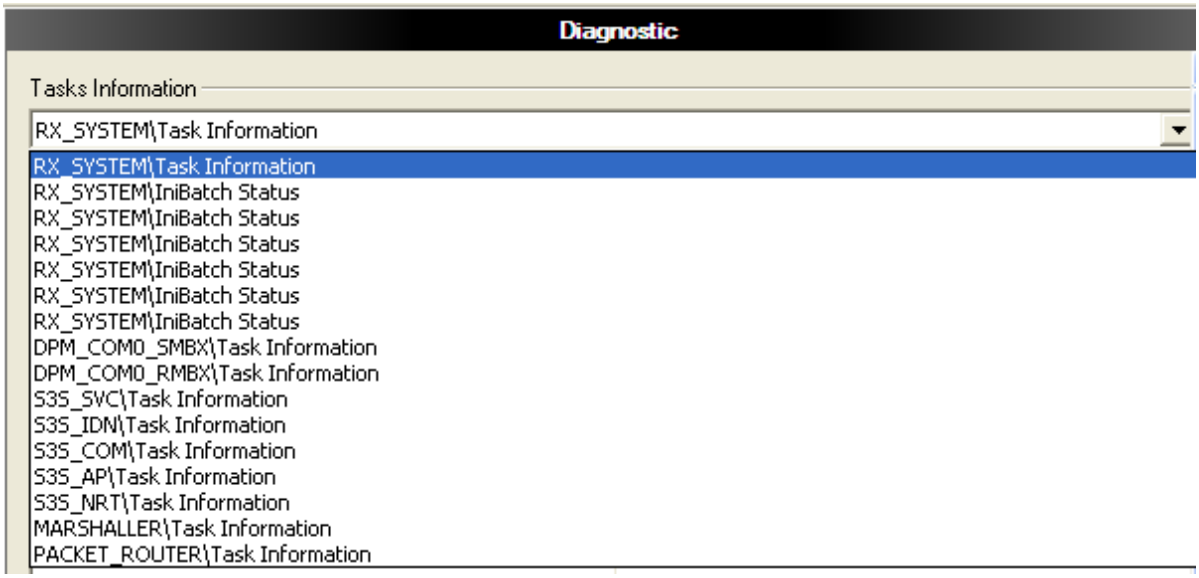


Figure 53: Extended Diagnosis sercos (cifX, comX and netJACK, Example cifX)

For the descriptions of tasks of the extended diagnosis sercos refer to section *Descriptions for Tasks with similar Functions* on page 166.

6.9 VARAN Client (Slave)

6.9.1 Overview VARAN Client (Slave)

Here you find an overview of the **Extended Diagnosis** for *VARAN Client (Slave)*.

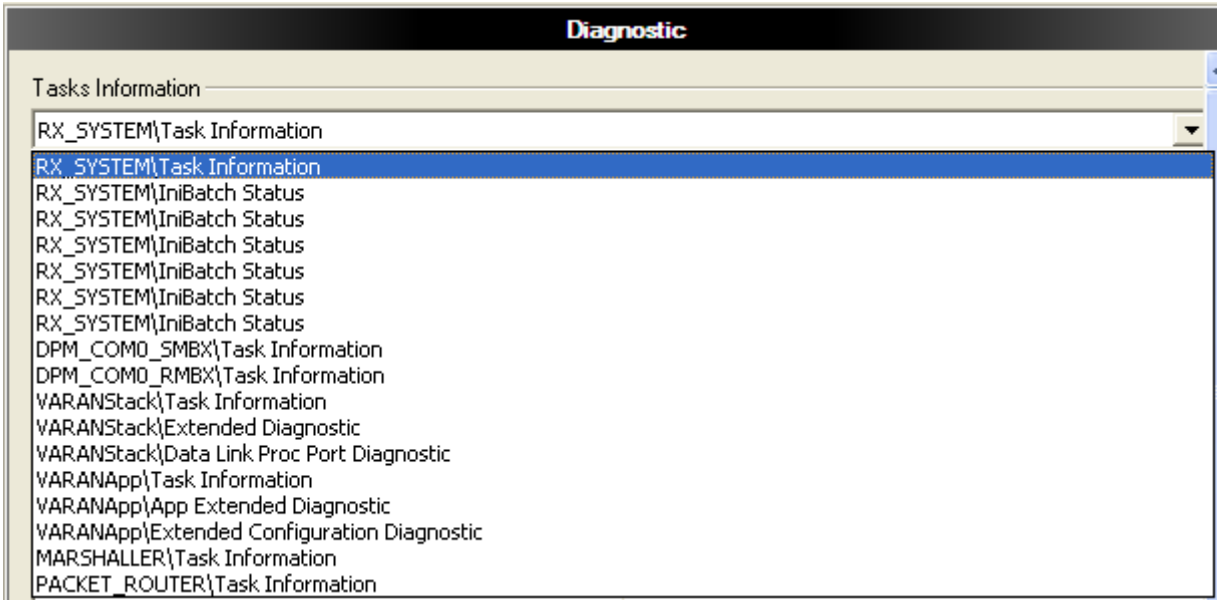


Figure 54: Extended Diagnosis VARAN Client (Slave) (cifX, comX and netJACK, Example cifX)

Task Information Group	Taks Information	Manual Page
VARANStack	*Extended Diagnostic*	104
	Data Link Proc Port Diagnostic	105
VARANApp	*App Configuration Diagnostic*	106
	Extended Configuration Diagnostic	107

Table 54: Descriptions of the Dialog Panes Extended Diagnosis VARAN Client (Slave)

For the description of further tasks refer to section *Descriptions for Tasks with similar Functions* on page 166.

6.9.2 Extended Diagnostic

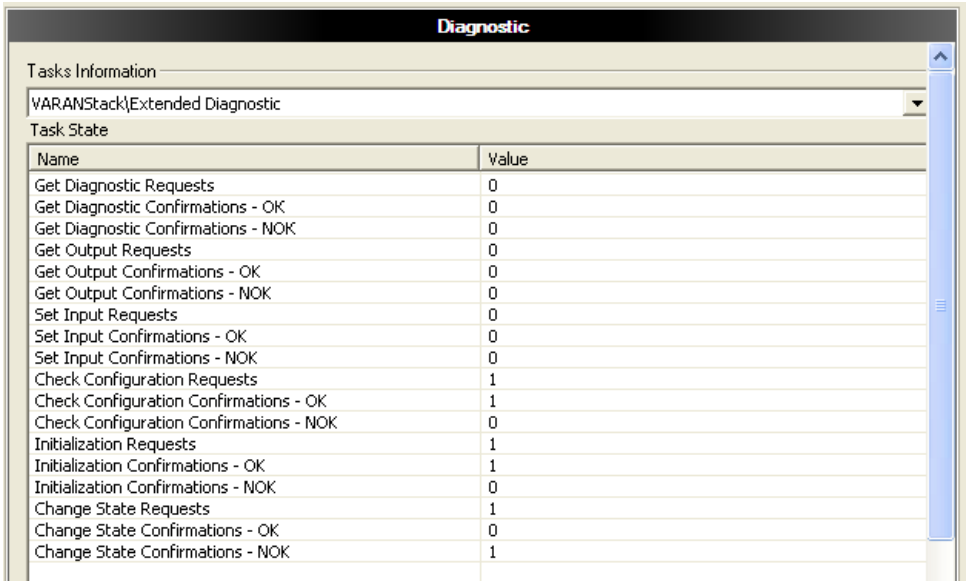


Figure 55: Extended Diagnosis > Extended Diagnostic

Name	Description
[*Service*]	Diagnosis counter: Indicates the services processed. (The services of the single packets are described in the VARAN Client Protocol API manual [15], in section VARAN_CLIENT_CMD_GET_DIAG_REQ/CNF_T.)

Table 55: Extended Diagnosis > Extended Diagnostic

6.9.3 Data Link Proc Port Diagnostic

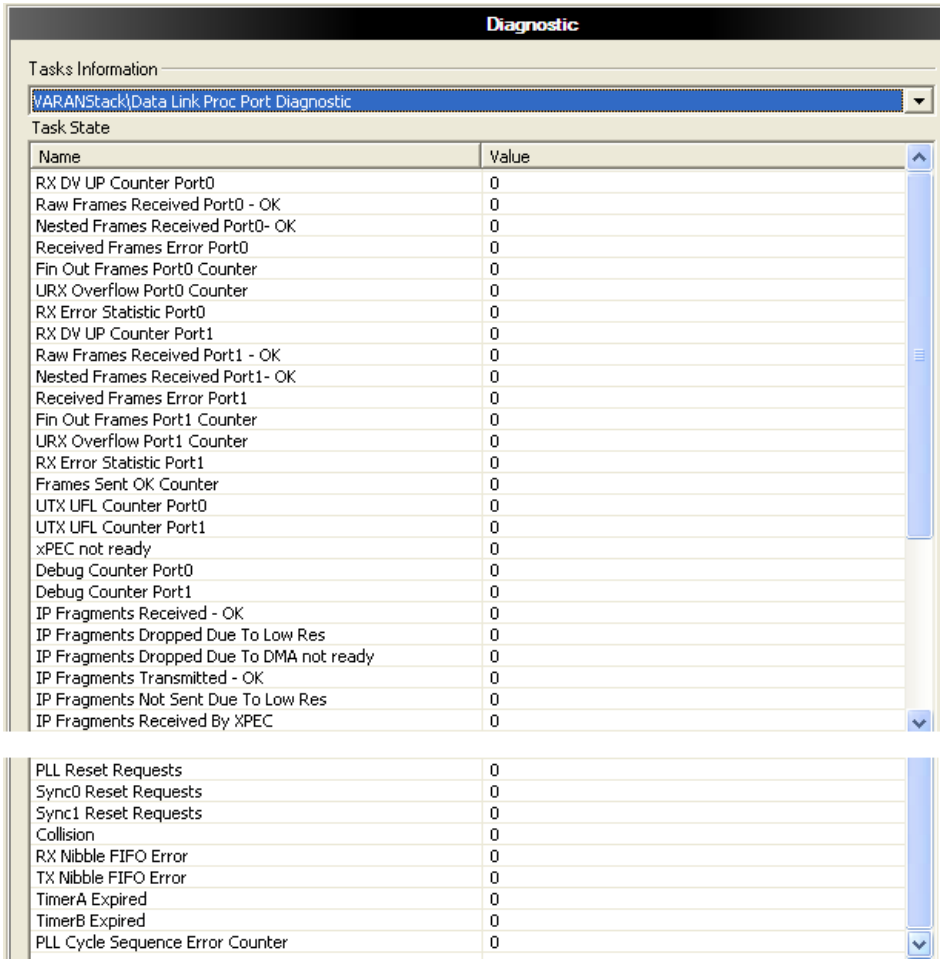


Figure 56: Extended Diagnosis > Data Link Proc Port Diagnostic

Name	Description
[*Service*]	Diagnosis counter: Indicates the services processed. (The services of the single packets are described in the VARAN Client Protocol API manual [15], in section VARAN_CLIENT_CMD_GET_DIAG_REQ/CNF_T.)

Table 56: Extended Diagnosis > Data Link Proc Port Diagnostic

6.9.4 App Configuration Diagnostic

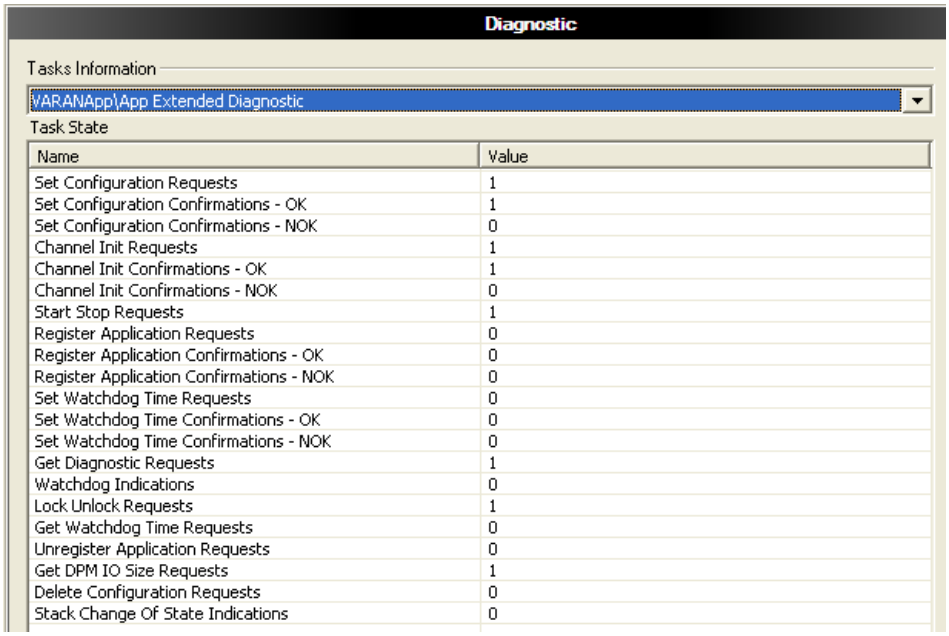


Figure 57: Extended Diagnosis > App Configuration Diagnostic

Name	Description
[*Service*]	Diagnosis counter: Indicates the services processed (refer also to the Dual-Port Memory Interface Manual [17]).

Table 57: Extended Diagnosis > App Configuration Diagnostic

6.9.5 Extended Configuration Diagnostic

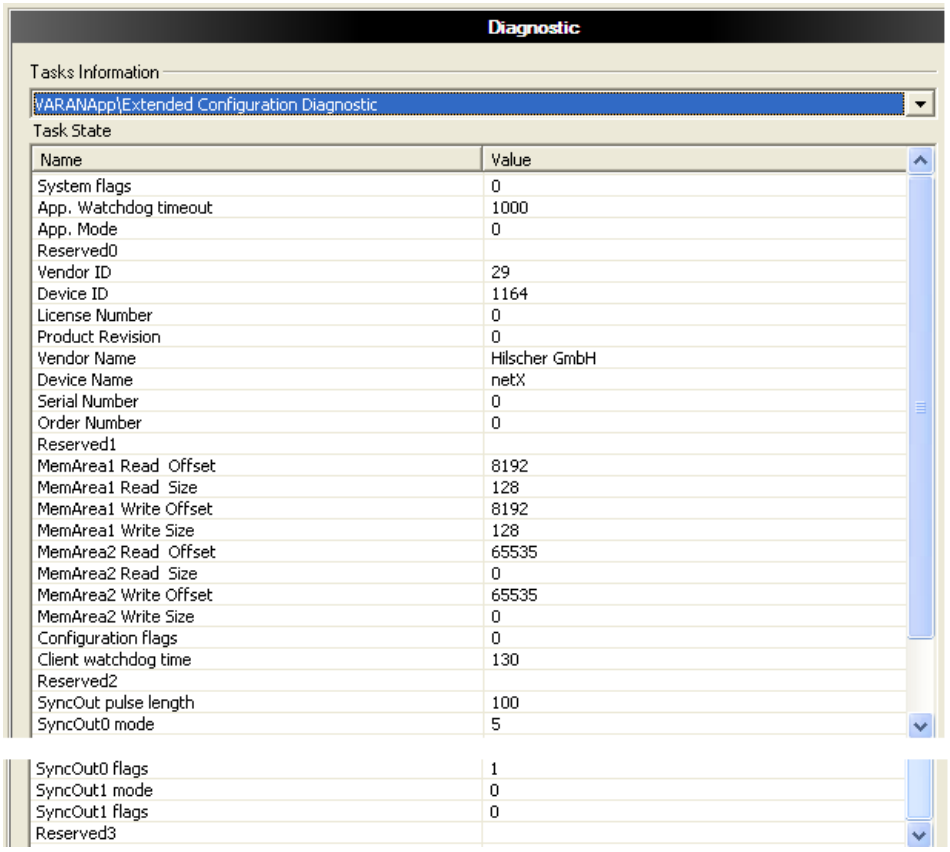


Figure 58: Extended Diagnosis > Extended Configuration Diagnostic

Parameter	Meaning	Range of Value/Value
System flags [Bus Startup]	Communication start application controlled or automatic	1 (=Application controlled), 0 (=Automatic), Default: 0
App. Watchdog timeout [Client Watchdog Time]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Off
App.Mode	Application mode 0 = VARAN_CLIENT_APP_MODE_IO	Default = 0
Vendor ID	Vendor Identifier: Identification number of the manufacturer. This value is specific for every single vendor.	0 … 2^{32} – 1, Hilscher: 29
Device ID	Device Identifier: Identification number of the device as indicated by the VARAN-BUS-NUTZERORGANISATION. This value is specific for every single device type and is fixed for every device. In case of an unknown device ID the manager rejects the client.	00 … 2^{32} – 1, CIFX RE/VRS: 1064; COMX100xx-RE/VRS: 1062; NIC50-RE/VRS: 1063; NJ50x-RE/VRS: 1060; NJ100xx-RE/VRS: 1061
License Number	License number of the device as specified by the VARAN-BUS-NUTZERORGANISATION.	0 … 2^{32} – 1, Default: 0
Product Revision [Product Version]	Version of the device as specified by the manufacturer.	0 … 2^{32} – 1, Default: 0
Vendor Name	Name of the manufacturer, specific for every single vendor.	Zeichenkette, 0 ... 64 Zeichen, Default: Hilscher GmbH

Parameter	Meaning	Range of Value/Value
Device Name	Name of the device as indicated by the VARAN-BUS-NUTZERORGANISATION.	Zeichenkette, 0 ... 64 Zeichen, Default: CIFX RE/VRS, COMX100xx-RE/VRS, NIC50-RE/VRS, NJ50x-RE/VRS, NJ100xx-RE/VRS
Serial Number	Serial number of the device as specified by the manufacturer.	0 … 2^{32} – 1, Default: 0
Order Number	Order number of the device as specified by the manufacturer.	0 … 2^{32} – 1, Default: 0
MemArea 1 Read Offset	Reading offset of the memory area 1in Byte.	0 … 65535 Byte, Default: 8192 Byte
Read Size / Memory Area 1	Reading size of the memory area 1in Byte.	0 … 128 Byte, Default: 128 Byte
Write Offset / Memory Area 1	Writing offset of the memory area 1in Byte.	0 … 65535 Byte, Default: 8192 Byte
Write Size / Memory Area 1	Writing size of the memory area 1 in Byte.	0 … 128 Byte, Default: 128 Byte
Read Offset / Memory Area 2*	Reading offset of the memory area 2 in Byte. (*currently not supported, set to default value)	0 … 65535 Byte, Default: 65535 Byte
Read Size / Memory Area 2*	Reading size of the memory area 2 in Byte. (*currently not supported, set to default value)	0 … 128 Byte, Default: 0 Byte
Write Offset / Memory Area 2*	Writing offset of the memory area 2 in Byte. (*currently not supported, set to default value)	0 … 65535 Byte, Default: 65535 Byte
Write Size / Memory Area 2*	Writing size of the memory area 2 in Byte. (*currently not supported, set to default value)	0 … 128 Byte, Default: 0 Byte
Confignuration Flags	Bit 0: Enable/Disable EMAC If set, the parameters for Memory Area 2 are enabled. Bit 1: Enable/Disable MemArea 2 If set, an integrated Extended MAC-Address (EMAC) for IP data exchange with client application is used. Bit 0 and 1 are currently not supported and they are set to default value.	Bit 0: 0 = EMAC disable, Bit 0: 1 = EMAC enable, Default value: 0 Bit 1: 0 = MemArea 2 disable, Bit 1: 1 = MemArea 2 enable, Default value: 0
Client Watchdog Time	Client watchdog time in ms. The Client watchdog time is the maximum cyclic communication timeout allowed, before Client signals a bus error.	[0 … 130] ms, Default = 130 ms, 0 = Off
SyncOut pulse length [Pulse Length (x10ns)]	Pulse length of the SYNC Out signal in 10ns steps. (e.g. the value 100 results as 10ns*100=1000ns=1µs pulse)	0 … 2.147.483.647, Default: 100

Sync OUT 0 / Sync OUT 1

Note!
Only the following 3 combinations between SyncOut0Mode + SyncOut1Mode are reasonable:
"Time for IN/OUT valid (TIO) – (**5**)" + "Disable – (**0**)",
"Time for IN valid (TI) – (**3**)" + "Time for OUT valid (T0) – (**4**)",
"Time for OUT valid (TO) – (**4**)" + "Time for IN valid (TI) – (**3**)"

Parameter	Meaning	Range of Value/Value
SyncOut0 mode	"SyncOut 0 Mode“ for the SYNC Out 0 signal, using the settings: "Disable – (**0**)": The output of the SYNC Out 0 signal is not available, "Time for IN valid (TI) – (**3**)": For the output of the SYNC Out 0 signal the time for the data IN is valid, "Time for OUT valid (TO) – (**4**)": For the output of the SYNC Out 0 signal the time for the data OUT is valid, "Time for IN/OUT valid (TIO) – (**5**)": For the output of the SYNC Out 0 signal the time for the data IN/OUT is valid.	0,3,4,5 Default: 5
SyncOut0 Flags	Bit 0: Enable/Disable Output The “Sync out 0 flag/Output” enables / disables the output of the Memory Area 1 SYNC Out 0 signal. Bit 1: Polarity active high/ low The “Sync out 0 flag/Polarity” defines the active high/low behavior of the SYNC Out 0 signal.	Bit 0: 0 = Output disable, Bit 0: 1 = Output enable, Default value: 1 Bit 1: 0 = active low, Bit 1: 1 = active high, Default value: 1
SyncOut1 mode	"SyncOut 1 Mode“ for the SYNC Out 1 signal, using the settings: "Disable – (**0**)": The output of the SYNC Out 1 signal is not available, "Time for IN valid (TI) – (**3**)": For the output of the SYNC Out 1 signal the time for the data IN is valid, "Time for OUT valid (TO) – (**4**)": For the output of the SYNC Out 1 signal the time for the data OUT is valid, "Time for IN/OUT valid (TIO) – (**5**)": For the output of the SYNC Out 1 signal the time for the data IN/OUT is valid.	0,3,4,5 Default: 0
SyncOut1 Flags	Bit 0: Enable/Disable Output The “Sync out 1 flag/Output” enables / disables the output of the Memory Area 2 SYNC Out 1 signal. Bit 1: Polarity active high/ low The “Sync out 1 flag/Polarity” defines the active high/low behavior of the SYNC Out 1 signal.	Bit 0: 0 = Output disable, Bit 0: 1 = Output enable, Default value: 0 Bit 1: 0 = active low, Bit 1: 1 = active high, Default value: 0

Table 58: Extended Diagnosis > Extended Configuration Diagnostic

6.10 PROFIBUS DP Slave

6.10.1 Overview PROFIBUS DP Slave

Here you find an overview of the **Extended Diagnosis** for *PROFIBUS Slave*.

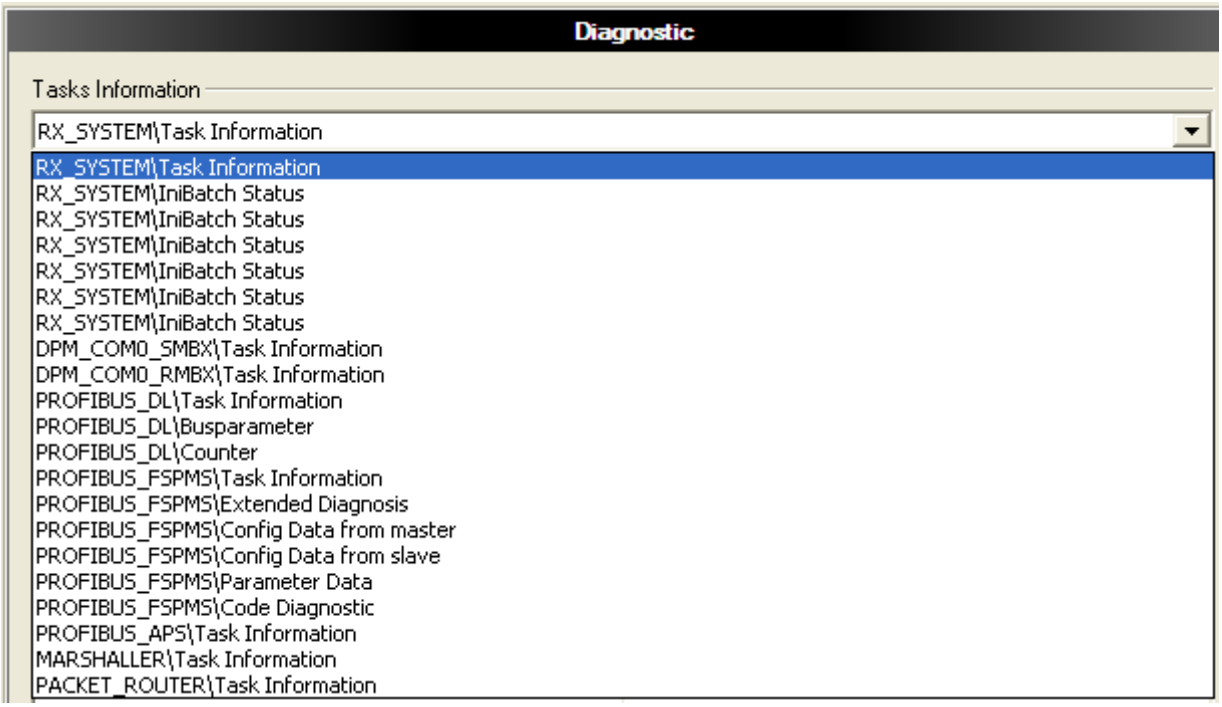


Figure 59: Extended Diagnosis PROFIBUS Slave (cifX, comX and netJACK, Example cifX)

Task Information Group	Taks Information	Manual Page
PROFIBUS_DL	*PROFIBUS_DL/Busparameter*	111
	PROFIBUS_DL/Counter	113
PROFIBUS_FSPMS	*PROFIBUS_FSPMS/Extended Diagnosis*	114
	PROFIBUS_FSPMS/ Config Data from Master	115
	PROFIBUS_FSPMS/ Config Data from Slave	115
	PROFIBUS_FSPMS/Parameter Data	116

Table 59: Descriptions of the Dialog Panes Extended Diagnosis PROFIBUS Slave

For the description of further tasks refer to section *Descriptions for Tasks with similar Functions* on page 166.

6.10.2 PROFIBUS_DL/Busparameter

Under **Extended Diagnosis > PROFIBUS_DL > Busparameter** the values of the configured bus parameters are displayed which are active at the bus.

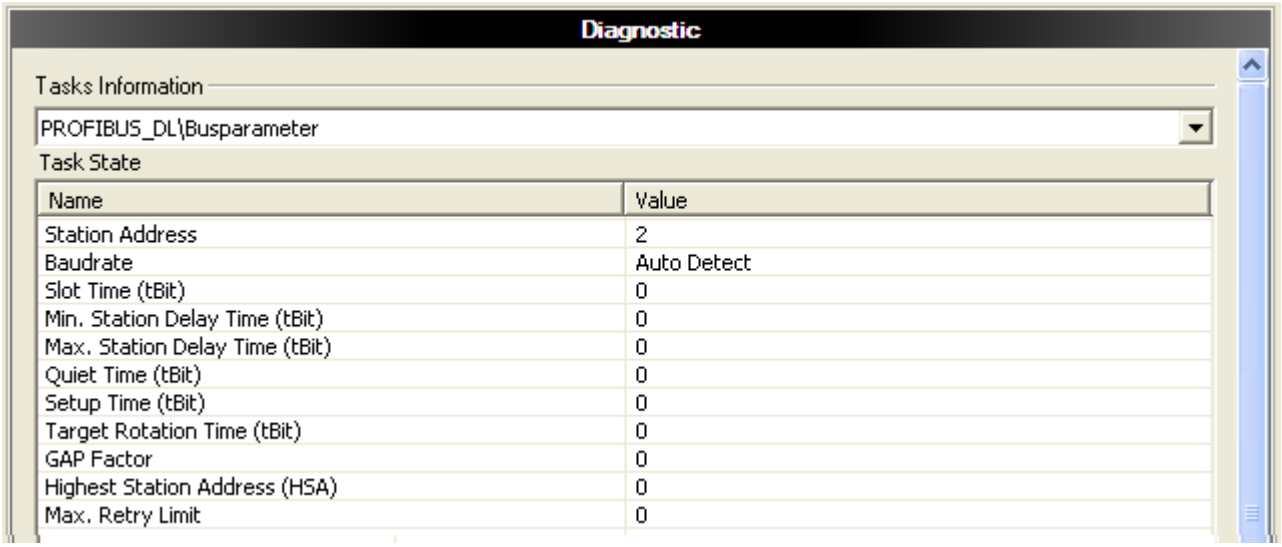


Figure 60: Extended Diagnosis > PROFIBUS_DL> Busparameter

<table>
<tr><th>Bus Parameters</th><th colspan="3">Meaning</th></tr>
<tr><td>Station Address</td><td colspan="3">The Station Address is the individual device address of the Master device on the bus.
Value range: 0 .. 125</td></tr>
<tr><td rowspan="3">Baud Rate</td><td colspan="3">The Baud Rate is the data transfer speed: number of Bits per second.
The Baud Rate must be set to be the same for all devices on the bus. The result of changing the Baud rate is that all other parameters must be re-calculated.</td></tr>
<tr><td>Baud Rate</td><td>Bit time (t_{Bit})</td><td>Max cable length (type A)</td></tr>
<tr><td>9,6 kBit/s
19,2 kBit/s
31,25 kBit/s
45,45 kBit/s
93,75 kBit/s
187,5 kBit/s
500 kBit/s
<u>1500 kBit/s</u>
3000 kBit/s
6000 kBit/s
12000 kBit/s</td><td>104,2 us
52,1 us
32 us
22 us
10,7 us
5,3 us
2 us
666,7 ns
333,3 ns
166,7 ns
83,3 ns</td><td>1200 m
1200 m
1200 m
1200 m
1200 m
1000 m
400 m
200 m
100 m
100 m
100 m</td></tr>
<tr><td>Slot time (tBit)</td><td colspan="3">'Wait for receipt' – Monitoring time of the sender (Requestor) of telegram for the acknowledgement of the recipient (Responder). After expiration, a retry occurs in accordance with the value of 'Max. telegram retries'.
Value range: 37 .. 16383 (The default value depends from the baud rate.)</td></tr>
<tr><td>Min. Station Delay Time (tBit)</td><td colspan="3">This is the shortest time period that must elapse before a remote recipient (Responder) may send an acknowledgement of a received query telegram. The shortest time period between the reception of the last Bit of a telegram to the sending of the first Bit of a following telegram.
Value range: 1 .. <u>11</u> . 65535</td></tr>
<tr><td>Max. Station Delay Time (tBit)</td><td colspan="3">This is the longest time period that must elapse before a Sender (Requestor) may send a further query telegram. Greatest time period between the reception of the last Bit of a telegram to the sending of the first Bit of a following telegram.
The Sender (Requestor, Master) must wait at least for this time period after the sending of an unacknowledged telegram (e.g. Broadcast only) before a new telegram is sent.
Value range: 1 .. 65535 (The default value depends from the baud rate.)</td></tr>
<tr><td>Quiet Time (tBit)</td><td colspan="3">This is the time delay that occurs for modulators (Modulator-trip time) and Repeaters (Repeater-switch time) for the change over from sending to receiving.
Value range: 0 .. 127 (The default value depends from the baud rate.)</td></tr>
</table>

Bus Parameters	Meaning
Setup Time (tBit)	Minimum period “reaction time” between the receipt of an acknowledgement to the sending of a new query telegram (Reaction) by the Sender (Requestor). Value range: 0 .. 255 (The default value depends from the baud rate.)
Target Rotation Time (tBit)	Pre-set nominal Token cycling time within the Sender authorization (Token) will cycle around the ring. How much time the Master still has available for sending data telegrams to the Slaves is dependent on the difference between the nominal and the actual token cycling time. The **Target rotation time (T_{TR})** is shown in Bit times (tBit) like the other Bus Parameters. Below the displayed Bit time, the **Target rotation time** is also displayed in milliseconds (ms). Value range: 1 .. 2^{24}-1 (=16.777.215) (The default value depends on the number of Slaves attached to the Master and their module configuration)
GAP Factor	Factor for determining after how many Token cycles an added participant is accepted into the Token ring. After expiry of the time period $G*T_{TR}$, the Station searches to see whether a further participant wishes to be accepted into the logical ring. Value range: 0 .. **10** .. 255
Highest Station Address (HSA)	The **Highest Station Address** is the highest bus address up to which a Master searches for another Master at the bus in order to pass on the Token. This station address must on no account be smaller than the Master station address. Value range: 1 .. **126**
Max. Retry Limit	Maximum number of repeats in order to reach a Station. Value range: 1 .. 15 (The default value depends from the baud rate.)

Table 60: Extended Diagnosis > PROFIBUS_DL> Busparameter

6.10.3 PROFIBUS_DL/Counter

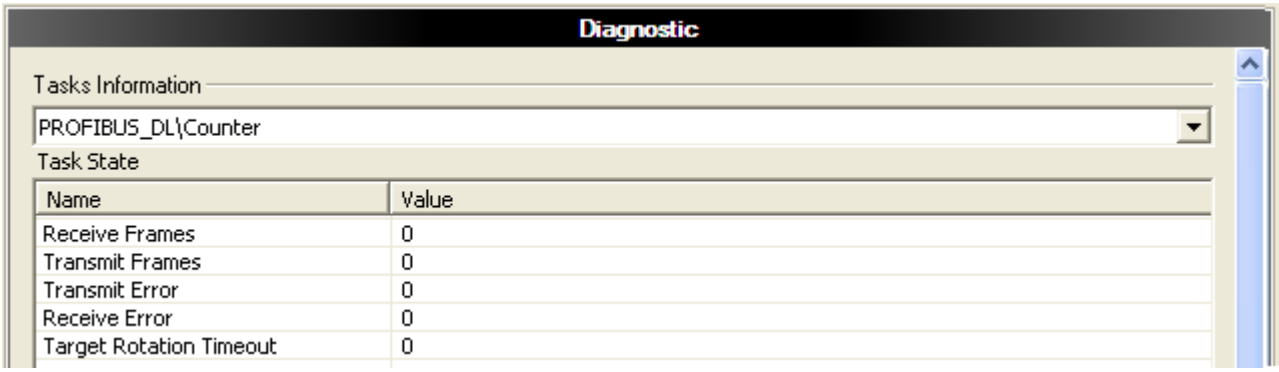


Figure 61: Extended Diagnosis > PROFIBUS_DL> Counter

The values of the counter **Receive Frames** and **Transmit Frames** show generally whether there is bus activity or not.

Name	Description
Receive Frames	Counter for number received frames
Transmit Frames	Counter for number transmitted frames
Transmit Error	Counter for number transmitted errors
Receive Error	Counter for number received errors
Target Rotation Timeout	Counter for number target rotation timeout

Table 61: Extended Diagnosis > PROFIBUS_DL> Counter

6.10.4 PROFIBUS_FSPMS/Extended Diagnosis

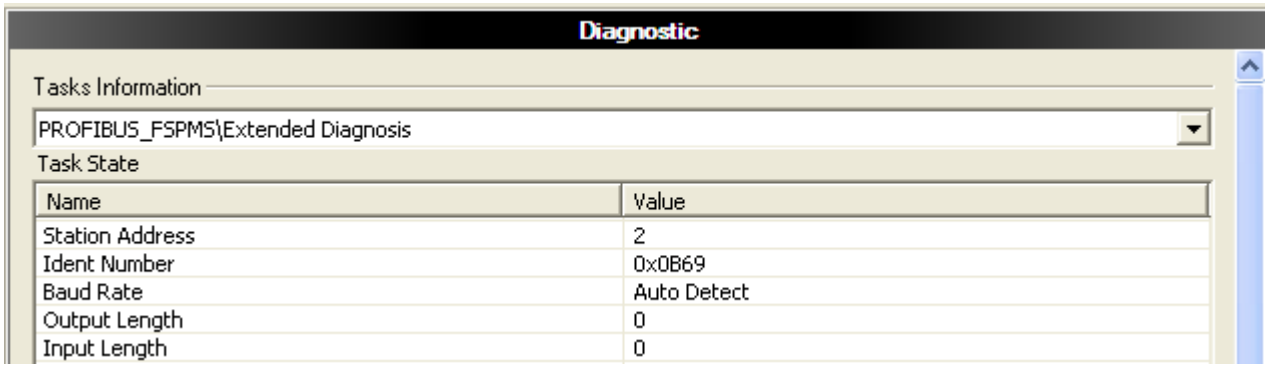


Figure 62: Extended Diagnosis > PROFIBUS_FSPMS > Extended Diagnosis

Name	Description	
Station Address	PROFIBUS address of the device Range of value: 0 … 125	
Ident Number	PROFIBUS Identification Number Range of value: 0 … 65535, Default: 0x0A12	
Baud Rate	9,6 kBit/s 19,2 kBit/s 93,75 kBit/s 187,5 kBit/s 500 kBit/s 1,5 MBit/s	3 MBit/s 6 MBit/s 12 MBit/s 31,25 kBit/s 45,45 kBit/s Auto detect
Output Length	Number of the output bytes Range of value: 0 … 244	
Input Length	Number of the input bytes Range of value: 0 … 244	

Table 62: Extended Diagnosis > PROFIBUS_FSPMS > Extended Diagnosis

6.10.5 PROFIBUS_FSPMS/ Config Data from Master

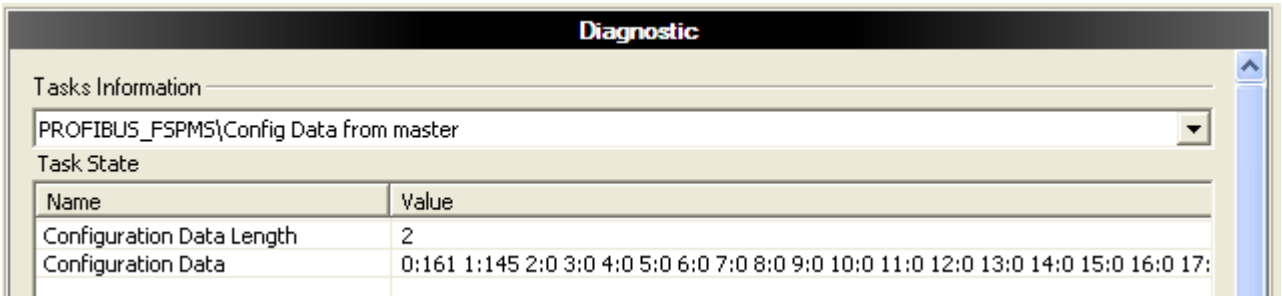


Figure 63: Extended Diagnosis > PROFIBUS_FSPMS > Config Data from Master

Name	Description
Configuration Data Length	Number of the configuration data from Master in byte Describes how many bytes of the configuration data are valid.
Configuration Data	Left value: numbering of every configuration data byte Right value: value of the respective configuration data byte

Table 63: Extended Diagnosis > PROFIBUS_FSPMS > Config Data from Master

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.

6.10.6 PROFIBUS_FSPMS/ Config Data from Slave

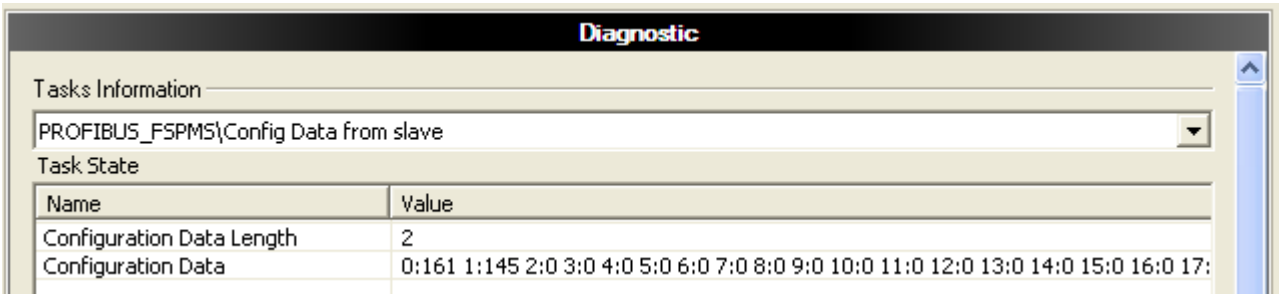


Figure 64: Extended Diagnosis > PROFIBUS_FSPMS > Parameter Data

Name	Description
Parameter Data Length	Number of the parameter data in byte Describes how many bytes of the parameter data are valid.
Parameter Data	Left value: numbering of every parameter data byte Right value: value of the respective parameter data byte

Table 64: Extended Diagnosis > PROFIBUS_FSPMS > Parameter Data

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.

6.10.7 PROFIBUS_FSPMS/Parameter Data

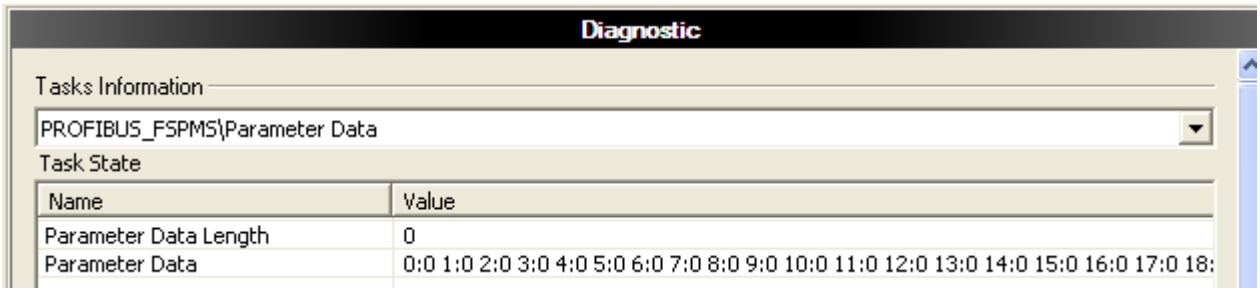


Figure 65: Extended Diagnosis > PROFIBUS_FSPMS > Parameter Data

Name	Description
Parameter Data Length	Number of the parameter data in byte Describes how many bytes of the parameter data are valid.
Parameter Data	Left value: numbering of every parameter data byte Right value: value of the respective parameter data byte

Table 65: Extended Diagnosis > PROFIBUS_FSPMS > Parameter Data

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.

6.11 PROFIBUS MPI

6.11.1 Overview PROFIBUS MPI

Here you find an overview of the **Extended Diagnosis** of PC cards cifX for *PROFIBUS MPI*

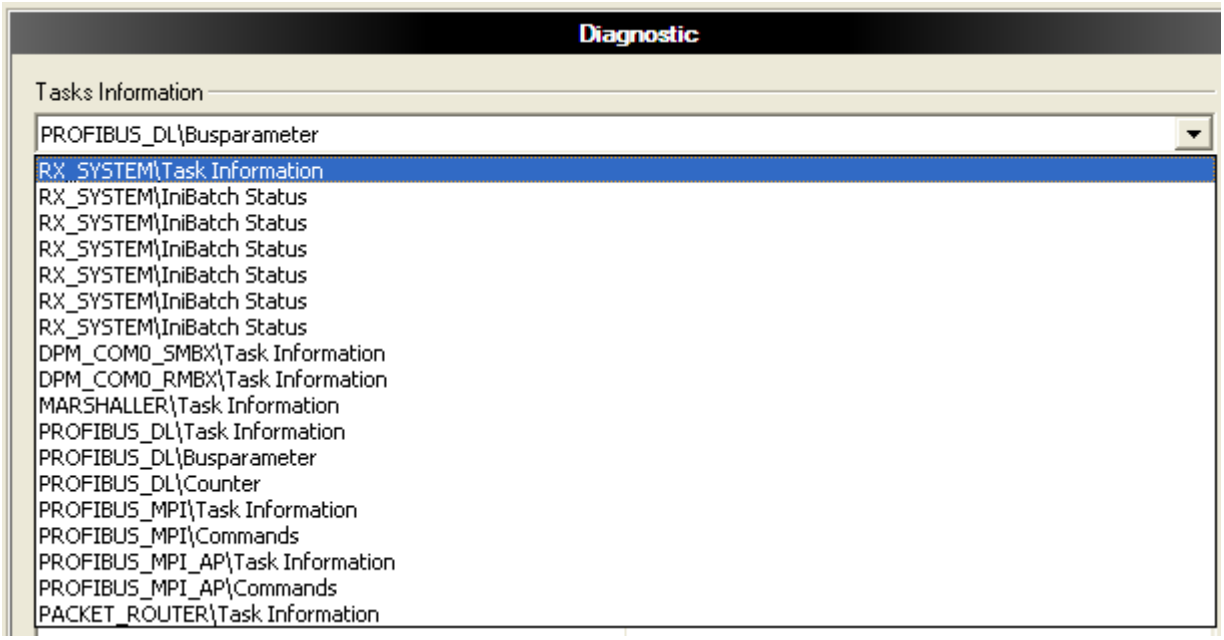


Figure 66: Extended Diagnosis PROFIBUS MPI (cifX)

Task Information Group	Taks Information	Manual Page
PROFIBUS_DL	*PROFIBUS_DL/Busparameter*	111
	PROFIBUS_DL/Counter	119
PROFIBUS_MPI	*PROFIBUS_MPI/Commands*	120
PROFIBUS_MPI_AP	*PROFIBUS_MPI_AP/Commands*	*121*

Table 66: Descriptions of the Dialog Panes Extended Diagnosis PROFIBUS MPI

For the description of further tasks refer to section *Descriptions for Tasks with similar Functions* on page 166.

6.11.2 PROFIBUS_DL/Busparameters

Under **Extended Diagnosis > PROFIBUS_DL> Busparameter** the values of the configured bus parameters active on the bus are displayed.

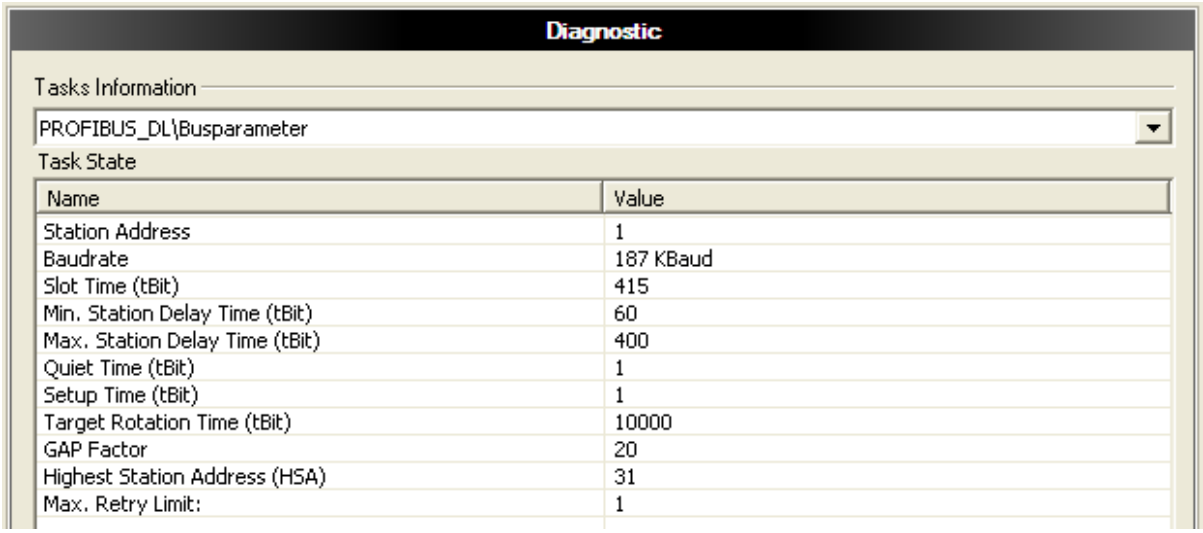


Figure 67: Extended Diagnosis > PROFIBUS_DL> Busparameter

Parameter	Meaning	Range of Value/ Value
Station Address	PROFIBUS MPI Address of the device.	0 … 126, Default: 1
Baudrate	PROFIBUS MPI Baud Rate Indicates the velocity of transmission: Number of bits per second.	9,6 kBit/s, 19,2 kBit/s, 31,25 kBit/s, 45,45 kBit/s, 93,75 kBit/s, 187,5 kBit/s, 500 kBit/s, 1,5 MBit/s, 3 MBit/s, 6 MBit/s, 12 MBit/s, Auto-Detect, Default: 187,5 kBit/s
Slot Time (tBit)	Slot Time (T_{SL})	37 ... 16383 tBit, Default: 415 tBit
Min. Station Delay Time (tBit)	Minimum Station Delay of Responders (min T_{SDR})	1 ... 1023 tBit, Default: 60 tBit
Max. Station Delay Time (tBit)	Maximum Station Delay of Responders (max T_{SDR})	1 ... 1023 tBit, Default: 400 tBit
Quiet Time (tBit)	Quiet Time (T_{QUI})	0 ... 127 tBit, Default: 1 tBit
Setup Time (tBit)	Setup Time (T_{SET})	1 … 255 tBit, Default: 1 tBit
Target Rotation Time (tBit)	Target Rotation Time (T_{TR})	>= 255 tBit, Default: 10000 tBit
GAP Factor	GAP Actualization Factor (G)	1 … 255, Default: 20
Highest Station Address (HAS)	Highest Station Address (H_{SA}), Station address of the highest active (Master) station.	1 … 126, Default: 31
Max. Retry Limit	Max number of telegram retries (Max_Retry_Limit)	1 ... 8, Default: 1

Table 67: Extended Diagnosis > PROFIBUS_DL> Busparameters

6.11.3 PROFIBUS_DL/Counter

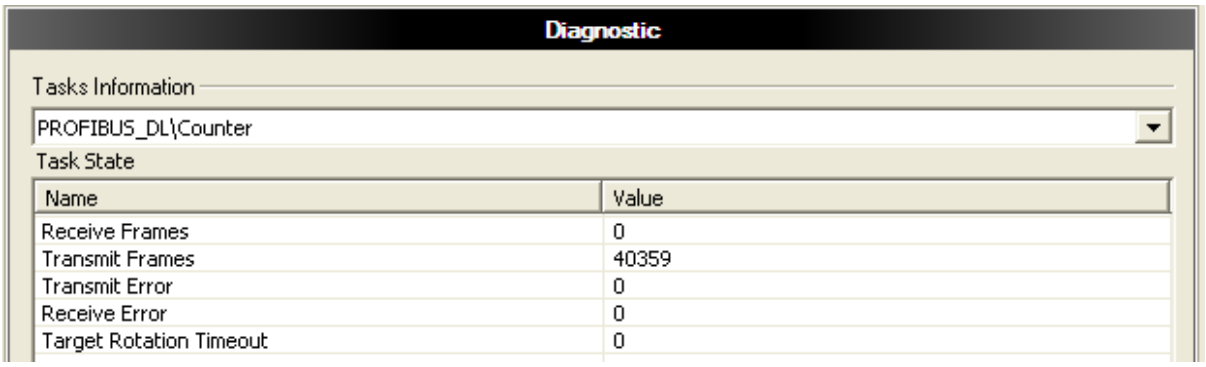


Figure 68: Extended Diagnosis > PROFIBUS_DL> Counter

The values of the counter **Receive Frames** and **Transmit Frames** show generally whether there is bus activity or not.

Name	Description
Receive Frames	Counter for number received frames
Transmit Frames	Counter for number transmitted frames
Transmit Error	Counter for number transmitted errors
Receive Error	Counter for number received errors
Target Rotation Timeout	Counter for number target rotation timeout

Table 68: Extended Diagnosis > PROFIBUS_DL> Counter

6.11.4 PROFIBUS_MPI/Commands

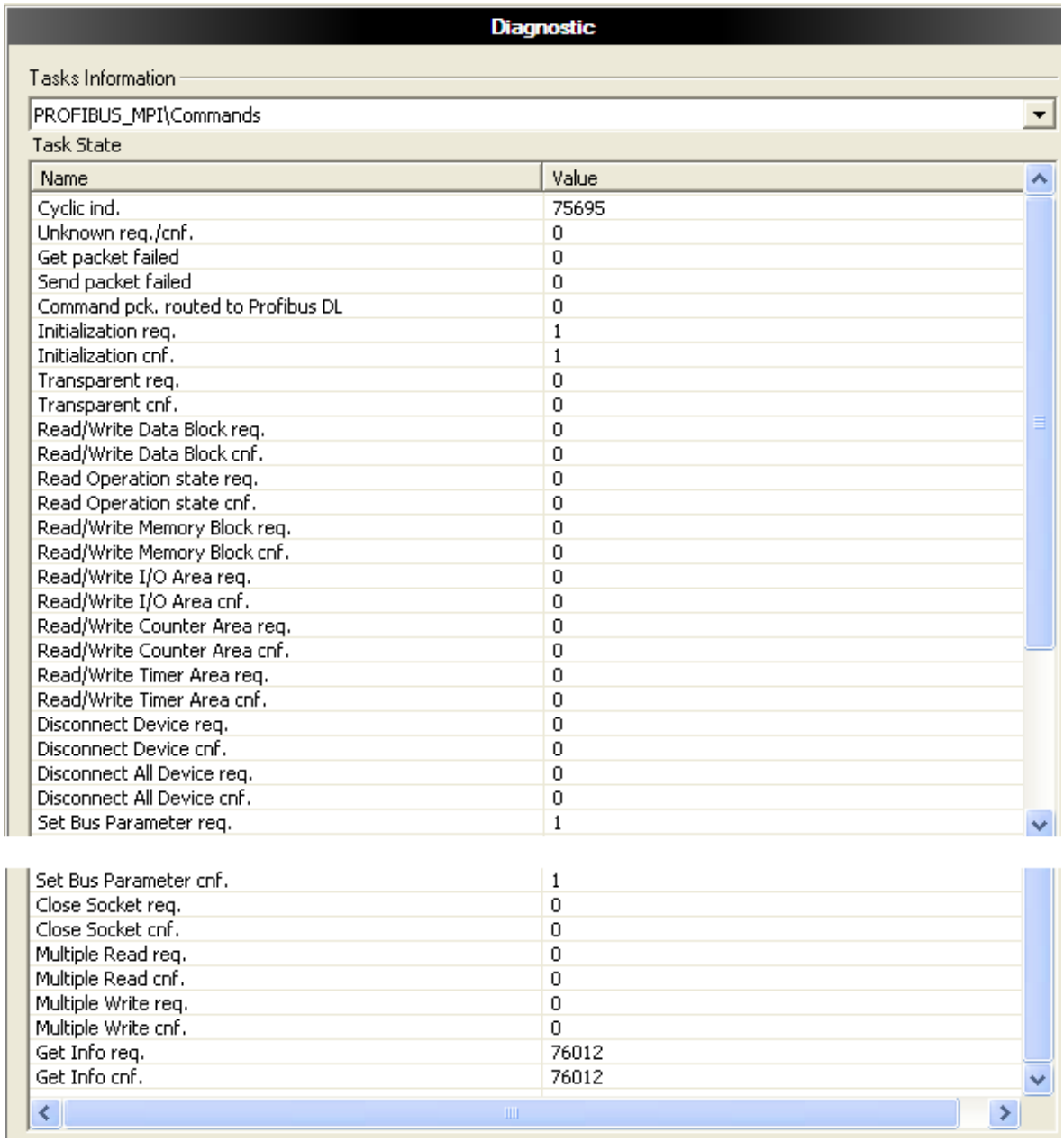


Figure 69: Extended Diagnosis > PROFIBUS_MPI > Commands (Example)

Name	Description
[*Service*]	Diagnosis counter: Indicates the services processed. (The services of the single packets are described in the PROFIBUS MPI Protocol API manual [4].)

Table 69: Extended Diagnosis > PROFIBUS_MPI > Commands

6.11.5 PROFIBUS_MPI_AP/Commands

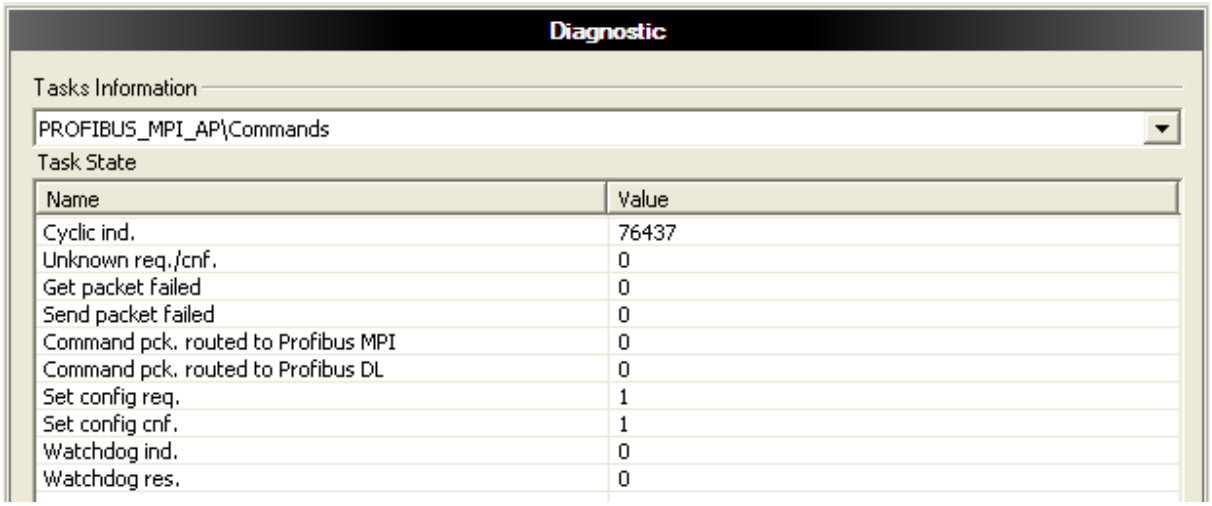


Figure 70: Extended Diagnosis > PROFIBUS_MPI_AP > Commands (Example)

Name	Description
[*Service*]	Diagnosis counter: Indicates the services processed. (The services of the single packets are described in the PROFIBUS MPI Protocol API manual [4].)

Table 70: Extended Diagnosis > PROFIBUS_MPI_AP > Commands

6.12 CC-Link Slave

6.12.1 Overview CC-Link Slave

Here you find an overview of the **Extended Diagnosis** for *CC-Link Slave*.

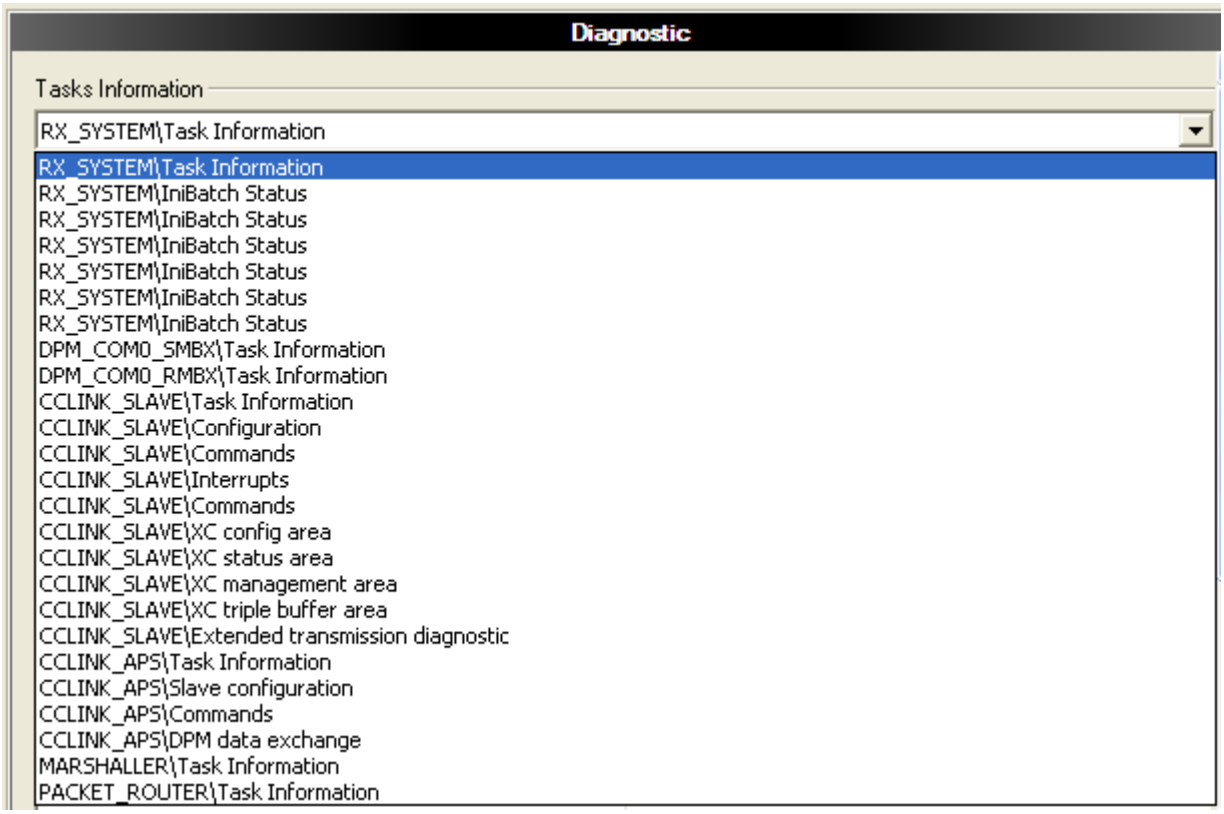


Figure 71: Extended Diagnosis CC-Link Slave (cifX, comX and netJACK, Example cifX)

Task Information Group	Taks Information	Manual Page
CCLINK_SLAVE	*CCLINK_SLAVE/Configuration*	123
	CCLINK_SLAVE/Commands	125
	CCLINK_SLAVE/Interrupts	126
	CCLINK_SLAVE/XC config area	127
	CCLINK_SLAVE/XC status area	130
	CCLINK_SLAVE/XC management area	133
	CCLINK_SLAVE/XC triple buffer area	135
	CCLINK_SLAVE/Extended transmission diagnostic	136
CCLINK_APS	*CCLINK_APS/Slave configuration*	137
	CCLINK_APS/Commands	138
	CCLINK_APS/DPM data exchange	139

Table 71: Descriptions of the Dialog Panes Extended Diagnosis CC-Link Slave

For the description of further tasks refer to section *Descriptions for Tasks with similar Functions* on page 166.

6.12.2 CCLINK_SLAVE/Configuration

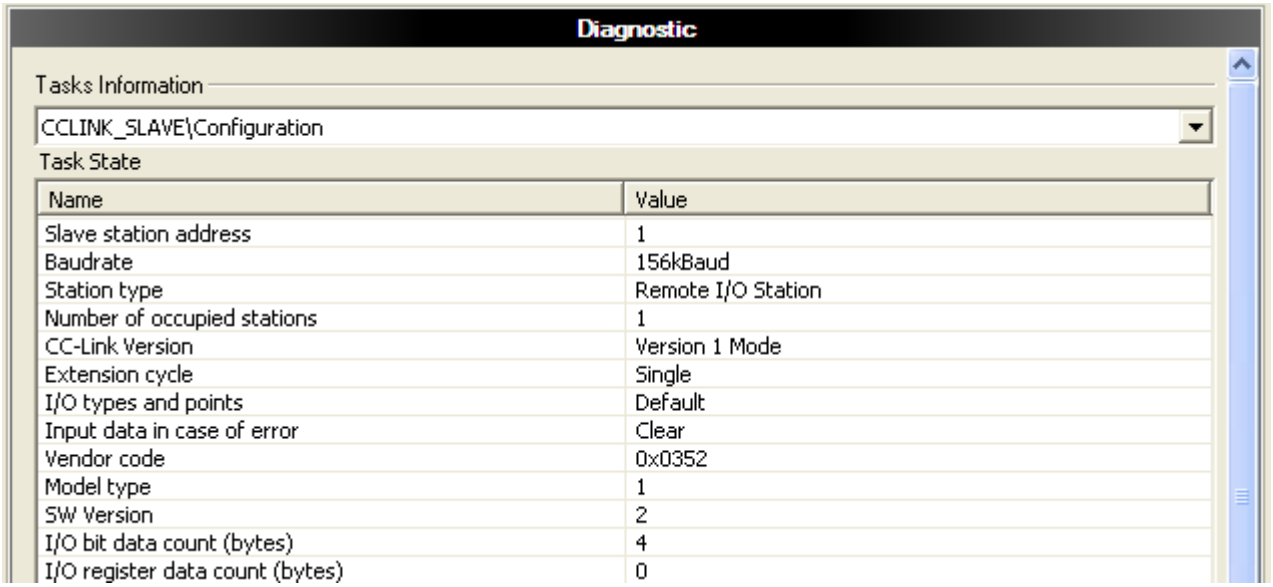


Figure 72: Extended Diagnosis > CCLINK_SLAVE > Configuration

Name	Description
Slave station address	Station address of CC-Link Slave Note: The number of occupied stations plus station address must not exceed the parameter range Range of value: 1 ... 64
Baudrate	0 = Not configured 1 = 156kBaud 2 = 625kBaud 3 = 2500kBaud 4 = 5MBaud 5 = 10MBaud
Station type	0 = Not configured 1 = Remote I/O Station 2 = Remote Device Station 3 = Intelligent Device Station
Number of occupied stations	Number of occupied stations Remote I/O Station: 1 (Default) Remote Device Station: 1 ... 4
CC Link Version	0 = Not configured 1 = Version 1 Mode 2 = Version 2 Mode
Extension cycle	Number of extension cycles 0 = Not configured 1 = Single/One cycle 2 = Double/Two cycles 3 = Quadruple/Four cycles 4 = Octuple/Eight cycles Allowed numbers for CC-Link version 1: Single Allowed numbers for CC-Link version 2: Single, Double, Quadruple, Octuple

Name	Description
I/O types and points	Total number of points; I/O type available: Mixed, Input, Output, Composite 0 = Not configured 1 = Default 2 = Mixed, depending on number of occupied station 3 = Mixed, 8 points (Bits) 4 = Mixed, 16 points (Bits) 5 = Mixed, 32 points (Bits) 6 = Input, depending on number of occupied station 7 = Input, 8 points (Bits) 8 = Input, 16 points (Bits) 9 = Input, 32 points (Bits) 10 = Output, depending on number of occupied station 11 = Output, 8 points (Bits) 12 = Output, 16 points (Bits) 13 = Output, 32 points (Bits) 14 = Composite, depending on number of occupied station 15 = Composite, 8 points (Bits) 16 = Composite, 16 points (Bits) 17 = Composite, 32 points (Bits) “Mixed” means that both input and output exist on the same module. “Composite” means a device that doesn’t use the same numbers for input and output.
Input data in case of error	0 = Not configured 1 = Clear 2 = Hold
Vendor code	Vendor code (If corresponding bit in the vendor code flags parameter is set) Range of value: 0 ... 65535, Default: 0x0352
Model type	Model type (If corresponding bit in the model type flags parameter is set) Range of value: 0 ... 255
SW Version	Software version (If corresponding bit in the software version flags parameter is set) Range of value: 0 ... 63, Default: 1
I/O bit data count (bytes)	Counter for I/O bits in bytes
I/O register data count (bytes)	Counter for I/O register in bytes

Table 72: Extended Diagnosis > CCLINK_SLAVE > Configuration

6.12.3 CCLINK_SLAVE/Commands

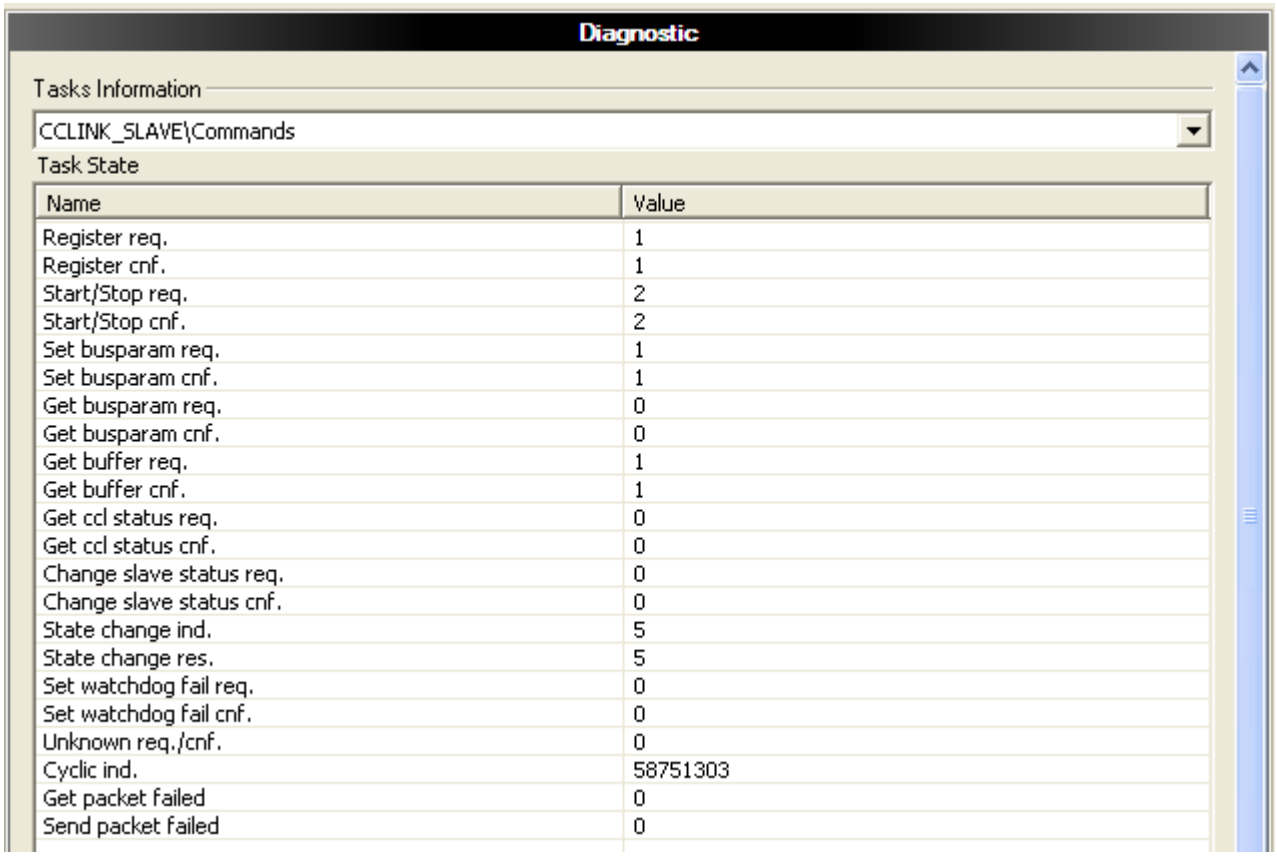


Figure 73: Extended Diagnosis > CCLINK_SLAVE > Commands

Name	Description
[*Service*]	Diagnosis counter: Indicates the services processed. (The services of the single packets are described in the CC-Link Slave Protocol API manual [7].)
Get packet failed	Number of errors during requesting a packet
Send packet failed	Number of errors during sending a packet

Table 73: Extended Diagnosis > CCLINK_SLAVE > Commands

6.12.4 CCLINK_SLAVE/Interrupts

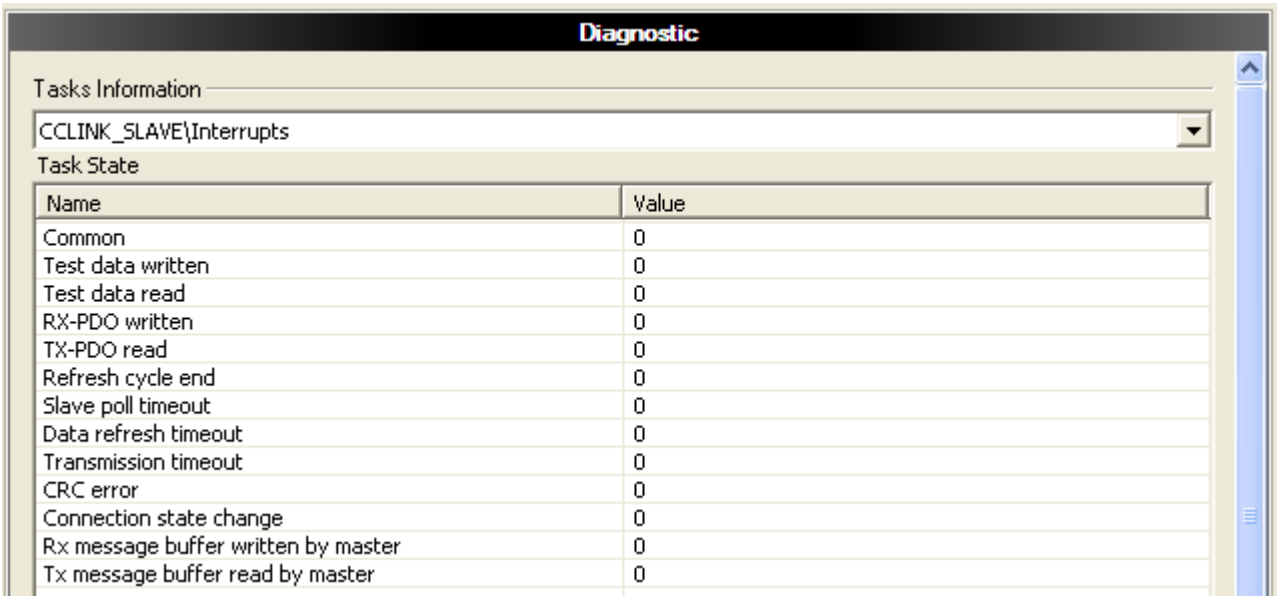


Figure 74: Extended Diagnosis > CCLINK_SLAVE > Interrupts

Name	Description
[*Service*]	Diagnosis counter: Indicates the services processed. (The services of the single packets are described in the CC-Link Slave Protocol API manual [7].)

Table 74: Extended Diagnosis > CCLINK_SLAVE > Interrupts

6.12.5 CCLINK_SLAVE/XC config area

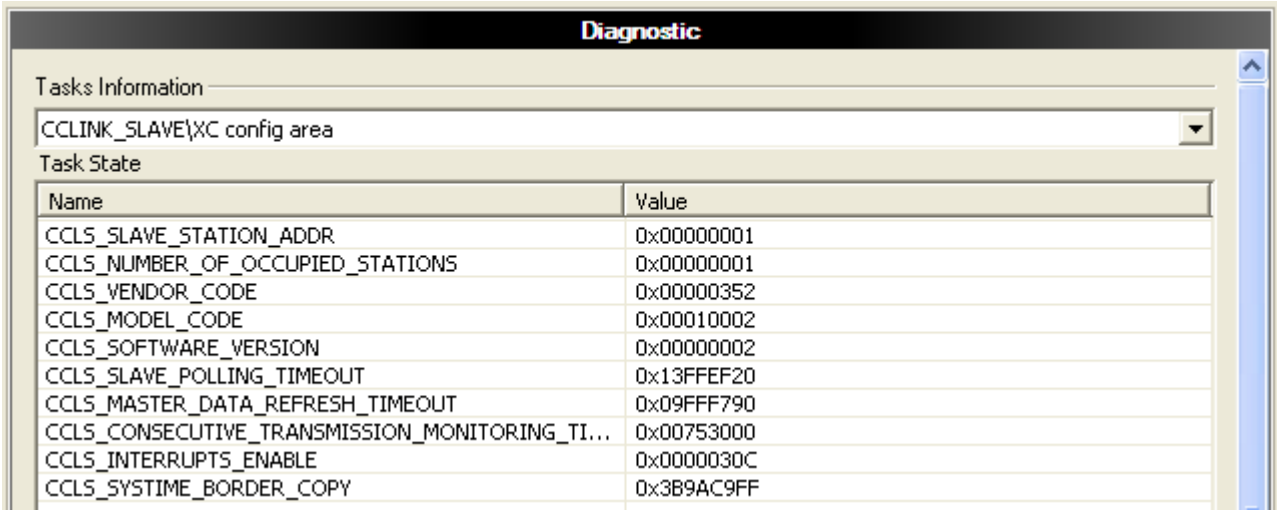


Figure 75: Extended Diagnosis > CCLINK_SLAVE > XC config area

ID	Value	Description
CCLS_SLAVE_STATION_ADDR	1 … 64	**Slave station address**
CCLS_NUMBER_OF_OCCUPIED_STATIONS	1 … 4	**Number of occupied stations**
CCLS_VENDOR_CODE		**Vendor code**
CCLS_MODEL_CODE	0 … 2^{32-1}	**Model code** The model codes are allocated in vendor modules. Bits 1 and 0: **Total number of IO-points** 00: dependent of the number of occupied stations, 01: 8 points, 10: 32 points, 11: 16 points Bits 3 and 2: **I/O types** 00: mixed: Where both inputs and outputs exist on the same module. The same I/O numbers (starting from RX0 and RY0, respectively) are used. 01: input, 10: output, 11: composite: A device that does not use the same numbers for input (RX) and output (RY) Bits 5 and 4: **Number of occupied stations** 00: 1 station occupied 01: 2 stations occupied 10: 3 stations occupied 11: 4 stations occupied Bit 7 and 6: *Reserved* Bit 8: **Switch Setting** 0: Normal, 1: Abnormal Bit 9: **Output status when error occurs** 0: Clear, 1: Hold Bits 13 to 10: *reserved*

ID	Value	Description
(continued) CCLS_MODEL_CODE		Bits 15 and 14: **Station Type** (Type of station) 00: remote I/O 01: remote device 10: intelligent device/local 11: Reserved Bits 23 and 16: **Model Type** (This byte defines the model of the slave station. This is allocated seperately for each model.) Bits 31 and 24: *reserved*
CCLS_SOFTWARE_VERSION	0 … 2^{32-1}	**Software version** (Machine information, set by ARM before CC-Link Slave is started) Bits 5 to 0: **Software Version Vendor** (This indicates the software version of each model; dependent on the vendor) Bits 7 and 6: **Protocol Version** (for Version 2) 00: Version 1 01: Version 2 10: Version 3 (Future function) 11: Version 4 (Future function) Bits 31 to 8: *reserved*
CCLS_SLAVE_POLLING_TIMEOUT	0 … 2^{32-1}	Slave polling timeout Bits 31 to 0: **Slave Station Polling**, Timout value in 10ns resolution, 0: timeout monitoring disabled
CCLS_MASTER_DATA_REFRESH_TIMEOUT	0 … 2^{32-1}	Master data refresh timeout Bits 31 to 0: **Master Data Refresh**, Timeout value in 10ns resolution, 0: timeout monitoring disabled
CCLS_CONSECUTIVE_TRANSMISSION_MONITORING_TIMEOUT	0 … 2^{32-1}	Consecutive transmission monitoring timeout Bits 31 to 0: **Monitoring of consecutive transmission time**, Timeout value in 10ns resolution, 0: timeout monitoring disabled
CCLS_INTERRUPTS_ENABLE	0 … 2^{32-1}	Bit 0: **RX Test Data written by Master** (enable event "Master has written new test-loopback data"; to inform ARM that Master Status was updated) Bit 1: **TX Test Data ready by Master** (enable event "Master has polled test-loopback data") Bit 2: **RX PDO written by Master** (enable event "Master has refreshed RXPDO") Bit 3: **TX PDO read by Master** (enable event "Master has polled TXPDO") Bit 4: **Refresh Cycle End** (enable event "Refresh cycle end")

ID	Value	Description
(continued) CCLS_INTERRUPTS_ENABLE		Bit 5: **Slave polling Timeout** (enable event "Slave polling timeout") Bit 6: **Master Data refresh Timeout** (enable event "Master data refresh timeout") Bit 7: **Consecutive Transmission Timeout** (enable event "Consecutive Transmission monitoring timeout") Bit 8: **CRC Error** (enable event "CRC error detected") Bit 9: **Slave connected State changed** (enable event "The connection state of the slave has changed") Bit 10: **RX Message Buffer written by Master** (enable event "Receive message buffer written by master") Bit 11: **TX Message Buffer read by Master** (enable event "Transmit message buffer read by master") Bit 15 to 12: *reserved* Bit 31 to 16: *reserved* (unusable)
CCLS_SYSTIME_BORDER_COPY	0 … 2^{32-1}	Bits 31 to 0: **Systime border copy**

Table 75: Extended Diagnosis > CCLINK_SLAVE > XC config area

6.12.6 CCLINK_SLAVE/XC status area

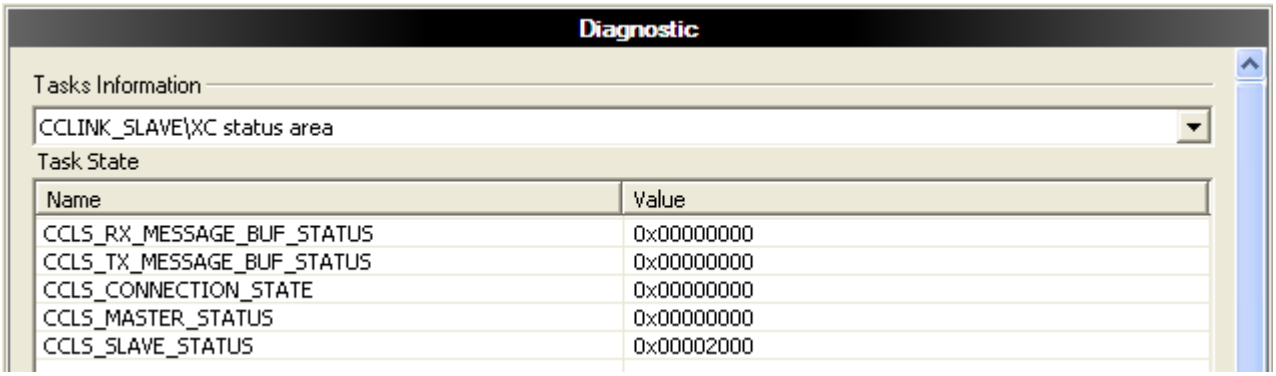


Figure 76: Extended Diagnosis > CCLINK_SLAVE > XC status area

ID	Value	Description
CCLS_RX_MESSAGE_BUF_STATUS	0 … 2^{32-1}	0: Receive Message Buffer Empty, otherwise: Receive Message Buffer Full, set by xPEC, reset by ARM
CCLS_TX_MESSAGE_BUF_STATUS	0 … 2^{32-1}	0: Transmit Message Buffer Empty, otherwise: Transmit Message Buffer Full, set by ARM, reset by xPEC
CCLS_CONNECTION_STATE	0 … 2^{32-1}	Bit 0: **State** (connection state of slave), 0: disconnected, 1: connected, written by xPEC Bit 1: **Network joined** 0: *reserved* 1: local slave was test polled by master Bits 15 and 2: *reserved* Bit 16: **Consecutive Transmission Timeout Status** 0: no error 1: error Bits 31 and 17: *reserved*
CCLS_MASTER_STATUS	0 … 2^{32-1}	Bit 0: **Status1 Master Status User Application Programm** (Operation status Master station user application program) 0: stop 1: run Bit 1: **Status1 Master Status User Application Programm Error Check** (An error occured in the master station user application program) 0: normal 1: error Bit 2: **Status1 refresh Startup** (Link refresh was started) 0: stop 1: start

ID	Value	Description
(continued) CCLS_MASTER_STATUS		Bit 3: **Status1 Transient data status** (Transient data is included) 0: no error 1: error Bit 4: **Status1 Transient data reception enabled** (Ready to receive transient data.) 0: disabled 1: enabled Bits 6 and 5: **Status1 Protocol version** 00: Version 1 01: Version 2 10: Version 3 (Future function) 11: Version 4 (Future function) Bit 7: **Status1 Master Station Type** (Station type of the master station) 0: Master station 1: Standby master station Bits 11 and 8: **Status 2 RY Info Transmission Points** (RY: Remote output length; multiplied with 32 bytes) Bits 15 and 12: **Stauts 2 RWW Info Transmission Points** (RWW: Remote register output length; multiplied with 64 bytes) Bit 31 to 16: *reserved*
CCLS_SLAVE_STATUS	0 … 2^{32-1}	Bit 0: **Station 1 fuse Status** (Salve station fuse has blown) 0: No error 1: Error Bit 1: **Station 1 unit Error invalid Number of Points** (Unit error (remote I/O stations), Invalid number of points flag; slave stations excluding remote I/O stations) 0: No error 1: Error Bit 2: **Station 1 no refresh receive** (No refresh transmission received from the master station; shall be set to zero because bit is inserted by xPEC) 0: Not received 1: Received Bit 3: **Station 1 no parameter receive** (No parameter information received from the master station) 0: Not received 1: Received Bit 4: **Station 1 switch change detection** (Switch setting changed after power is turned on or reset cancelled) 0: Not present 1: Present

ID	Value	Description
(continued) CCLS_SLAVE_STATUS	0 … 2^{32-1}	Bit 5: **Station 1 cyclic Communication** (Cyclic communication is enabled) 0: Enabled 1: Disabled Bit 6: *reserved* Bit 7: **Station 1 Watchdog Error** (Watchdog timer error has occurred) 0: Not present 1: Present Bit 8: **Station 2 Transient Data Status** (Transient data is included; shall be set to zero because bit is inserted by xPEC) 0: No transient data is included 1: Transient data is included Bit 9: **Status2 Transient data reception enabled** (Ready to receive transient data; shall be set to zero because bit is inserted by xPEC) 0: Disabled 1: Enabled Bit 10: **Status2 Transient Type** (Type of transient data) 0: 1:n communication (intelligent device station) 1: n:n communication (local station) Bit 11: *reserved* Bit 12: **Status2 Transmission Route Status** (Transmission route error) 0: No error 1: Error Bit 12: **Station 2 res fixed to one** *reserved* (set to 1) Bit 15 to 14: **State 2 extended Cycle Setting** (Extended cycle setting; Version 2) 00: Single setting (1*) 01: Double setting (2*) 10: Quadruple setting (4*) 11: Octuple setting (8*) Bit 31 to 16: *reserved*

Table 76: Extended Diagnosis > CCLINK_SLAVE > XC status area

6.12.7 CCLINK_SLAVE/XC management area

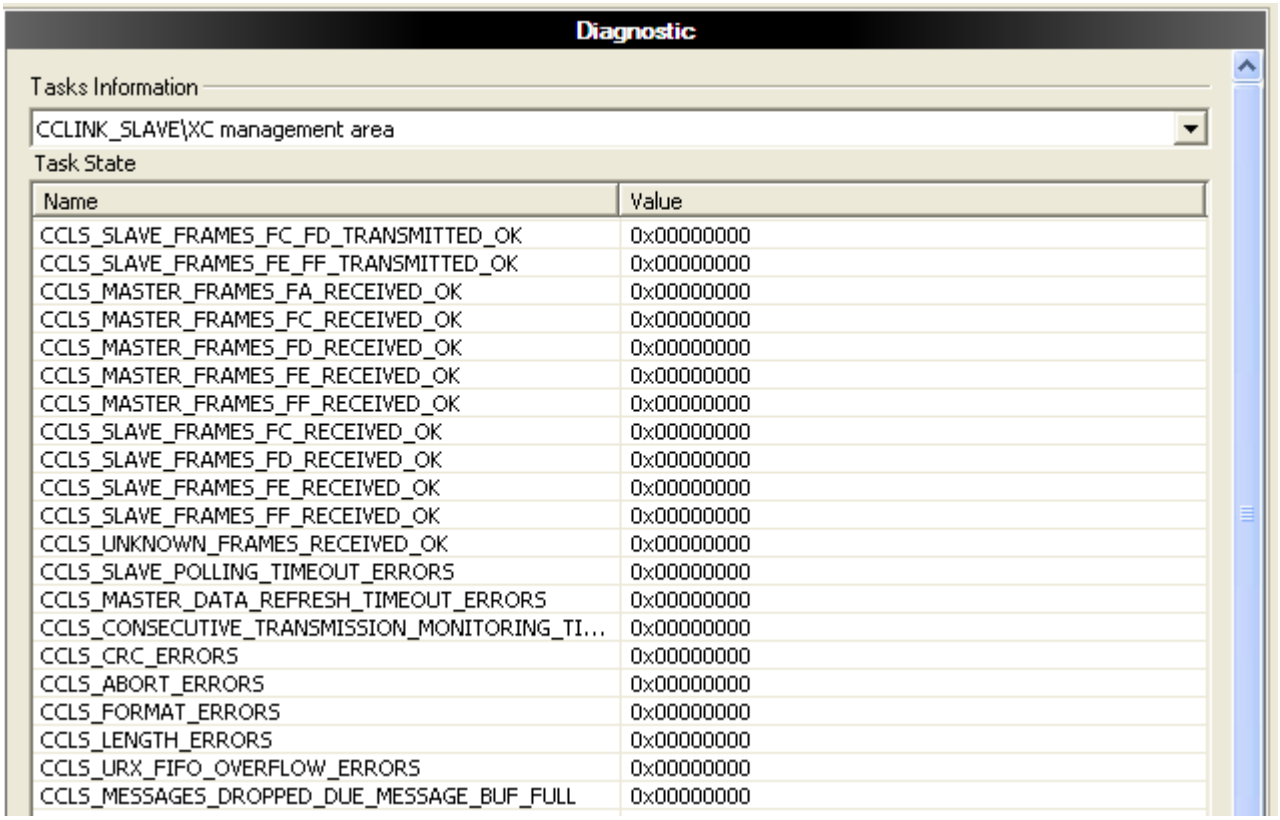


Figure 77: Extended Diagnosis > CCLINK_SLAVE > XC management area

ID	Value	Description
CCLS_SLAVE_FRAMES_FC_FD_TRANSMITTED_OK	0 … 2^{32-1}	Counter for frames of type "Slave station test loop-back data" that are successfully transmitted
CCLS_SLAVE_FRAMES_FE_FF_TRANSMITTED_OK	0 … 2^{32-1}	Counter for frames of type "Slave station response (refresh) data" that are successfully transmitted
CCLS_MASTER_FRAMES_FA_RECEIVED_OK	0 … 2^{32-1}	Counter for frames of type "Refresh cycle end data" that are successfully received
CCLS_MASTER_FRAMES_FC_RECEIVED_OK	0 … 2^{32-1}	Counter for frames of type "Master station test polling data" that are successfully received
CCLS_MASTER_FRAMES_FD_RECEIVED_OK	0 … 2^{32-1}	Counter for frames of type "Master station test polling and test data" that are successfully received
CCLS_MASTER_FRAMES_FE_RECEIVED_OK	0 … 2^{32-1}	Counter for frames of type "Master station polling data" that are successfully received
CCLS_MASTER_FRAMES_FF_RECEIVED_OK	0 … 2^{32-1}	Counter for frames of type "Master station polling and refresh data" that are successfully received
CCLS_SLAVE_FRAMES_FC_RECEIVED_OK	0 … 2^{32-1}	Counter for frames of type "Slave station test loopback data" that are successfully received
CCLS_SLAVE_FRAMES_FD_RECEIVED_OK	0 … 2^{32-1}	Counter for frames of type "Slave station response (refresh) data" that are successfully received
CCLS_SLAVE_FRAMES_FE_RECEIVED_OK	0 … 2^{32-1}	Counter for frames of type "Slave station response (refresh) data" that are successfully received

ID	Value	Description
CCLS_SLAVE_FRAMES_FF_RECEIVED_OK	0 … 2^{32-1}	Counter for frames of type "Slave station test loopback data" that are successfully received
CCLS_UNKNOWN_FRAMES_RECEIVED_OK	0 … 2^{32-1}	Counter for frames of unknown type that are successfully received
CCLS_SLAVE_POLLING_TIMEOUT_ERRORS	0 … 2^{32-1}	Counter for Slave Station Polling Timout errors
CCLS_MASTER_DATA_REFRESH_TIMEOUT_ERRORS	0 … 2^{32-1}	Counter for Master Data Refresh Timeout errors
CCLS_CONSECUTIVE_TRANSMISSION_MONITORING_TIMEOUT_ERRORS	0 … 2^{32-1}	Counter for consecutive transmission timeout errors
CCLS_CRC_ERRORS	0 … 2^{32-1}	Counter for frames that do not pass the CRC check
CCLS_ABORT_ERRORS	0 … 2^{32-1}	Counter for frames that were aborted during reception
CCLS_FORMAT_ERRORS	0 … 2^{32-1}	Counter for frames that have not the expected frame format
CCLS_LENGTH_ERRORS	0 … 2^{32-1}	Counter for frames that have not the expected frame length
CCLS_URX_FIFO_OVERFLOW_ERRORS	0 … 2^{32-1}	Counter for frames that have an urx fifo overflow error
CCLS_MESSAGES_DROPPED_DUE_MESSAGE_BUF_FULL	0 … 2^{32-1}	Counter for times a message was dropped due message buffer was full

Table 77: Extended Diagnosis > CCLINK_SLAVE > XC management area

6.12.8 CCLINK_SLAVE/XC triple buffer area

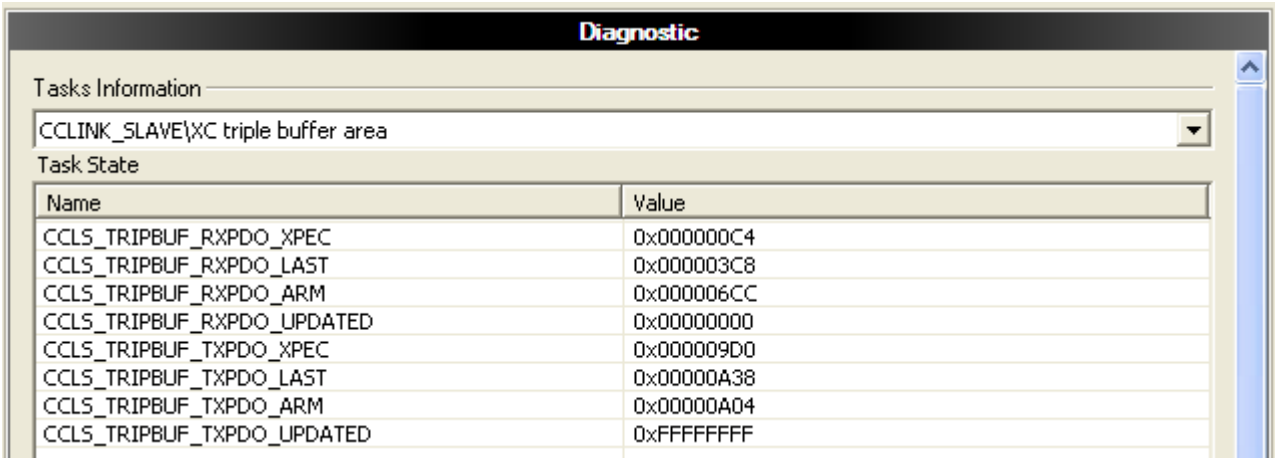


Figure 78: Extended Diagnosis > CCLINK_SLAVE > XC triple buffer area

ID	Value	Description
CCLS_TRIPBUF_RXPDO_XPEC	0 … 2^{32-1}	xPEC register of Tripple Buffer for RxPDO
CCLS_TRIPBUF_RXPDO_LAST	0 … 2^{32-1}	LAST register of Tripple Buffer for RxPDO
CCLS_TRIPBUF_RXPDO_ARM	0 … 2^{32-1}	ARM register of Tripple Buffer for RxPDO
CCLS_TRIPBUF_RXPDO_UPDATED	0 … 2^{32-1}	Updated register of Tripple Buffer for RxPDO
CCLS_TRIPBUF_TXPDO_XPEC	0 … 2^{32-1}	xPEC register of Tripple Buffer for TxPDO
CCLS_TRIPBUF_TXPDO_LAST	0 … 2^{32-1}	LAST register of Tripple Buffer for TxPDO
CCLS_TRIPBUF_TXPDO_ARM	0 … 2^{32-1}	ARM register of Tripple Buffer for TxPDO
CCLS_TRIPBUF_TXPDO_UPDATED	0 … 2^{32-1}	Updated register of Tripple Buffer for TxPDO

Table 78: Extended Diagnosis > CCLINK_SLAVE > XC triple buffer area

6.12.9 CCLINK_SLAVE/Extended transmission diagnostic

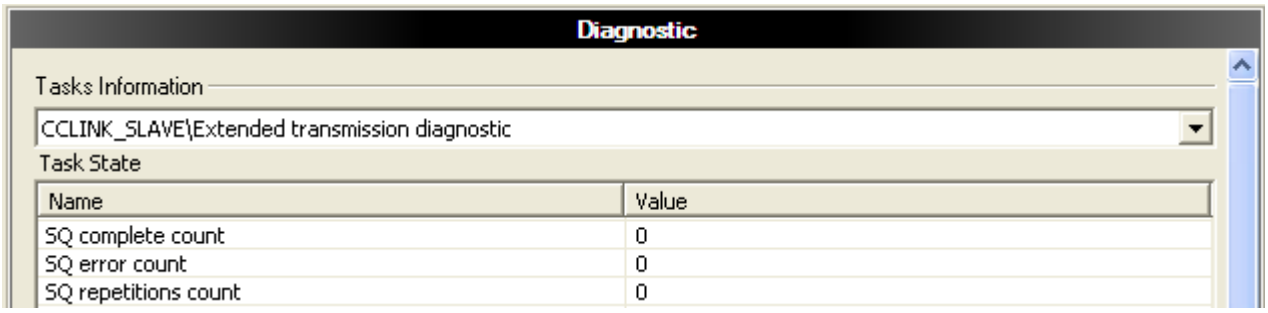


Figure 79: Extended Diagnosis > CCLINK_SLAVE > Extended transmission diagnostic

ID	Description
SQ complete count	Number of complete data transfers (SQ = sequence)
SQ error count	Number of detected sequence errors in the data transfers (SQ = sequence)
SQ repetitions count	Number of repeated data sequences (SQ = sequence)

Table 79: Extended Diagnosis > CCLINK_SLAVE > Extended transmission diagnostic

6.12.10 CCLINK_APS/Slave configuration

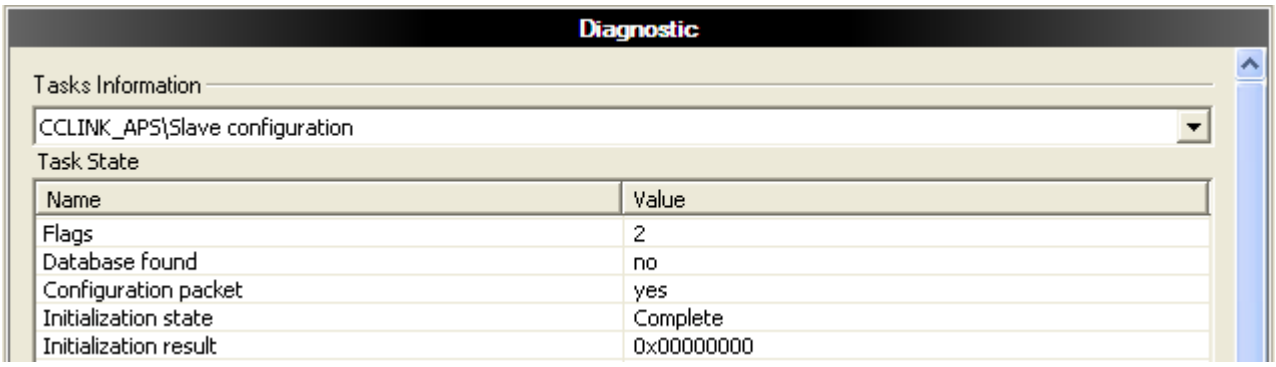


Figure 80: Extended Diagnosis > CCLINK_APS > Slave Configuration

Name	Description
Flags	Indication of flags
Database found	Indicates if the configuration database has been found (yes) or not (no)
Configuration packet	Indicates if the configuration shall be done with a configuration packet (yes) or not (no)
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	Status code for error-free initializing = 0x00000000, otherwise error code

Table 80: Extended Diagnosis > CCLINK_APS > Slave Configuration

6.12.11 CCLINK_APS/Commands

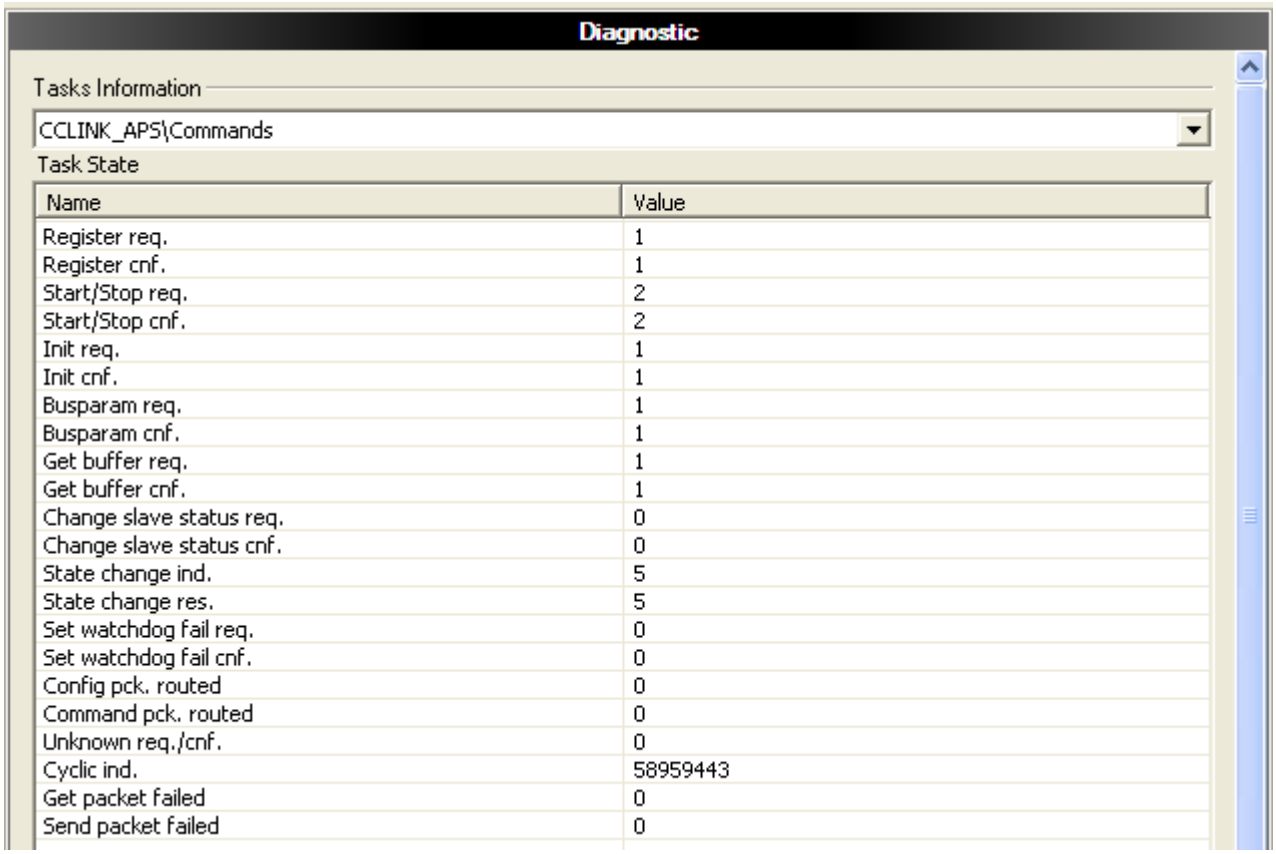


Figure 81: Extended Diagnosis > CCLINK_APS > Commands

Name	Description
[*Service*]	Diagnosis counter: Indicates the services processed. (The services of the single packets are described in the CC-Link Slave Protocol API manual [7].)
Get packet failed	Number of errors during requesting a packet
Send packet failed	Number of errors during sending a packet

Table 81: Extended Diagnosis > CCLINK_APS > Commands

6.12.12 CCLINK_APS/DPM data exchange

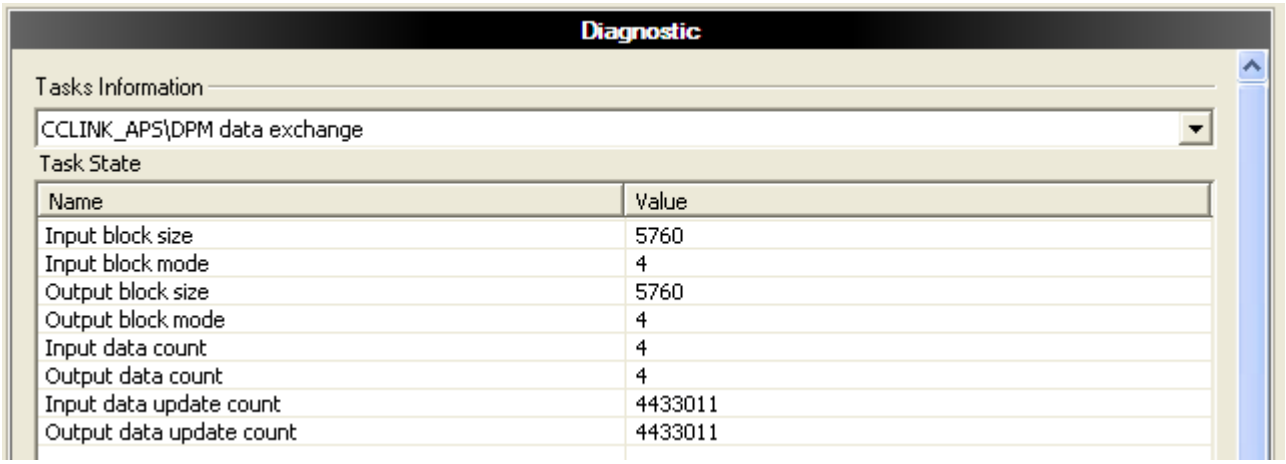


Figure 82: Extended Diagnosis > CCLINK_APS > DPM data exchange

Name	Description
Input block size	Size of the Input Data Image (cyclic data from the network) The default size of the input data image is 5760 byte.
Input block mode	Handshake Mode supported by the protocol stack, to synchronize process data exchange with the host application. 0x00 For compatibility reasons, This value is identical to 0x04 - Buffered Host Controlled IO Data Transfer 0x02 Buffered Device Controlled IO Data Transfer 0x03 Uncontrolled Mode 0x04 Buffered Host Controlled IO Data Transfer
Output block size	Size of the Output Data Image (cyclic data to the network) The default size of the output data image is 5760 byte.
Output block mode	See Input block mode
Input data count	The data count of input data in bytes
Output data count	The data count of output data in bytes
Input data update count	The total data count of input data in bytes
Output data update count	The total data count of output data in bytes

Table 82: Extended Diagnosis > CCLINK_APS > DPM data exchange

6.13 CompoNet Slave

6.13.1 Overview CompoNet Slave

Here you find an overview of the **Extended Diagnosis** for *CompoNet Slave*.

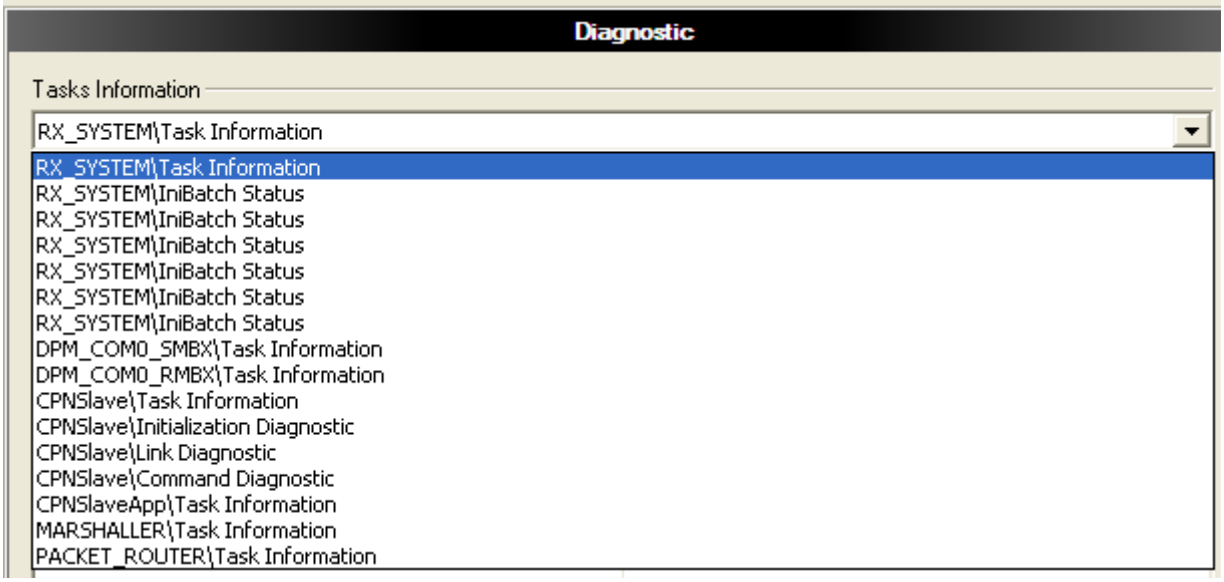


Figure 83: Extended Diagnosis CompoNet Slave (cifX, comX and netJACK, Example cifX)

Task Information Group	Taks Information	Manual Page
CPNSlave	*CPNSlave/Initialization Diagnostic*	*141*
	CPNSlave/Link Diagnostic	*142*
	CPNSlave/Command Diagnostic	*143*

Table 83: Descriptions of the Dialog Panes Extended Diagnosis CompoNet Slave

For the description of further tasks refer to section *Descriptions for Tasks with similar Functions* on page 166.

6.13.2 CPNSlave/Initialization Diagnostic

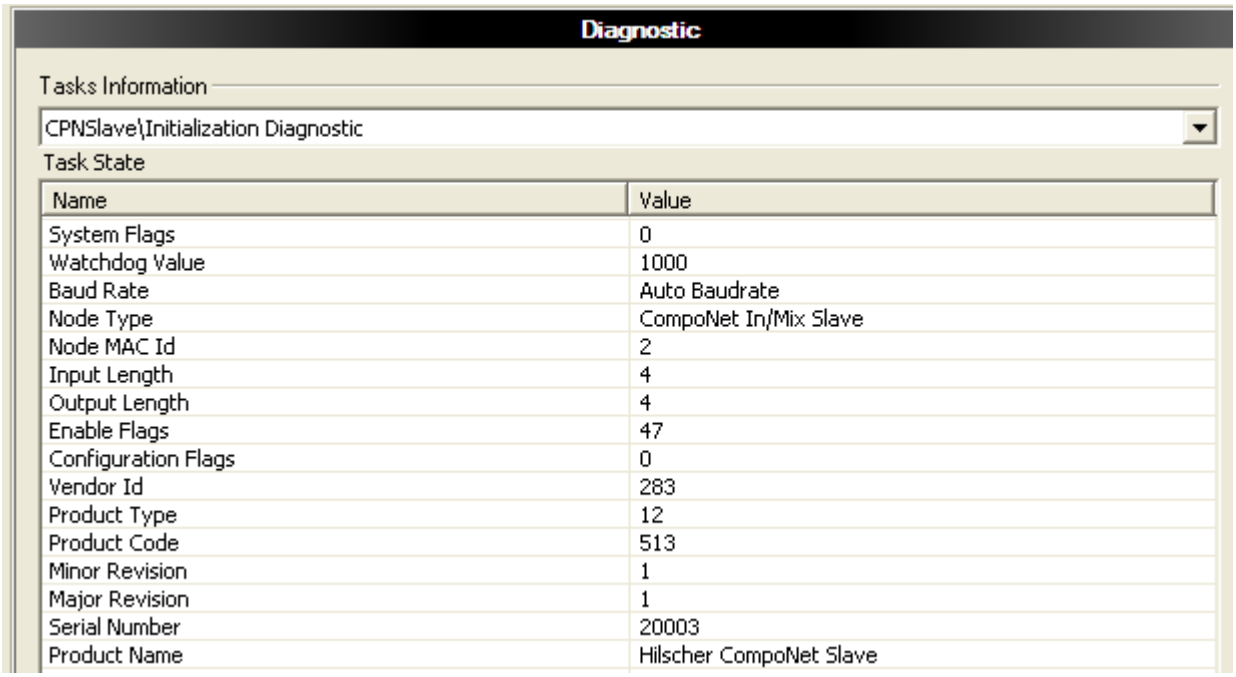


Figure 84: Extended Diagnosis > CPNSlave > Initialization Diagnostic

Name	Description
System Flags	Display of the current value of the system flags. Currently always zero (0)
Watchdog Value	The set value of the watchdog time. The value zero (0) means the watchdog must not be activated. A value greater than zero (0) indicates the watchdog time.
Baudrate	Baud rate of the CompoNet connection
Node Type	Node type of the CompoNet Slave
Node MAC Id	The MAC ID defines the CompoNet address of the device within the network
Input Length	Input length: number of input bits, value range: 2 ... 256 bits
Output Length	Initial length: number of output bits, value range: 2 ... 256 bits
Enable Flags	Number of enabled flags, which will use a default value (default)
Configuration Flags	Number of configuration flags, which must be configured
Vendor ID	Identification number of the manufacturer
Product Type	Communication Adapter
Product Code	Product code of the device, according to manufacturer's instructions
Minor Revision	Minor Revision of the task
Major Revision	Major Revision of the task
Serial Number	Serial number of the device
Product Name	The variable product name is a string (text string), which is a short description of the product / product family.

Table 84: Extended Diagnosis > CPNSlave > Initialization Diagnostic

6.13.3 CPNSlave/Link Diagnostic

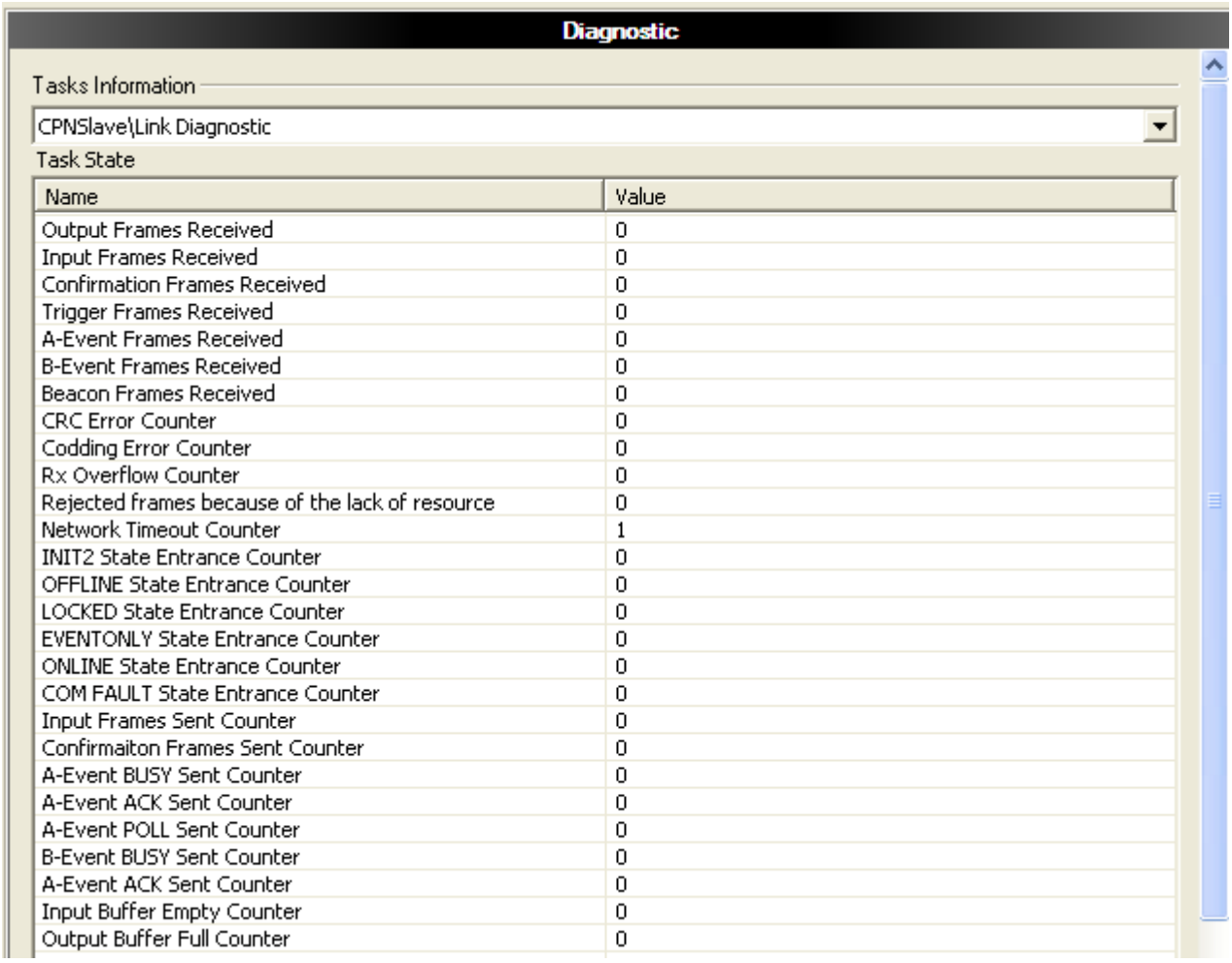


Figure 85: Extended Diagnosis > CPNSlave > Link Diagnostic

Name	Description
[*Service*]	Diagnosis counter of the Link Diagnostic layer. Indicates the services processed. (The services of the single packets are described in the CompoNet Slave Protocol API manual [12] [12].)

Table 85: Extended Diagnosis > CPNSlave > Link Diagnostic

6.13.4 CPNSlave/Command Diagnostic

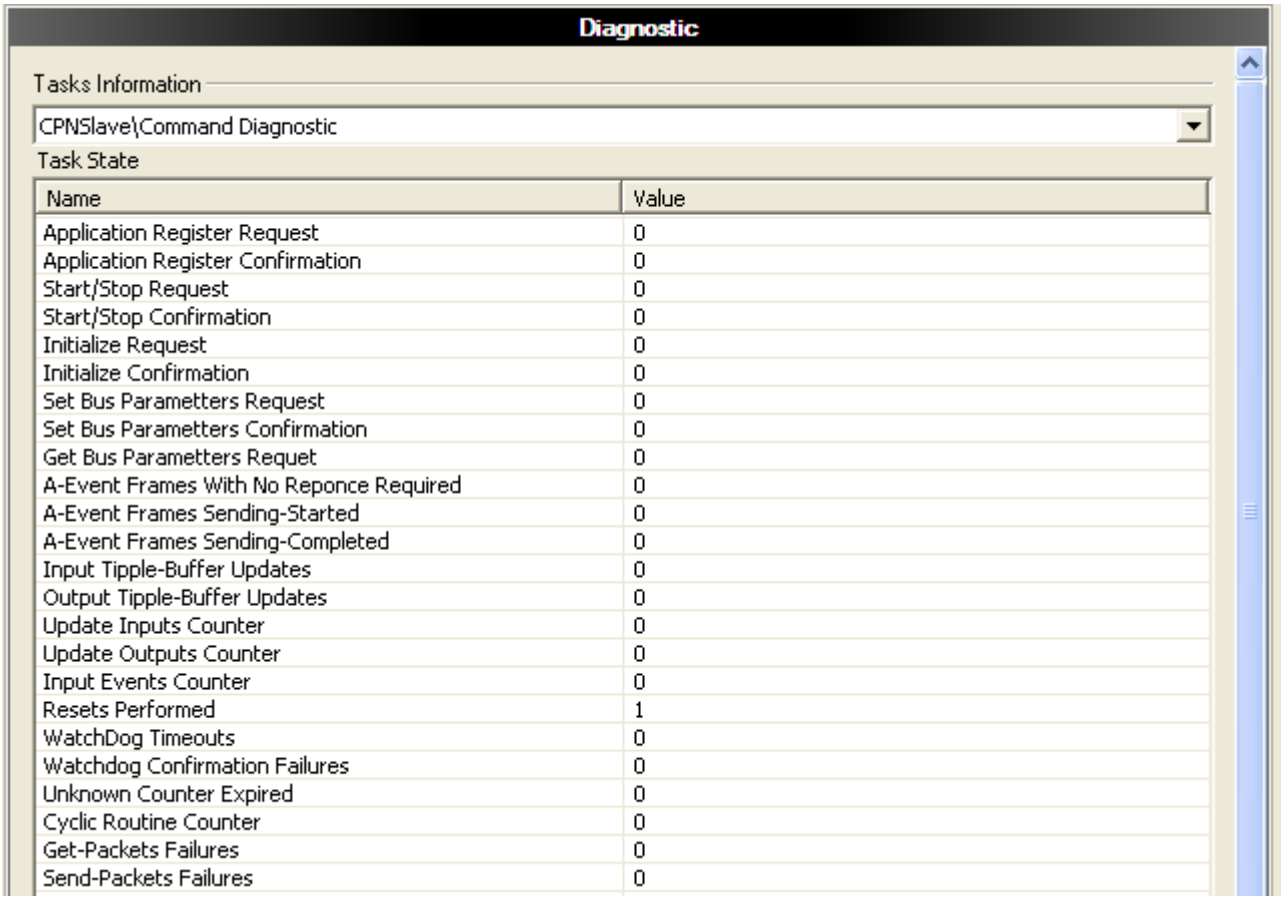


Figure 86: Extended Diagnosis > CPNSlave > Command Diagnostic

Name	Description
[*Service*]	Diagnosis counter of the Command Diagnostic layer. Indicates the services processed. (The services of the single packets are described in the CompoNet Slave Protocol API manual [12] [12].)
Get-Packet Failures	Number of errors during requesting a packet
Send Packet Failures	Number of errors during sending a packet

Table 86: Extended Diagnosis > CPNSlave > Command Diagnostic

6.14 CANopen Slave

6.14.1 Overview CANopen Slave

Here you find an overview of the **Extended Diagnosis** for *CANopen Slave*.

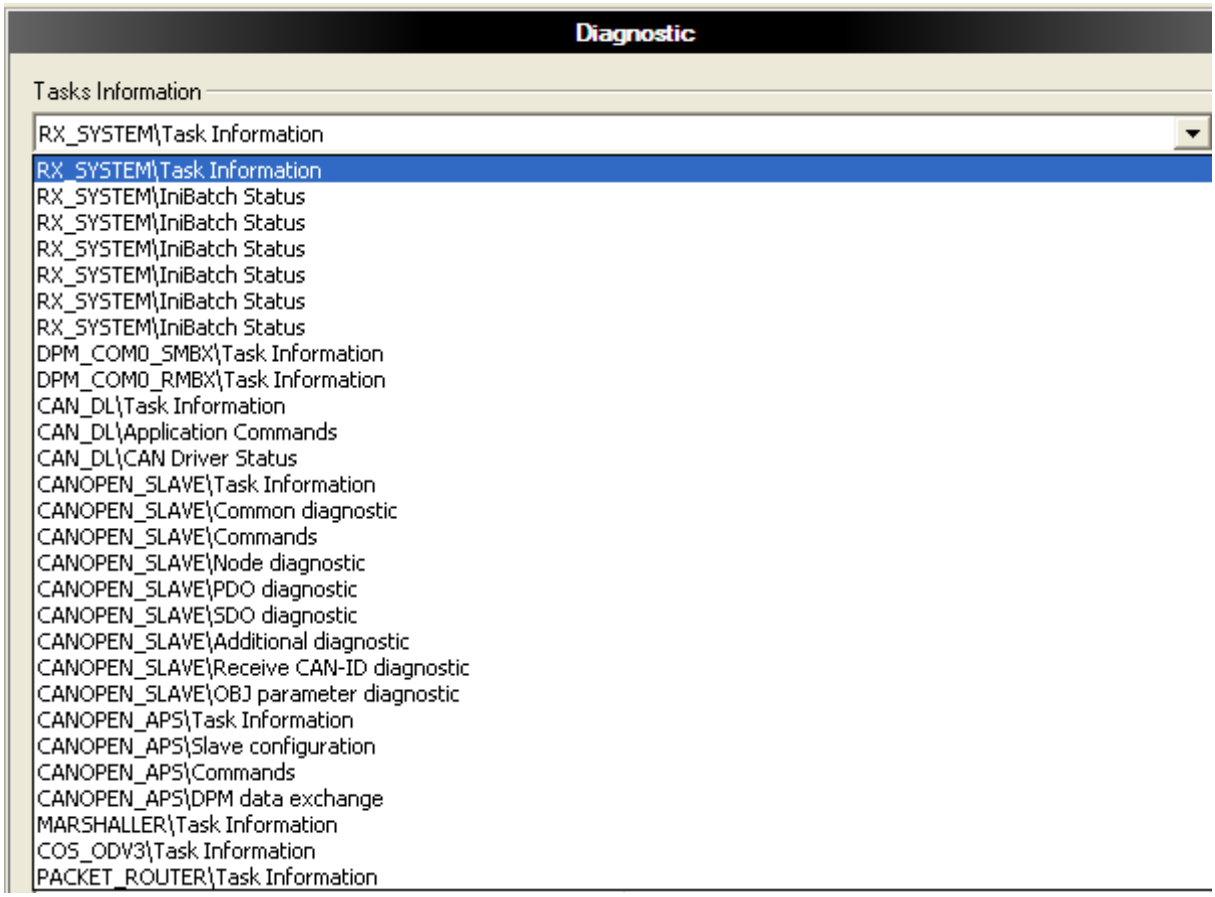


Figure 87: Extended Diagnosis CANopen Slave (cifX, comX and netJACK, Example cifX)

Task Information Group	Taks Information	Manual Page
CAN_DL	*CAN_DL/AP Commands Counter*	145
	CAN_DL/CAN Driver Status	146
CANOPEN_SLAVE	*CANOPEN_SLAVE/Common Diagnostic*	147
	CANOPEN_SLAVE/Commands	148
	CANOPEN_SLAVE/Node diagnostic	150
	CANOPEN_SLAVE/PDO diagnostic	151
	CANOPEN_SLAVE/SDO diagnostic	152
	CANOPEN_SLAVE/Additional diagnostic	153
	CANOPEN_SLAVE/Receive CAN-ID diagnostic	154
	CANOPEN_SLAVE/OBJ parameter diagnostic	156
CANOPEN_APS	*CANOPEN_APS/Slave configuration*	157
	CANOPEN_APS/Commands	158
	CANOPEN_APS/DPM data exchange	159

Table 87: Descriptions of the Dialog Panes Extended Diagnosis CANopen Slave

For the description of further tasks refer to section *Descriptions for Tasks with similar Functions* on page 166.

6.14.2 CAN_DL/AP Commands Counter

Diagnostic

Tasks Information

CAN_DL\Application Commands

Task State

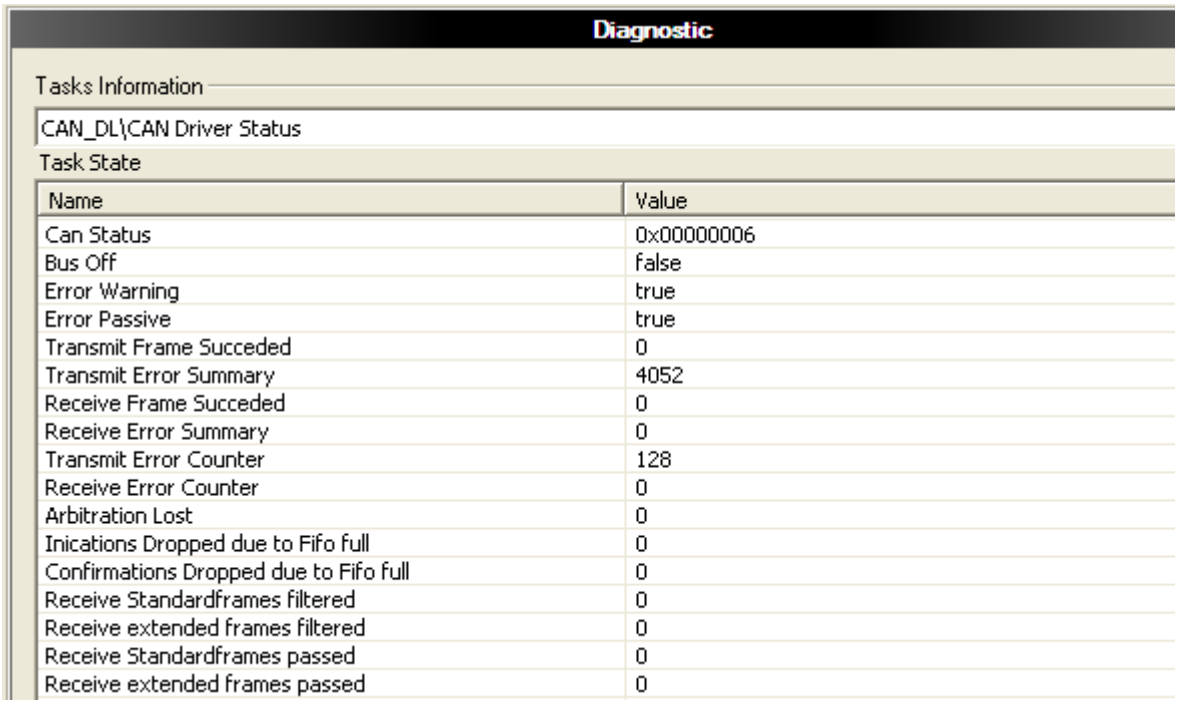
Name	Value
Data Request	3
Positive Confirmations	0
Negative Confirmations	3
Can DL Indications	0
Can DL Responses	0
Can DL Start Request	2
Positive Start Confirmations	2
Negative Start Confirmations	0
Stop Requests	2
Positive Stop Confirmations	2
Negative Stop Confirmations	0
Application Register Requests	2
Positive Application Register Confirmations	2
Negative Application Register Confirmations	0
Set Parameter Requests	2
Positive Set Parameter Confirmations	2
Negative Set Parameter Confirmations	0
Set Filter Requests	0
Positive Set Filter Confirmations	0
Negative Set Filter Confirmations	0
Enable Receive Id Requests	2
Positive Enable Receive Id Confirmations	2
Negative Enable Receive Id Confirmations	0
Event Indications	4
Event Responses	4
Event Acknowledge Request	0
Positive Event Confirmations	0
Negative Event Confirmations	0
Transmit Abort Request	0
Positive Transmit Abort Confirmations	0
Negative Transmit Abort Confirmations	0
Init Request	2
Positive Init Confirmations	2
Negative Init Confirmations	0
Hi Priority Data Request	0
Positive Hi Priority Data Confirmations	0
Negative Hi Priority Data Confirmations	0

Figure 88: Extended Diagnosis > CAN_DL > AP Commands Counter

Name	Description
[*Service*]	Diagnosis counter of the CAN layer. Indicates the services processed. (The services of the single packets are described in the CANopen Slave Protocol API manual [8].)

Table 88: Extended Diagnosis > CAN_DL > AP Commands Counter

6.14.3 CAN_DL/CAN Driver Status



Name	Value
Can Status	0x00000006
Bus Off	false
Error Warning	true
Error Passive	true
Transmit Frame Succeded	0
Transmit Error Summary	4052
Receive Frame Succeded	0
Receive Error Summary	0
Transmit Error Counter	128
Receive Error Counter	0
Arbitration Lost	0
Inications Dropped due to Fifo full	0
Confirmations Dropped due to Fifo full	0
Receive Standardframes filtered	0
Receive extended frames filtered	0
Receive Standardframes passed	0
Receive extended frames passed	0

Figure 89: Extended Diagnosis > CAN_DL > CAN Driver Status

Name	Description
CAN Status Bus Off Error Warning ErrorPassive	Diagnosis status of CAN specific error levels. Indicates the respective status of the CAN bus. (For further information refer to the CANopen Slave Protocol API manual [8].)
[*Service*]	Diagnosis counter of CAN errors. Indicates the services processed. (The services of the single packets are described in the CANopen Slave Protocol API manual [8].)

Table 89: Extended Diagnosis > CAN_DL > CAN Driver Status

6.14.4 CANOPEN_SLAVE/Common Diagnostic

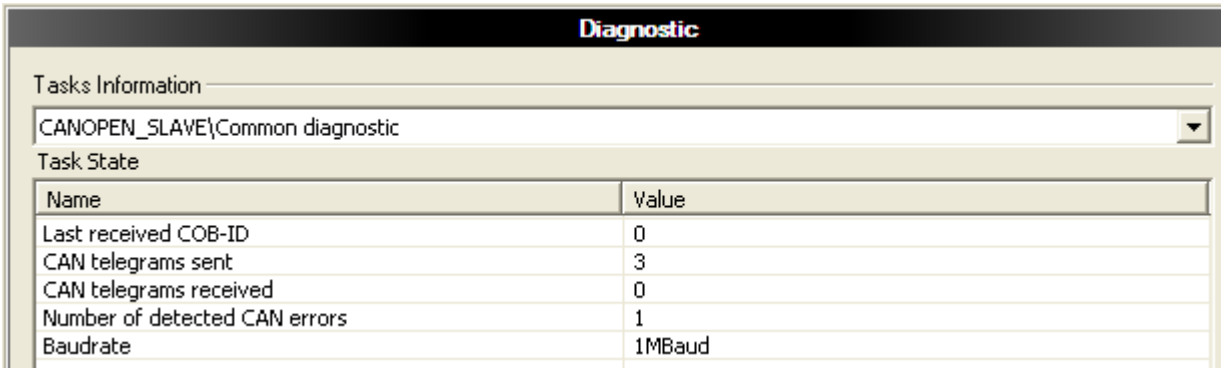


Figure 90: Extended Diagnosis > CANOPEN_SLAVE > Common Diagnostic

Name	Description	
Last received COB-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	
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

Table 90: Extended Diagnosis > CANOPEN_SLAVE > Common Diagnostic

6.14.5 CANOPEN_SLAVE/Commands

Diagnostic

Tasks Information

CANOPEN_SLAVE\Commands

Task State

Name	Value
Register req.	1
Register cnf.	1
Start/Stop req.	2
Start/Stop cnf.	2
Busparam req.	1
Busparam cnf.	1
Get buffer req.	1
Get buffer cnf.	1
State change ind.	4579
State change res.	4579
Set watchdog fail req.	0
Set watchdog fail cnf.	0
Data exch. req.	0
Data exch. 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.	2
CAN_DL event res.	2
CAN_DL send data req.	3
CAN_DL send data cnf. pos.	3
CAN_DL send data cnf. neg.	0
CAN_DL enable id req.	2
CAN_DL enable id cnf. pos.	2
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.	0
CAN_DL recv data res.	0
Unknown req./cnf.	0
Cyclic ind.	990367
Get packet failed	0
Send packet failed	2

Figure 91: Extended Diagnosis > CANOPEN_SLAVE > Commands

Name	Description
[*Service*]	Diagnosis counter: Indicates the services processed. (The services of the single packets are described in the CANopen Slave Protocol API manual [8].)
Get packet failed	Number of errors during requesting a packet
Send packet failed	Number of errors during sending a packet

Table 91: Extended Diagnosis > CANOPEN_SLAVE > Commands

6.14.6 CANOPEN_SLAVE/Node diagnostic

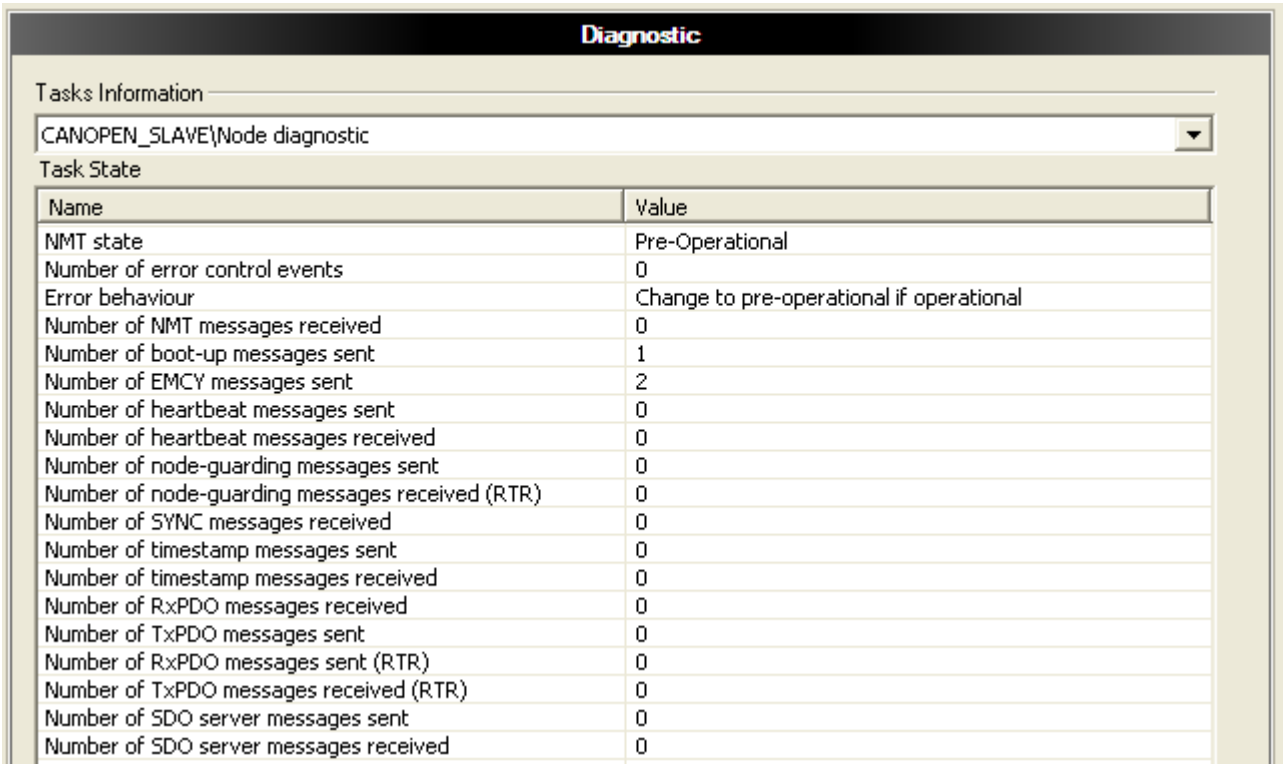


Figure 92: Extended Diagnosis > CANOPEN_SLAVE > Node diagnostic

Name	Description	
NMT state	Possible NMT states: - Reset (at the beginning - reset node, later on - reset communication) - Stopped - Operational - Pre-operational NMT = Network Management (For further information refer to the CANopen Slave Protocol API manual [8].)	
Error behaviour	Possible Error behaviour - Change to pre-operational if operational - No change of NMT state - Change to stopped	
[*Services*]	Number of error control events	Diagnosis counter
	Number of NMT messages received Number of boot-up messages sent Number of EMCY messages sent Number of heartbeat messages sent Number of heartbeat messages received Number of node-guarding messages sent Number of node-guarding messages received (RTR) Number of SYNC messages received Number of timestamp messages sent Number of timestamp messages received Number of RxPDO messages received Number of TxPDO messages sent Number of RxPDO messages sent (RTR) Number of TxPDO messages received (RTR) Number of SDO server messages sent Number of SDO server messages received	Internal diagnosis counter: Indicate the send/received messages.

Table 92: Extended Diagnosis > CANOPEN_SLAVE > Node diagnostic

6.14.7 CANOPEN_SLAVE/PDO diagnostic

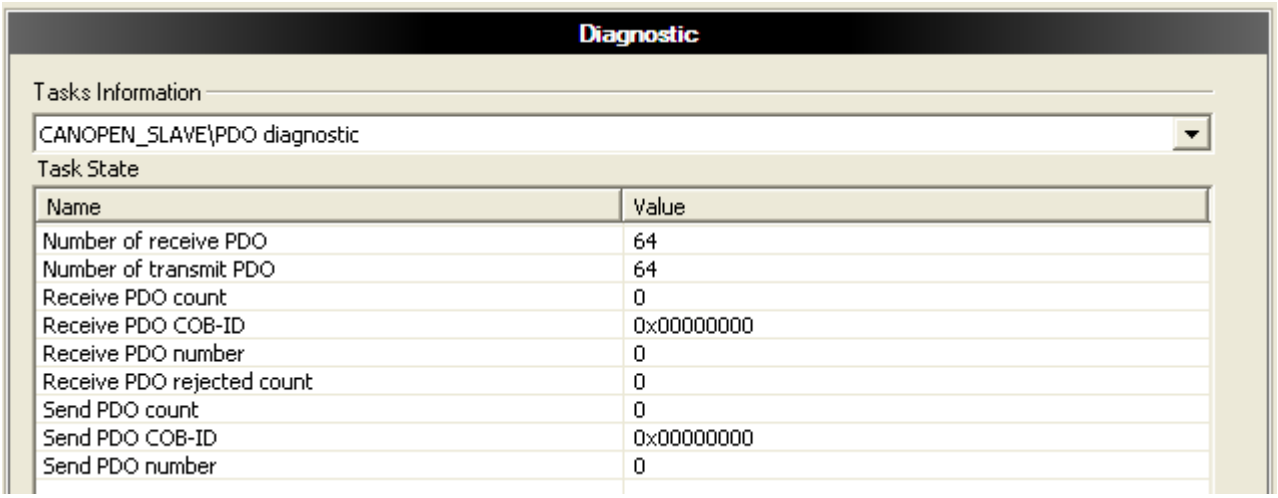


Figure 93: Extended Diagnosis > CANOPEN_SLAVE > PDO diagnostic

Name	Description
[Services]	Diagnosis counter: Indicate the services processed. (The services of the single packets are described in the CANopen Slave Protocol API manual [8].)
Number of receive PDOs	Number of the used receive PDOs
Number of transmit PDOs	Number of the used send PDOs
Receive PDO number	Number of the last received PDO
Receive PDO COB-ID	CAN identifier related to the COB (Communication Object) of the last received PDO. Receive PDOs are output data of the Master and are received from the node. (For further information refer to CANopen Slave Protocol API manual [8].)
Send PDO COB-ID	CAN identifier related to the COB (Communication Object) of the last sent PDO. Send PDOs are input data of the Master and are send by the node. (For further information refer to CANopen Slave Protocol API manual [8].)

Table 93: Extended Diagnosis > CANOPEN_SLAVE > PDO diagnostic

6.14.8 CANOPEN_SLAVE/SDO diagnostic

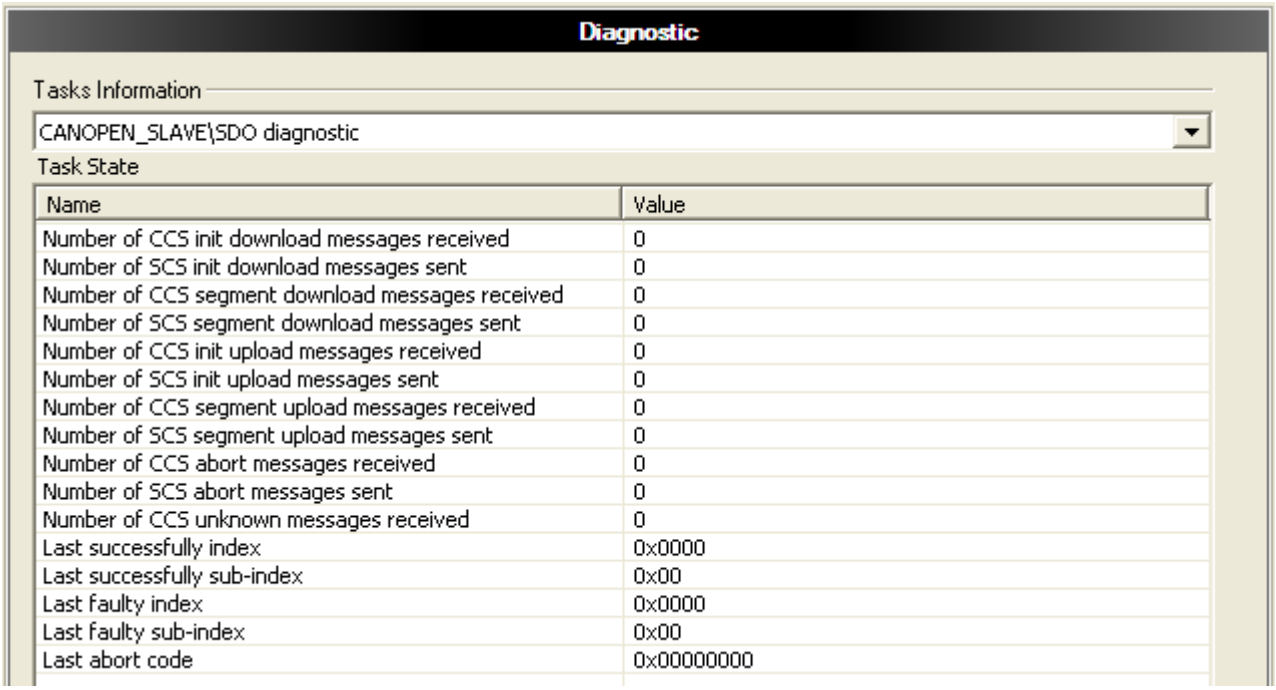


Figure 94: Extended Diagnosis > CANOPEN_SLAVE > SDO diagnostic

Name	Description
[*Service*]	Diagnosis counter: Indicate the services processed. (The services of the single packets are described in the CANopen Slave Protocol API manual [8].)
Last successfully index Last successfully sub-index Last faulty index Last faulty sub-index Last abort code	Last index/sub-index of the successful or faulty send/received service data object (SDO) and code for abort.

Table 94: Extended Diagnosis > CANOPEN_SLAVE > SDO diagnostic

6.14.9 CANOPEN_SLAVE/Additional diagnostic

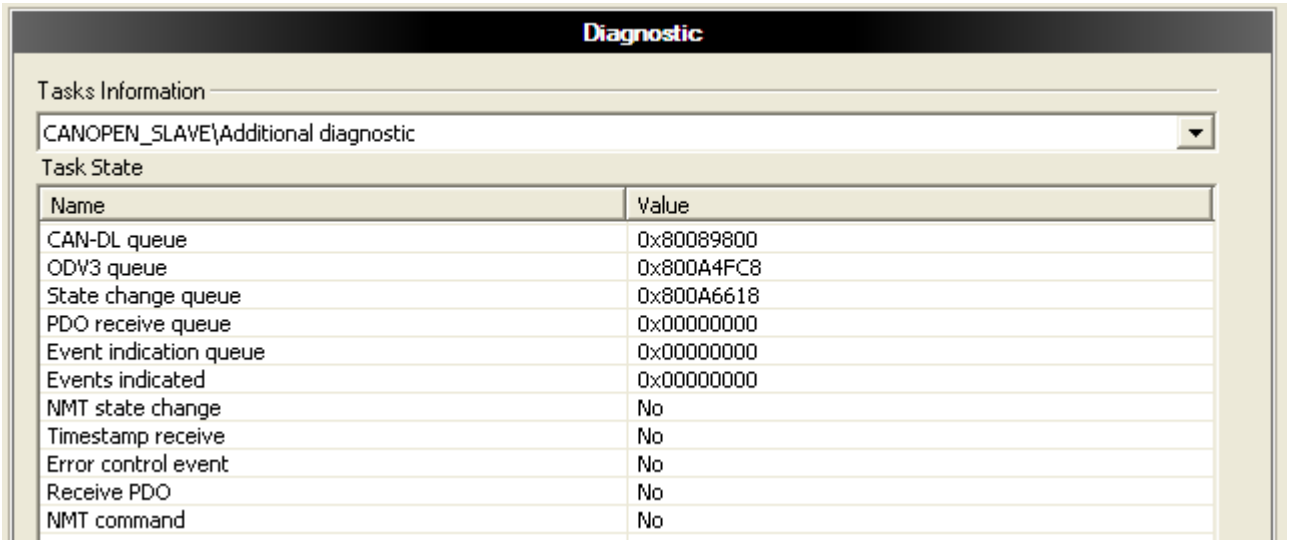


Figure 95: Extended Diagnosis > CANOPEN_SLAVE > Additional diagnostic

Name	Description
CAN-DL queue ODV3 queue State change queue PDO receive queue Event indication queue	Handle to another queue for the internal packet communication (to be used for trouble shooting)
Events indicated	Event Bit-list (1 Bit per event) to indicate the events in hexadecimal form Example: NMT state change: *Yes* Timestamp receive: *Yes* Error control event: *Yes* Receive PDO: *Yes* NMT command: *Yes* Then “Events indicated” = 0x00000001F
NMT state change Timestamp receive Error control event Receive PDO NMT command	Events, indicated to the application No: This event is not indicated to the application. Yes: this event is indicated to the application.

Table 95: Extended Diagnosis > CANOPEN_SLAVE > Additional diagnostic

6.14.10 CANOPEN_SLAVE/Receive CAN-ID diagnostic

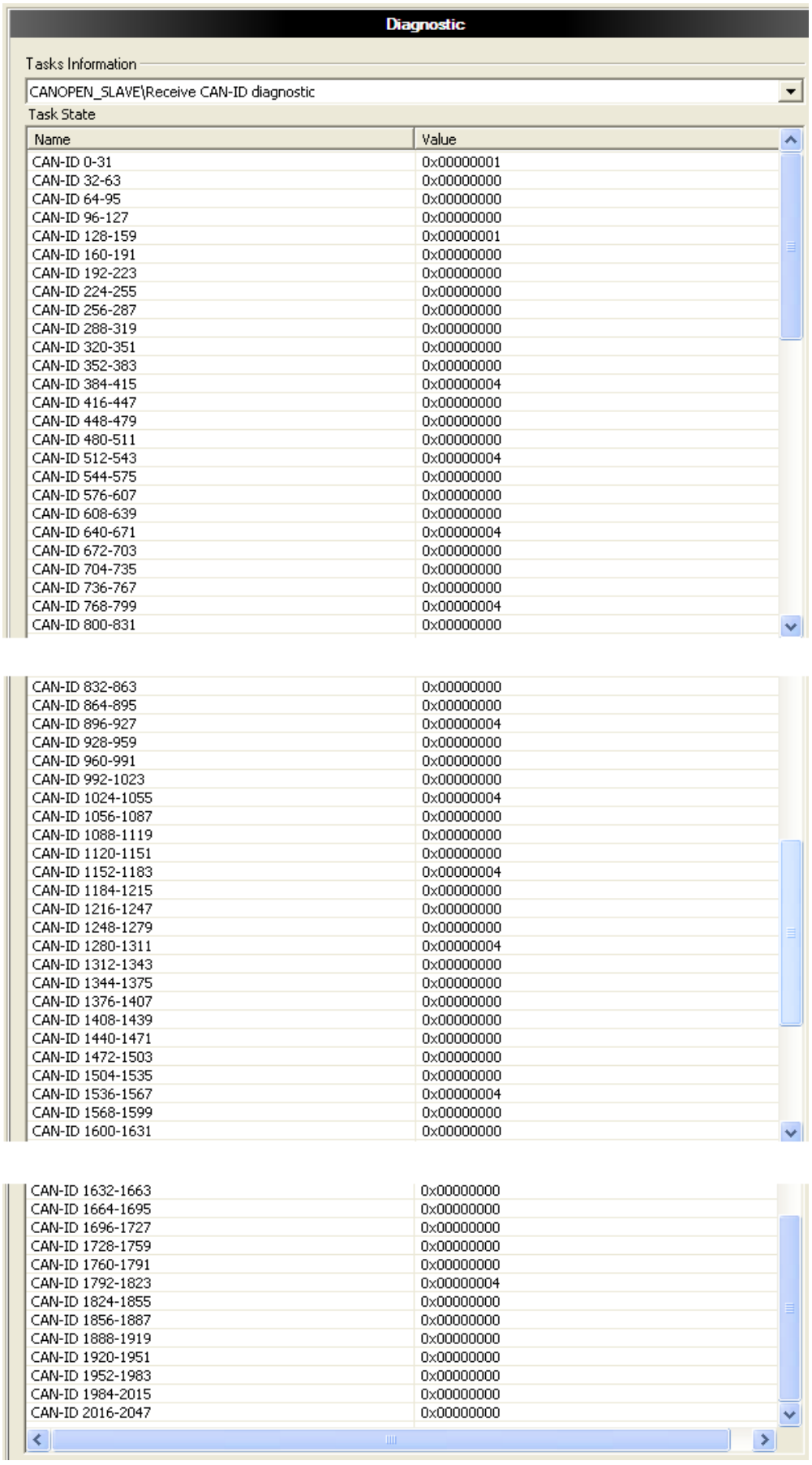


Figure 96: Extended Diagnosis > CANOPEN_SLAVE > Receive CAN-ID diagnostic

Name	Description
CAN-ID 0-31 .. CAN-ID 2016-2047	One bit-field for each CAN-ID, which is currently processed by the stack. Range of values for the CAN-ID: CAN-ID 0 to CAN-ID 2047 Example: Name / Value / Meaning CAN-ID **0**-31 / 0x0000000**1** / CAN-ID **0** has been processed CAN-ID 32-63 / 0x00000000 / no CAN-ID has been processed CAN-ID 64-95 / 0x00000000 / no CAN-ID has been processed CAN-ID 96-127 / 0x00000000 / no CAN-ID has been processed CAN-ID **128**-159 / 0x0000000**1** / CAN-ID **128** has been processed .. CAN-ID 384-415 / 0x0000000**4** / CAN-ID **386** has been processed .. CAN-ID 2016-2047 / 0x00000000 / no CAN-ID has been processed

Table 96: Extended Diagnosis > CANOPEN_SLAVE > Receive CAN-ID diagnostic

6.14.11 CANOPEN_SLAVE/OBJ parameter diagnostic

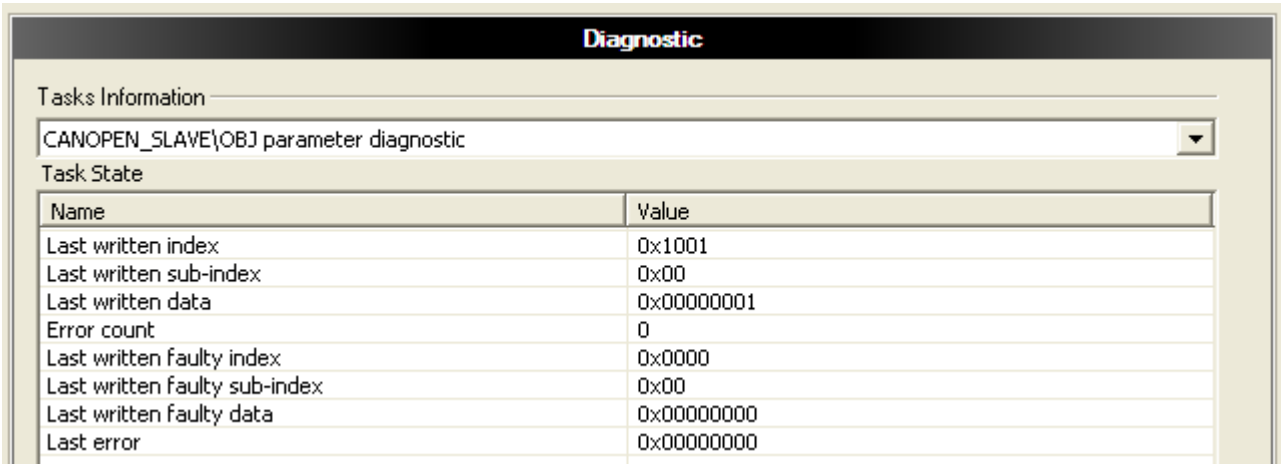


Figure 97: Extended Diagnosis > CANOPEN_SLAVE > OBJ parameter diagnostic

Name	Description
Last written index Last written sub-index Last written data	Last written index/sub-index/data of the OBJ parameter diagnostic.
Error count	Diagnosis counter (described in the CANopen Slave Protocol API manual [8].)
Last written faulty index Last written faulty sub-index Last written faulty data Last Error	Last written faulty index/sub-index/data and last error of the OBJ parameter diagnostic.

Table 97: Extended Diagnosis > CANOPEN_SLAVE > OBJ parameter diagnostic

6.14.12 CANOPEN_APS/Slave configuration

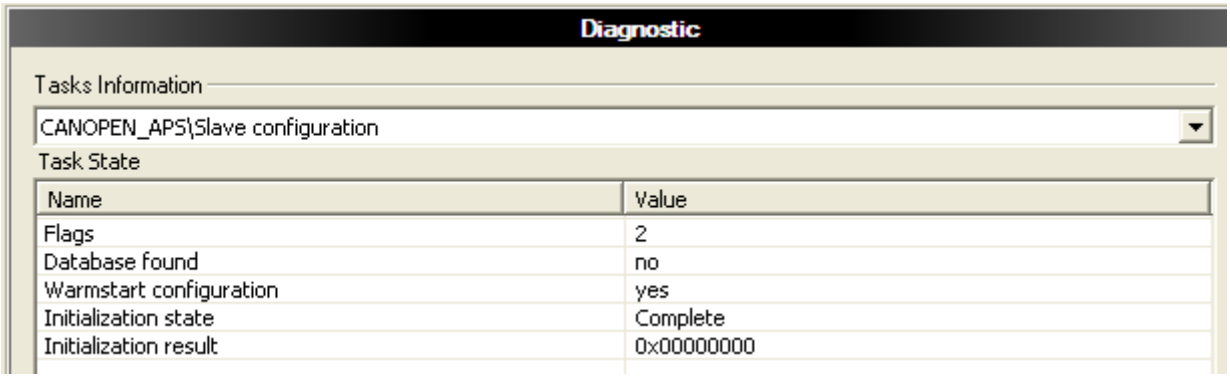
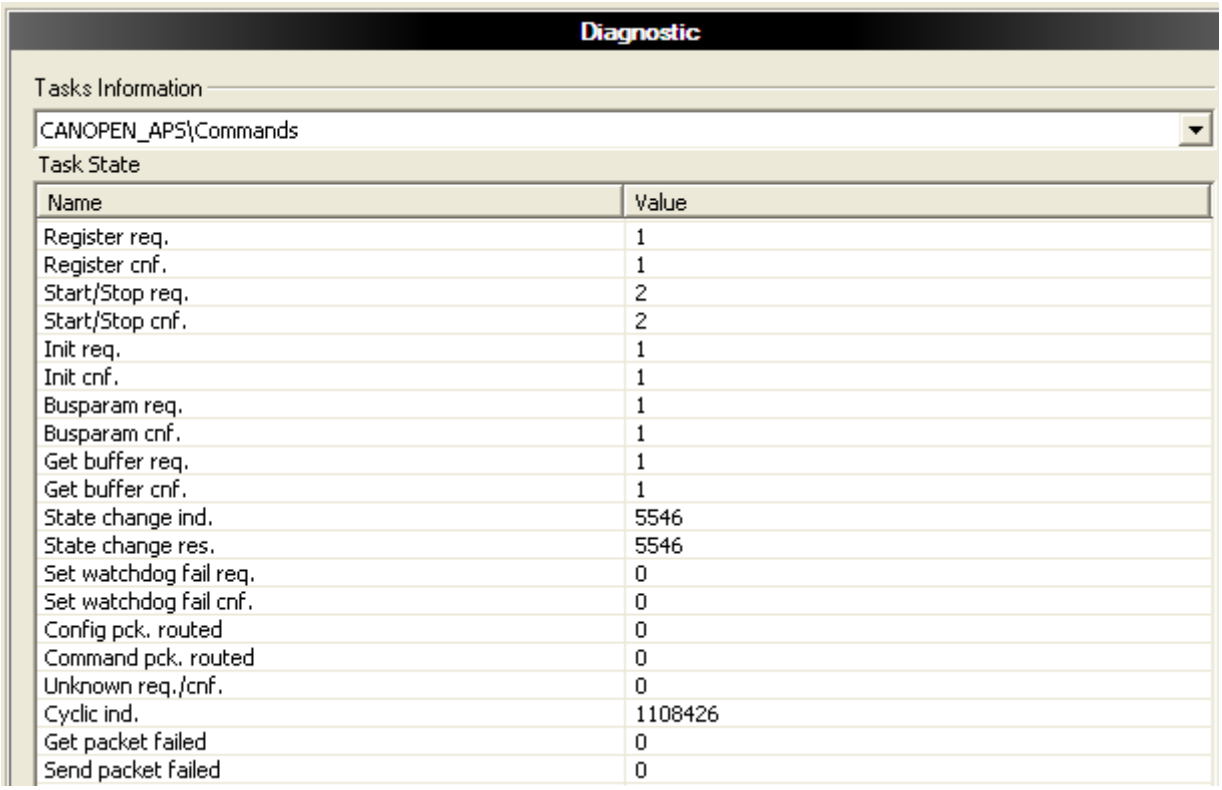


Figure 98: Extended Diagnosis > CANOPEN_APS > Slave Configuration

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	Error code of the initialization, 0 = no error

Table 98: Extended Diagnosis > CANOPEN_APS > Slave Configuration

6.14.13 CANOPEN_APS/Commands



Name	Value
Register req.	1
Register cnf.	1
Start/Stop req.	2
Start/Stop cnf.	2
Init req.	1
Init cnf.	1
Busparam req.	1
Busparam cnf.	1
Get buffer req.	1
Get buffer cnf.	1
State change ind.	5546
State change res.	5546
Set watchdog fail req.	0
Set watchdog fail cnf.	0
Config pck. routed	0
Command pck. routed	0
Unknown req./cnf.	0
Cyclic ind.	1108426
Get packet failed	0
Send packet failed	0

Figure 99: Extended Diagnosis > CANOPEN_APS > Commands

Name	Description
[*Service*]	Diagnosis counter: Indicates the services processed. (The services of the single packets are described in the CANopen Slave Protocol API manual [8].)
Get packet failed	Number of errors during requesting a packet
Send packet failed	Number of errors during sending a packet

Table 99: Extended Diagnosis > CANOPEN_APS > Commands

6.14.14 CANOPEN_APS/DPM data exchange

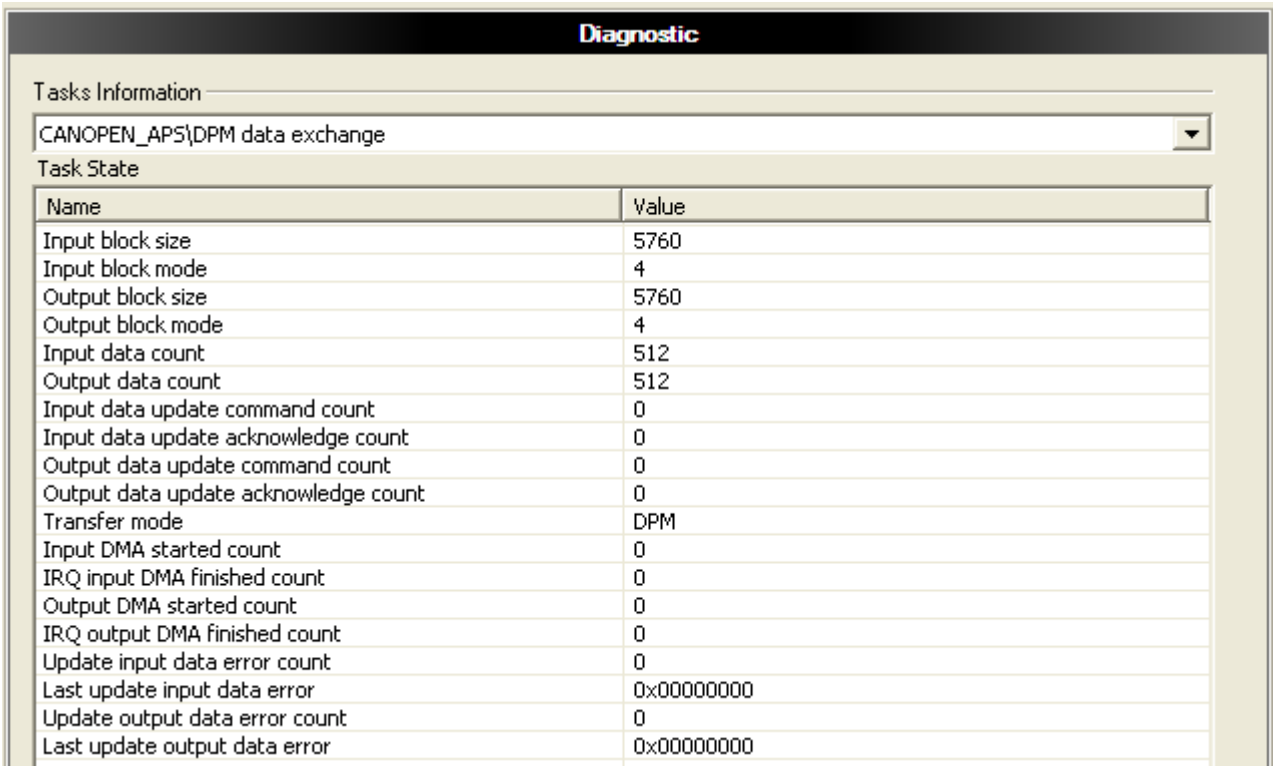


Figure 100: Extended Diagnosis > CANOPEN_APS > DPM data exchange

Name	Description
[*Services*]	Diagnosis counter: Indicate the services processed. (The services of the single packets are described in the CANopen Slave Protocol API manual [8].)
Input block size	Block size of the receive data in Byte
Input block mode	value (read out from the device) = 4 (host control)
Output block size	Block size of the send data in Byte
Output block mode	value (read out from the device) = 4 (host control)
Input data count	Diagnosis counter: Indicates the input data as an UINT 32 value*. * 1 UINT 32 = 4 Byte
Output data count	Diagnosis counter: Indicates the output data as an UINT 32 value*. * 1 UINT 32 = 4 Byte
Transfer mode	Possible values: “DPM”, “DMA” DPM = Dual-Port Memory DMA = Direct Memory Access
Last update input data error	Hexadecimal number identifying the last error occurred during the receive data are updated.
Last update output data error	Hexadecimal number identifying the last error occurred during the send data are updated.

Table 100: Extended Diagnosis > CANOPEN_APS > Commands

6.15 DeviceNet Slave

6.15.1 Overview DeviceNet Slave

Here you find an overview of the **Extended Diagnosis** for *DeviceNet Slave*.

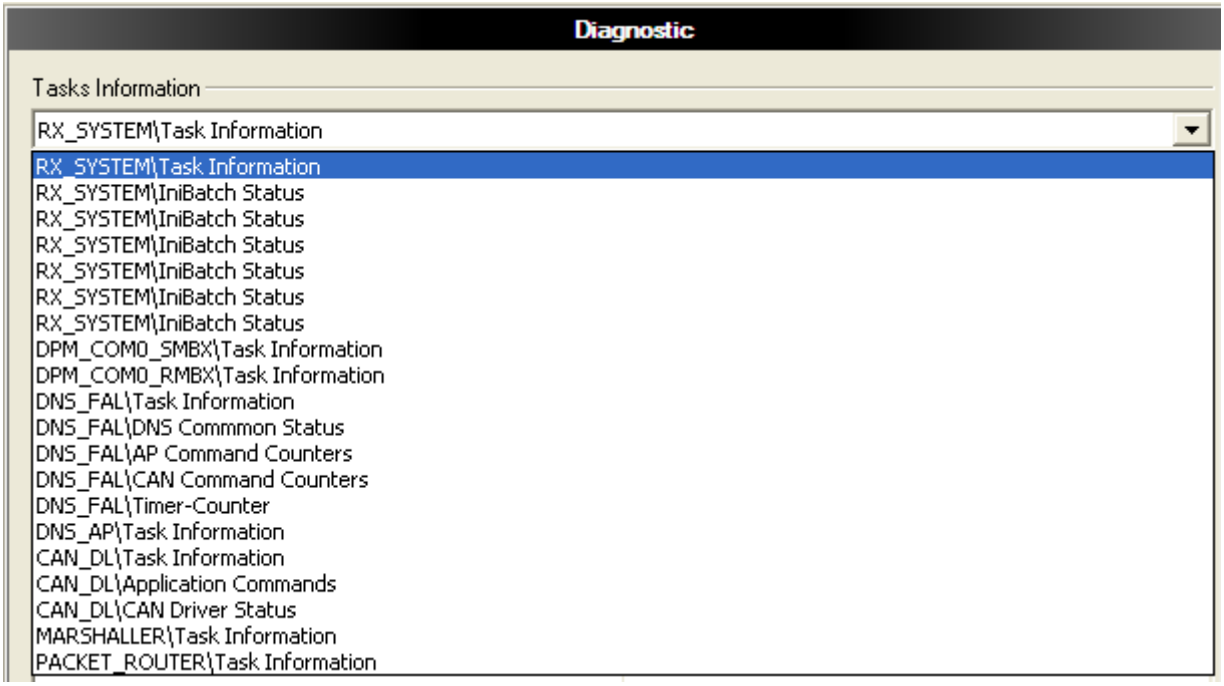


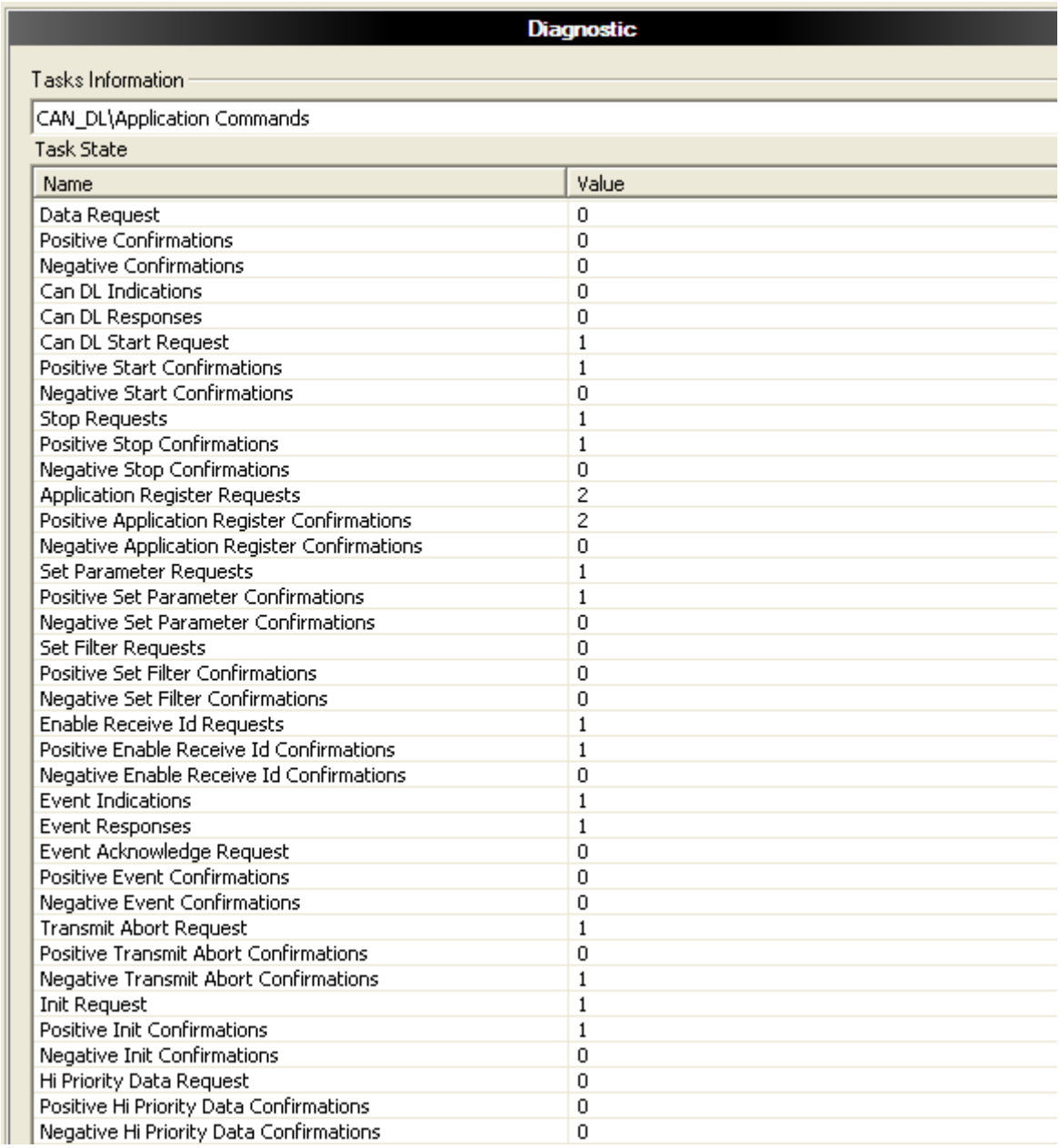
Figure 101: Extended Diagnosis DeviceNet Slave (cifX, comX and netJACK, Example cifX)

Task Information Group	Taks Information	Manual Page
CAN_DL	*CAN_DL/AP Commands Counter*	*161*
	CAN_DL/CAN Driver Status	*162*
DNS_FAL	*DNS_FAL/DNS Common Status*	*163*
	DNS_FAL/AP Commands Counter	*164*
	DNS_FAL/CAN Command Counter	*164*
	DNS_FAL/Timer Counter	*165*

Table 101: Descriptions of the Dialog Panes Extended Diagnosis DeviceNet Slave

For the description of further tasks refer to section *Descriptions for Tasks with similar Functions* on page 166.

6.15.2 CAN_DL/AP Commands Counter



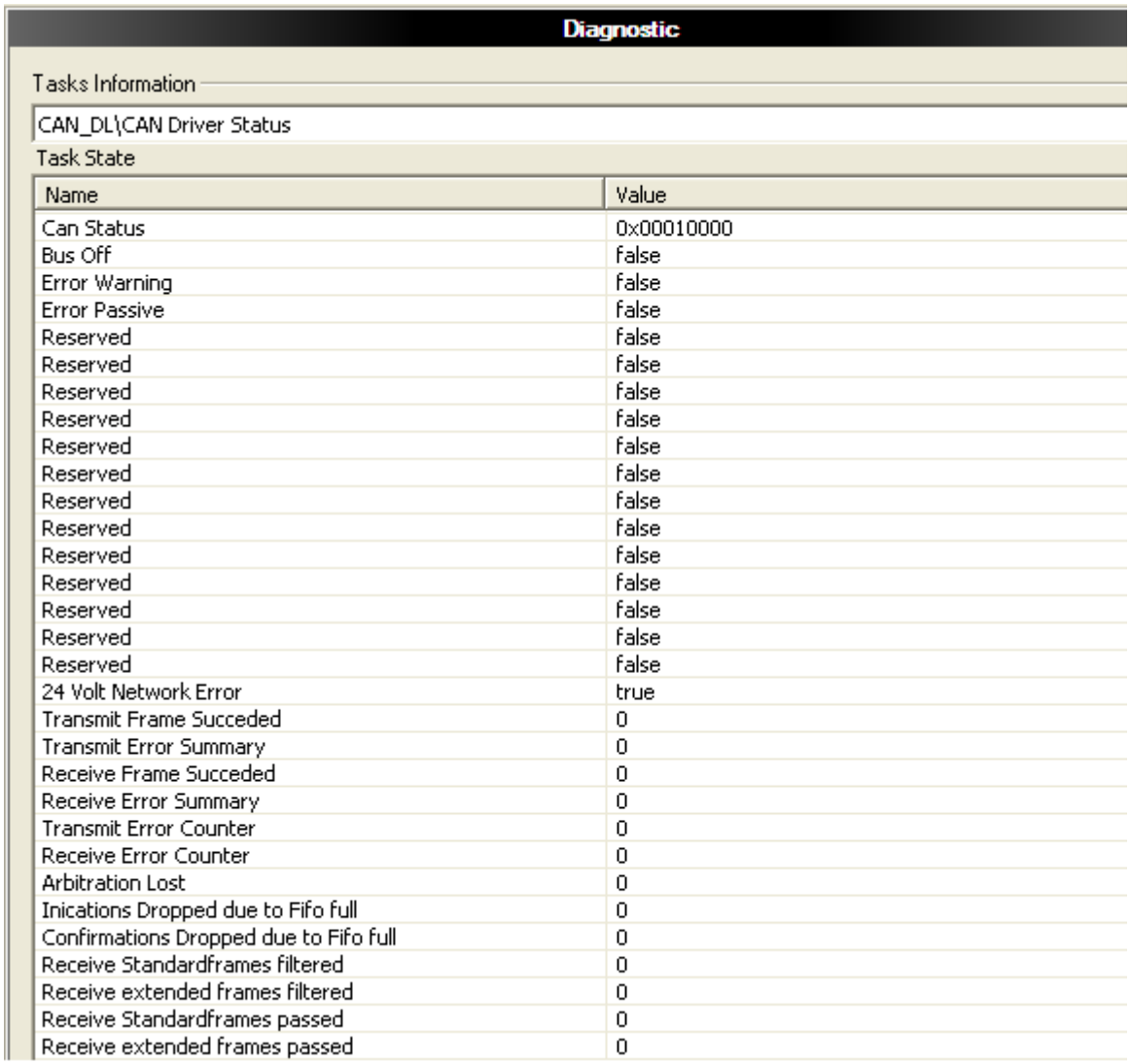
Name	Value
Data Request	0
Positive Confirmations	0
Negative Confirmations	0
Can DL Indications	0
Can DL Responses	0
Can DL Start Request	1
Positive Start Confirmations	1
Negative Start Confirmations	0
Stop Requests	1
Positive Stop Confirmations	1
Negative Stop Confirmations	0
Application Register Requests	2
Positive Application Register Confirmations	2
Negative Application Register Confirmations	0
Set Parameter Requests	1
Positive Set Parameter Confirmations	1
Negative Set Parameter Confirmations	0
Set Filter Requests	0
Positive Set Filter Confirmations	0
Negative Set Filter Confirmations	0
Enable Receive Id Requests	1
Positive Enable Receive Id Confirmations	1
Negative Enable Receive Id Confirmations	0
Event Indications	1
Event Responses	1
Event Acknowledge Request	0
Positive Event Confirmations	0
Negative Event Confirmations	0
Transmit Abort Request	1
Positive Transmit Abort Confirmations	0
Negative Transmit Abort Confirmations	1
Init Request	1
Positive Init Confirmations	1
Negative Init Confirmations	0
Hi Priority Data Request	0
Positive Hi Priority Data Confirmations	0
Negative Hi Priority Data Confirmations	0

Figure 102: Extended Diagnosis > CAN_DL > AP Commands Counter

Name	Description
[*Service*]	Diagnosis counter of the CAN layer. Indicates the services processed. (The services of the single packets are described in the DeviceNet Slave Protocol API manual [11].)

Table 102: Extended Diagnosis > CAN_DL > AP Commands Counter

6.15.3 CAN_DL/CAN Driver Status



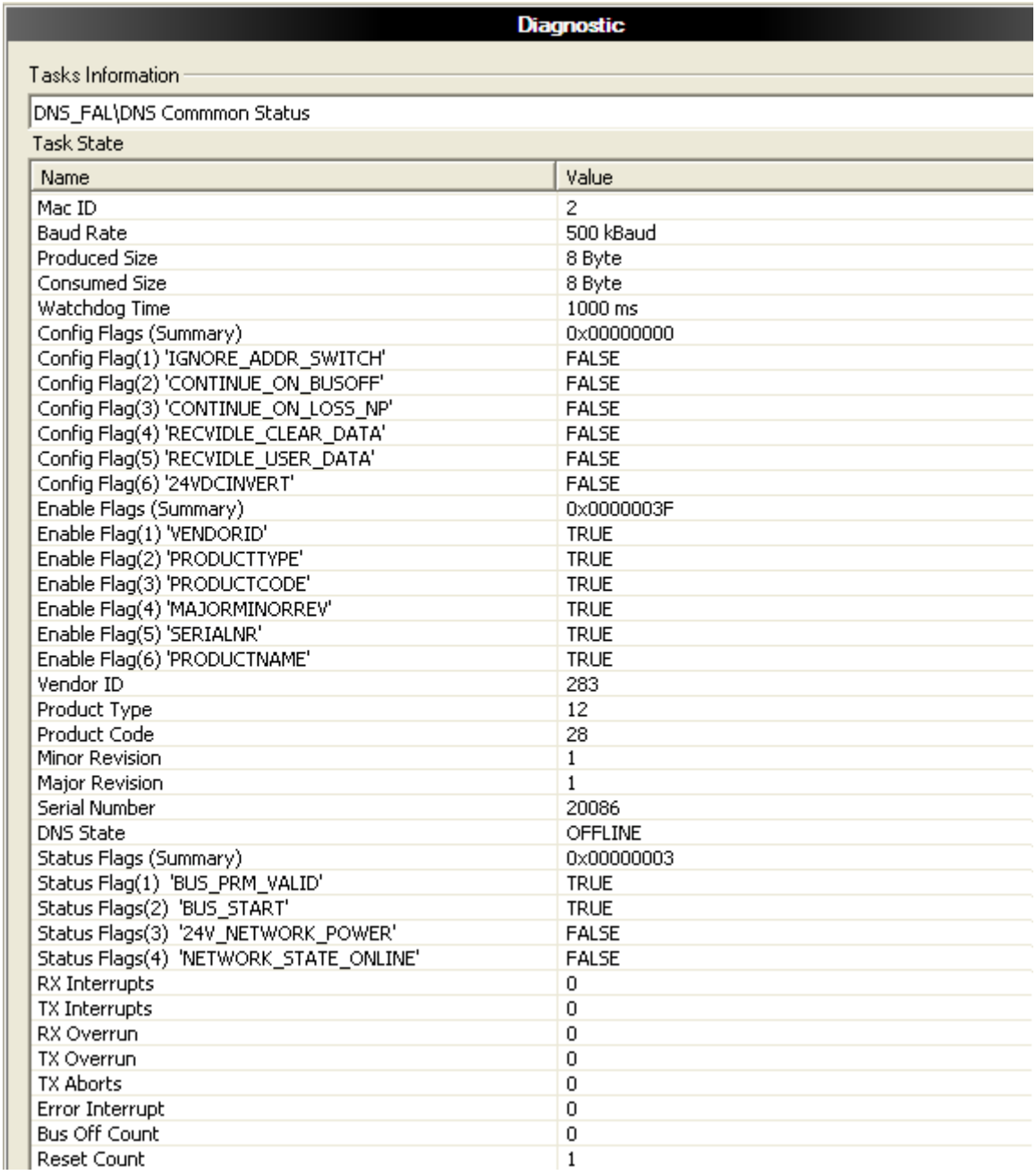
Name	Value
Can Status	0x00010000
Bus Off	false
Error Warning	false
Error Passive	false
Reserved	false
Reserved	false
Reserved	false
Reserved	false
Reserved	false
Reserved	false
Reserved	false
Reserved	false
Reserved	false
Reserved	false
Reserved	false
Reserved	false
Reserved	false
Reserved	false
Reserved	false
24 Volt Network Error	true
Transmit Frame Succeded	0
Transmit Error Summary	0
Receive Frame Succeded	0
Receive Error Summary	0
Transmit Error Counter	0
Receive Error Counter	0
Arbitration Lost	0
Inications Dropped due to Fifo full	0
Confirmations Dropped due to Fifo full	0
Receive Standardframes filtered	0
Receive extended frames filtered	0
Receive Standardframes passed	0
Receive extended frames passed	0

Figure 103: Extended Diagnosis > CAN_DL > CAN Driver Status

Name	Description
CAN Status Bus Off Error Warning ErrorPassive	Diagnosis status of CAN specific error levels. Indicates the respective status of the CAN bus. (For further information refer to the DeviceNet Slave Protocol API manual [11].)
Reserved	Diagnosis status
24 Volt Network Error	Diagnosis status of DeviceNet specific error levels. (For further information refer to the DeviceNet Slave Protocol API manual [11].)
[*Service*]	Diagnosis counter of CAN errors. Indicates the services processed. (The services of the single packets are described in the DeviceNet Slave Protocol API manual [11].)

Table 103: Extended Diagnosis > CAN_DL > CAN Driver Status

6.15.4 DNS_FAL/DNS Common Status



Name	Value
Mac ID	2
Baud Rate	500 kBaud
Produced Size	8 Byte
Consumed Size	8 Byte
Watchdog Time	1000 ms
Config Flags (Summary)	0x00000000
Config Flag(1) 'IGNORE_ADDR_SWITCH'	FALSE
Config Flag(2) 'CONTINUE_ON_BUSOFF'	FALSE
Config Flag(3) 'CONTINUE_ON_LOSS_NP'	FALSE
Config Flag(4) 'RECVIDLE_CLEAR_DATA'	FALSE
Config Flag(5) 'RECVIDLE_USER_DATA'	FALSE
Config Flag(6) '24VDCINVERT'	FALSE
Enable Flags (Summary)	0x0000003F
Enable Flag(1) 'VENDORID'	TRUE
Enable Flag(2) 'PRODUCTTYPE'	TRUE
Enable Flag(3) 'PRODUCTCODE'	TRUE
Enable Flag(4) 'MAJORMINORREV'	TRUE
Enable Flag(5) 'SERIALNR'	TRUE
Enable Flag(6) 'PRODUCTNAME'	TRUE
Vendor ID	283
Product Type	12
Product Code	28
Minor Revision	1
Major Revision	1
Serial Number	20086
DNS State	OFFLINE
Status Flags (Summary)	0x00000003
Status Flag(1) 'BUS_PRM_VALID'	TRUE
Status Flags(2) 'BUS_START'	TRUE
Status Flags(3) '24V_NETWORK_POWER'	FALSE
Status Flags(4) 'NETWORK_STATE_ONLINE'	FALSE
RX Interrupts	0
TX Interrupts	0
RX Overrun	0
TX Overrun	0
TX Aborts	0
Error Interrupt	0
Bus Off Count	0
Reset Count	1

Figure 104: Extended Diagnosis > DNS_FAL > DNS Common Status

Name	Description
[*Status*]	Common DNS Diagnosis status. Indicates the current status of the single tasks. (For further information refer to the DeviceNet Slave Protocol API manual [11].)

Table 104: Extended Diagnosis > DNS_FAL > DNS Common Status

6.15.5 DNS_FAL/AP Commands Counter

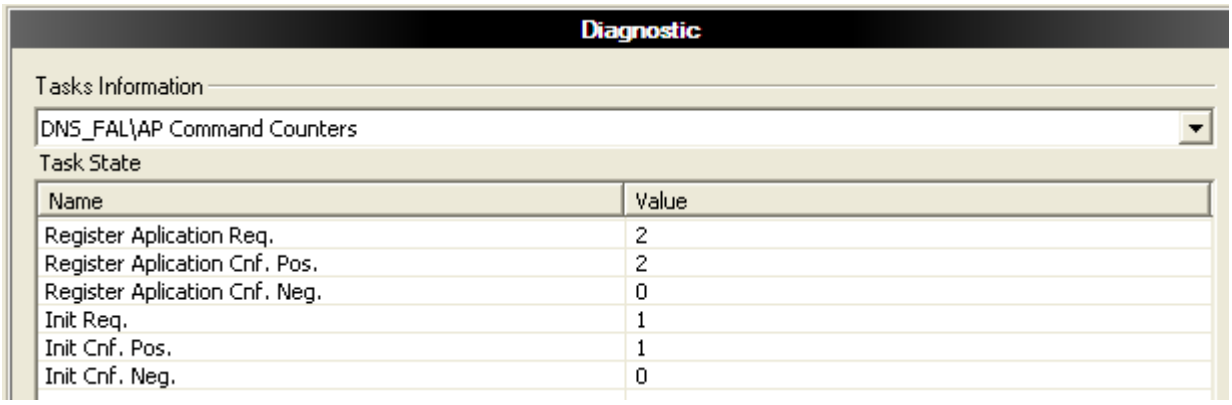


Figure 105: Extended Diagnosis > DNS_FAL > AP Commands Counter

Name	Description
[*Service*]	Diagnosis counter of the FAL layer. Indicates the services processed. (The services of the single packets are described in the DeviceNet Slave Protocol API manual [11].)

Table 105: Extended Diagnosis > DNS_FAL > AP Commands Counter

6.15.6 DNS_FAL/CAN Command Counter

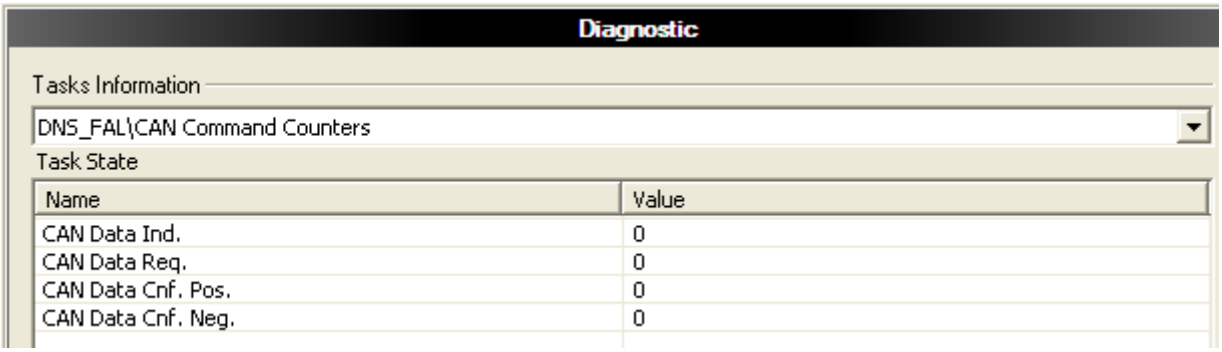


Figure 106: Extended Diagnosis > DNS_FAL > CAN Command Counter

Name	Description
[*Service*]	Diagnosis counter of the FAL layer. Indicates the services processed. (The services of the single packets are described in the DeviceNet Slave Protocol API manual [11].)

Table 106: Extended Diagnosis > DNS_FAL > CAN Command Counter

6.15.7 DNS_FAL/Timer Counter

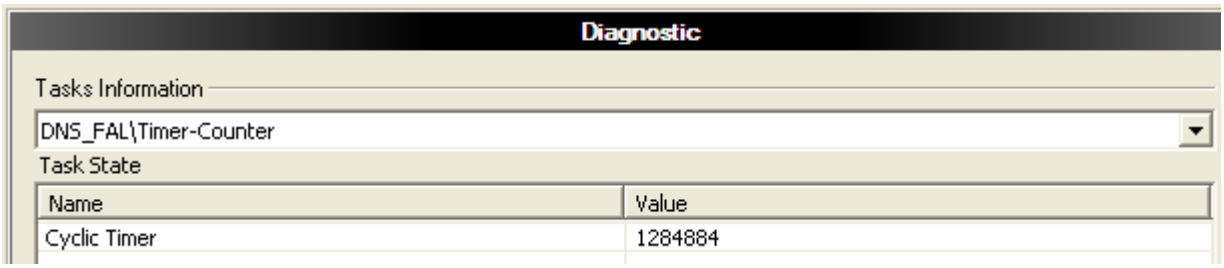


Figure 107: Extended Diagnosis > DNS_FAL > Timer Counter

Name	Description
[*Service*]	Diagnosis counter of the FAL layer. Indicates the services processed. (The services of the single packets are described in the DeviceNet Slave Protocol API manual [11].)

Table 107: Extended Diagnosis > DNS_FAL > Timer Counter

6.16 Descriptions for Tasks with similar Functions

Task Information Group	Taks Information	Manual Page
Task Information	*Task Information*	166
IniBatch Status	*IniBatch Status*	167
General Diagnosis Information	*General Diagnosis Information*	168
Code Diagnosis	*Code Diagnosis*	169
TCPUDP	*TCPUDP/IP Information*	170
	TCPUDP/IP Ethernet Counter	171
	TCPUDP/IP Packet Counter	172
	TCPUDP/IP Code Diagnosis	173
	TCPUDP/TCP_UDP Information	174
	TCPUDP/TCP_UDP Code Diagnosis	175

Table 108: Descriptions of the Dialog Panes Extended Diagnosis for Tasks with similar Functions

6.16.1 Task Information

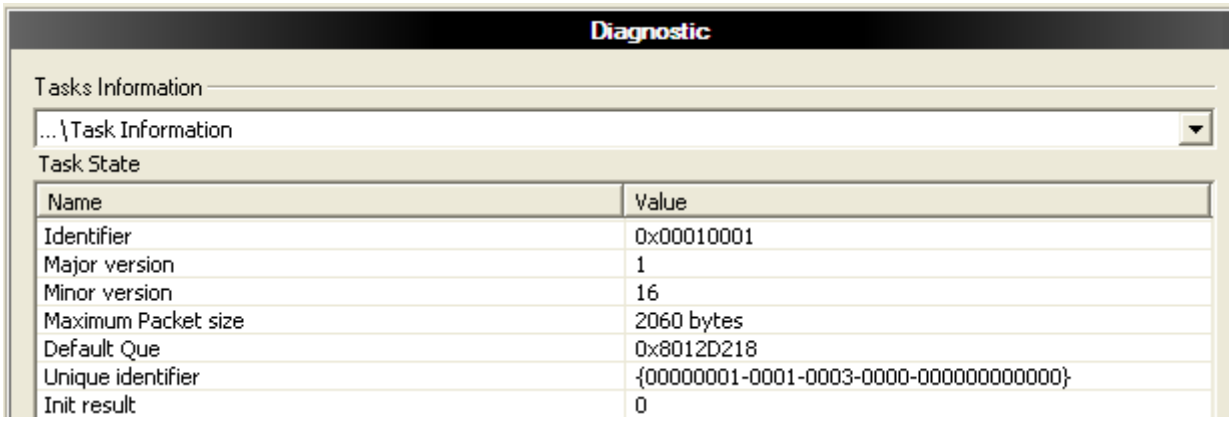


Figure 108: Extended Diagnosis > [Folder Name] > Task Information, Example

Name	Description
Identifier	Identification number of the task
Major version	Task version, contains incompatible changes
Minor version	Task version, contains compatible changes
Maximum package size	Maximum package 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 109: Extended Diagnosis > [Folder Name] > Task Information

6.16.2 IniBatch Status

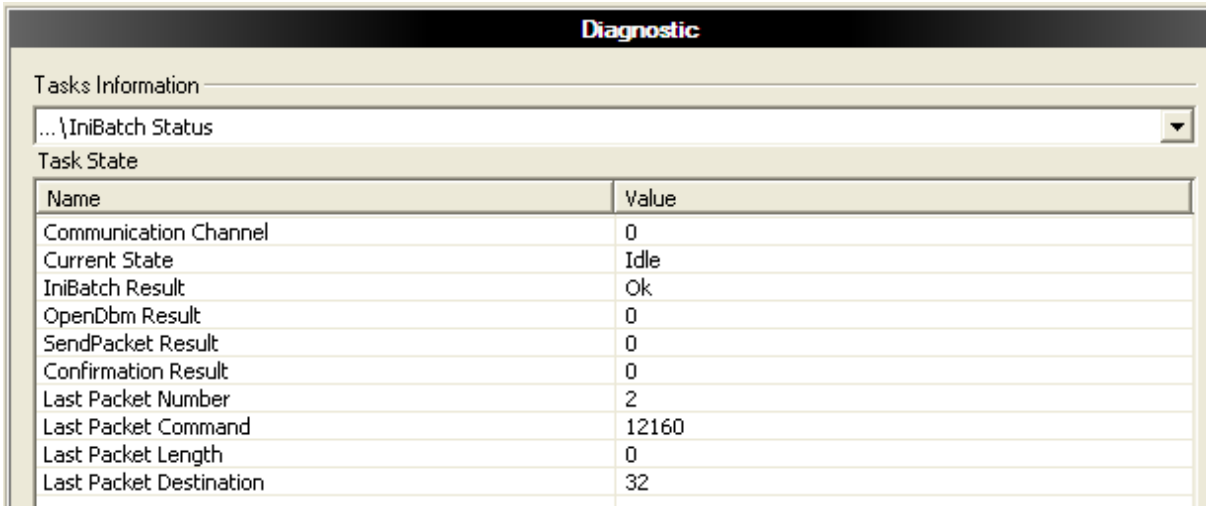


Figure 109: Extended Diagnosis > [Folder Name] > IniBatch Status, Example

Name	Description
Communication Channel	Number of the communication channel used by the device.
Current State	0 =Idle 1 =IniBatch packets in progress 2 =Retrying to send last packet 3 = Error
IniBatch Result	0 = Ok; 1 = No DBM file 2 = No Packet table 3 = No data set available 4 = Data set is shorter than packet length 5 = Packet Buffer is shorter than Packet length 6 = Invalid packet destination 7 = Logical queue not defined 8 = Send packet failed 9 = Too many retries 10 = 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 package 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 110: Extended Diagnosis > [Folder Name] > IniBatch Status, Example Display

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

6.16.3 General Diagnosis Information

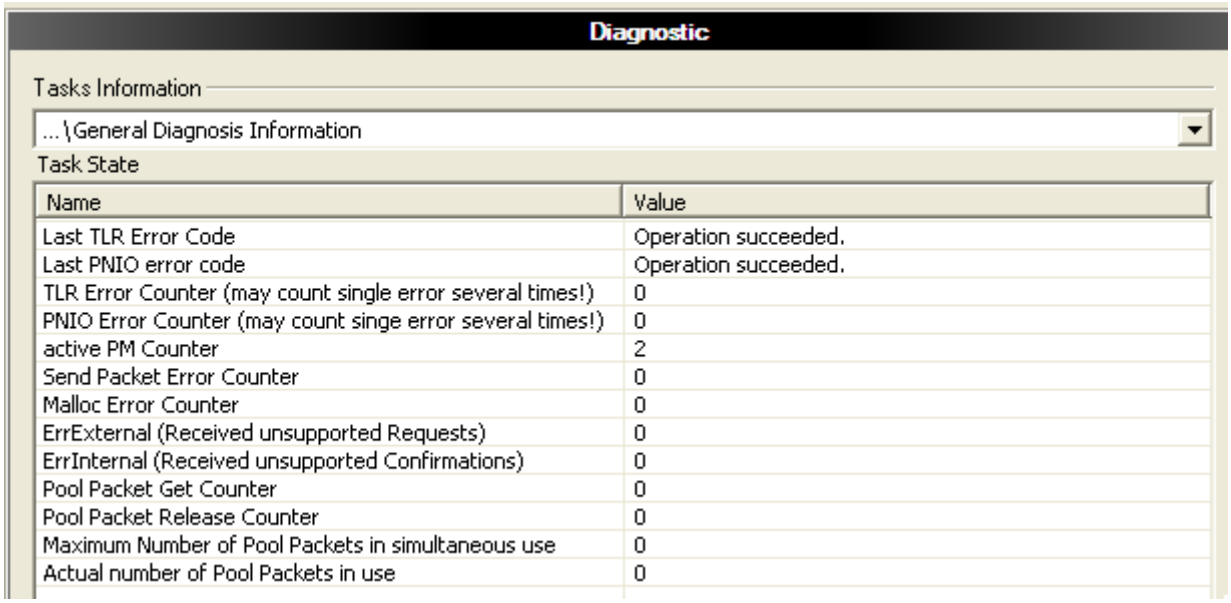


Figure 110: Extended Diagnosis > [Folder Name] > General Diagnosis Information, Example

Name	Description
Last TLR error code	Error code of the last internally occurred error
Last PNIO error code	Error code of the last externally occurred error which has been reported by an IO Device.
TLR Error Counter (may count single error several times!)	Counts the number of occurred TLR error codes.
PNIO Error Counter (may count single error several times!)	Counts the number of occurred PNIO error codes.
Active PM Counter	Counter of the active protocol machines in the task.
Send Packet Error Counter	Counts how often a task sends a packet to another task and this fails.
Malloc Error Counter	Counts how often storage capacity is requested in the operating system and the operating system cannot follow this demand.
ErrExternal (Received unsupported Requests)	Counts how often an unknown request packet was received.
ErrInternal (Received unsupported Confirmations)	Counts how often an unknown confirmation packet was received.
Pool Packet Get Counter*	Counts how many packets are retrieved from the pool.
Pool Packet Release Counter*	Counts how many packets are given back to the pool.
Maximum Number of Pool Packets in simultaneous use*	Counts how many pool elements were used simultaneously as a maximum.
Actual number of Pool Packets in use*	Counts how many pool elements are used currently.
*Not used for the diagnosis task „T_RPC“.	

Table 111: Extended Diagnosis > [Folder Name] > General Diagnosis Information

6.16.4 Code Diagnosis

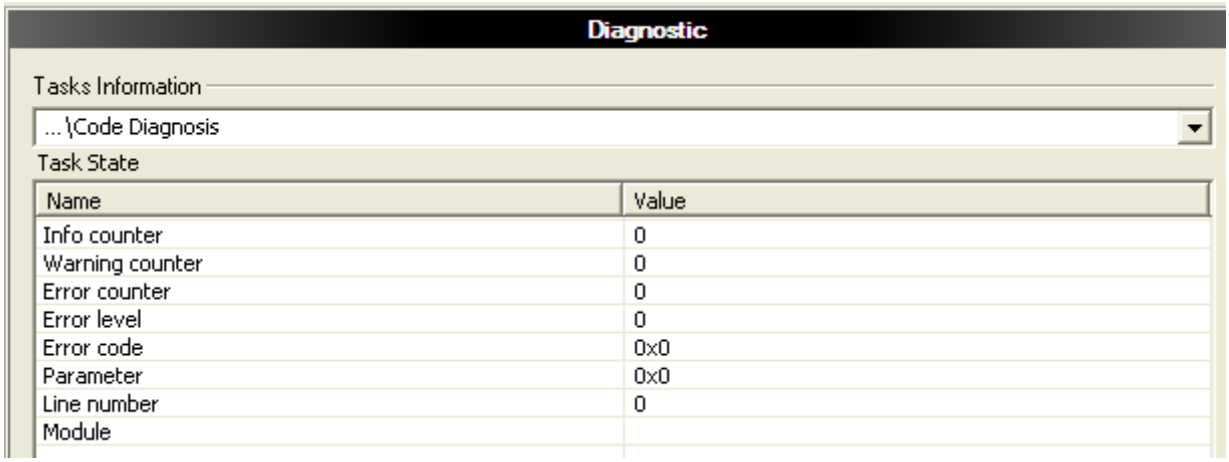


Figure 111: Extended Diagnosis > [Folder Name] > Code Diagnosis, Example

Name	Description
Info Counter	Counter for information reports
Warning Counter	Counter for warning reports
Error Counter	Counter for errors
Error Level	Level of the last occurred error
Error Code	Code of the last occurred error
Parameter	Additional information to the error
Line number	Line number inside the software module
Module	Software module

Table 112: Extended Diagnosis > [Folder Name] > Code Diagnosis

6.16.5 TCPUDP/IP Information

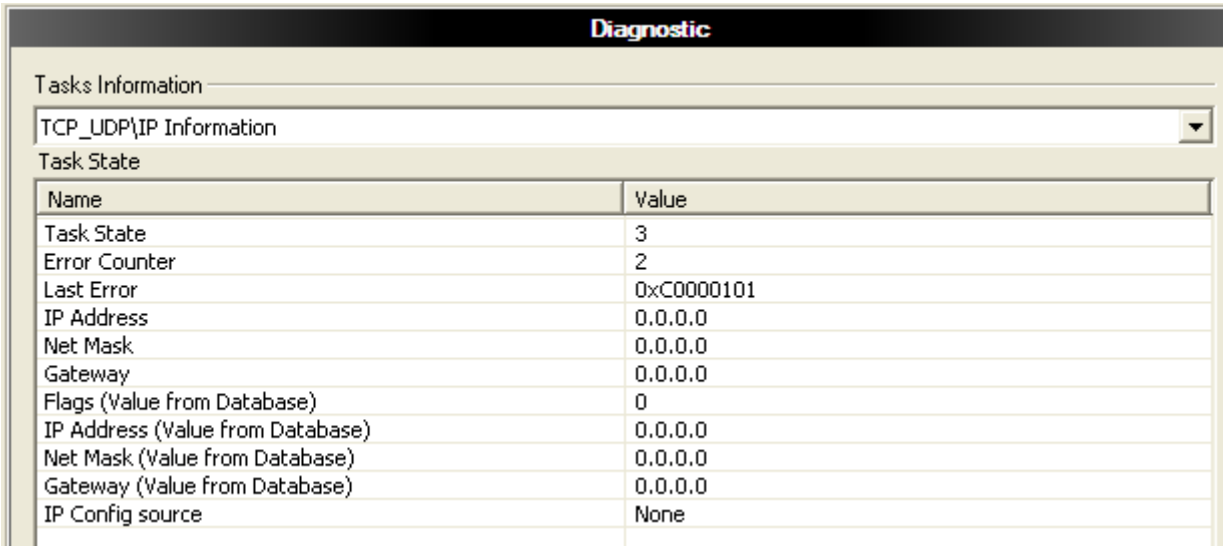


Figure 112: Extended Diagnosis > [Folder Name] > IP Information, Example

Name	Description
Task State	Actual state of the protocol process: 0 = Task not initialized 1 = Task is running 2 = Task initialized 3 = Initialization error
Error Counter	Counter for errors
Last Error	Last occurred error (Description see appropriate manual)
IP Address	IP address of the Slave station
Net Mask	Network mask of the Slave station
Gateway	Gateway address of the Slave station

Table 113: Extended Diagnosis > [Folder Name] > IP Information

6.16.6 TCPUDP/IP Ethernet Counter

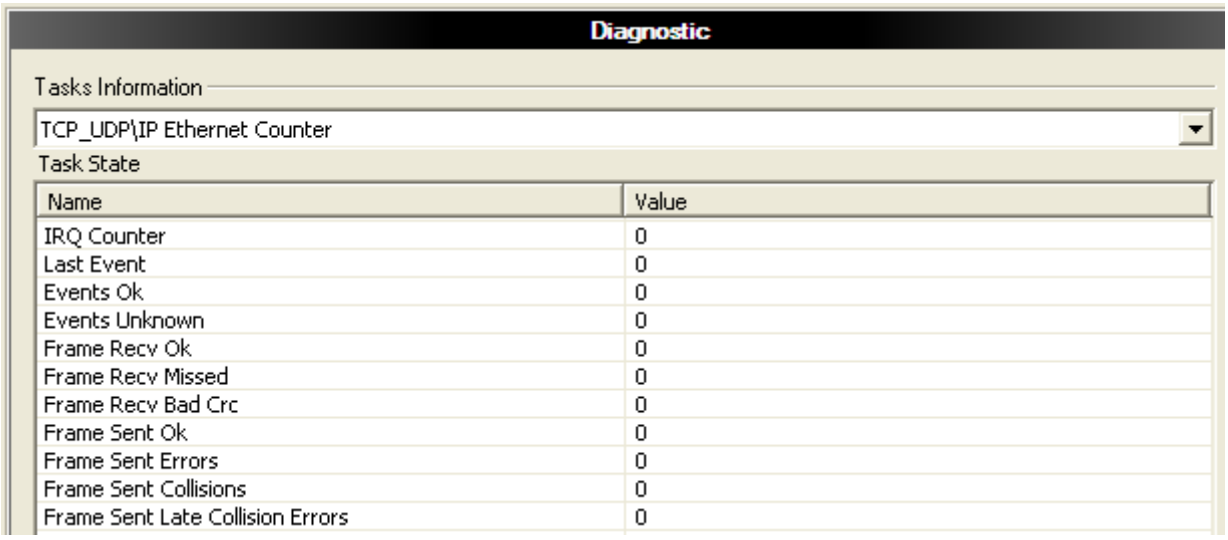


Figure 113: Extended Diagnosis > [Folder Name] > IP Ethernet Counter, Example

Name	Description
IRQ Count	Counter for interrupts of the EtherNet/IP adapter station
Last Event	Last occurred interrupt type
Events Ok	Counter for known interrupt types
Events unknown	Counter for unknown interrupt types
Frame Recv Ok	Counter for received Ethernet frames
Frame Recv Missed	Counter for missed Ethernet frames
Frame Recv bad Crc	Counter for Ethernet frames with CRC errors
Frame Sent ok	Counter for sent Ethernet frames
Frame sent Errors	Counter for sending errors
Frame sent Collisions	Counter for sending collisions
Frame sent late Collision Errors	Counter for late sending collisions

Table 114: Extended Diagnosis > [Folder Name] > IP Ethernet Counter

6.16.7 TCPUDP/IP Packet Counter

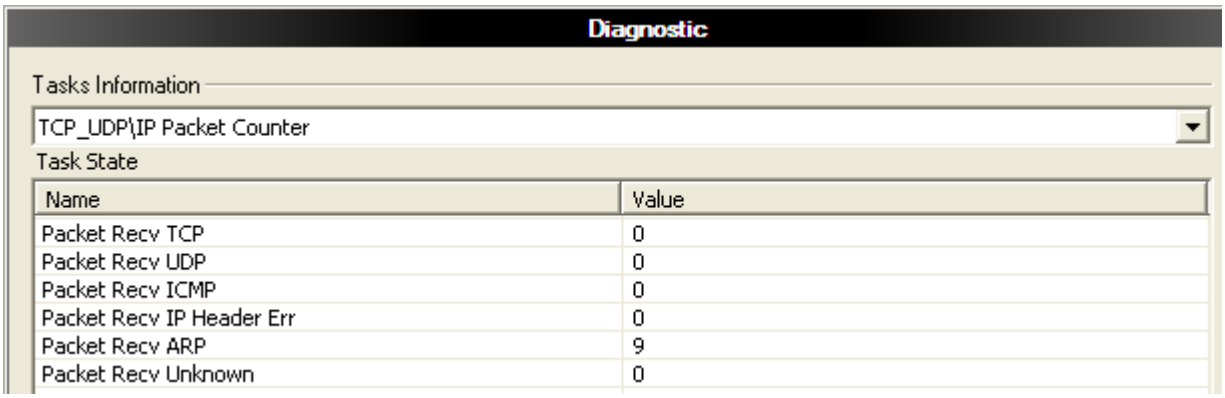


Figure 114: Extended Diagnosis > [Folder Name] > IP Packet Counter, Example

Name	Description
Packet Recv TCP	Counter for received TCP packets
Packet Recv UDP	Counter for received UDP packets
Packet Recv ICMP	Counter for received ICMP packets
Packet Recv IP Header Err	Counter for received IP packets with errors
Packet Recv ARP	Counter for received ARP packets
Packet Recv Unknown	Counter for received packets of an unknown type

Table 115: Extended Diagnosis > [Folder Name] > IP Packet Counter

6.16.8 TCPUDP/IP Code Diagnosis

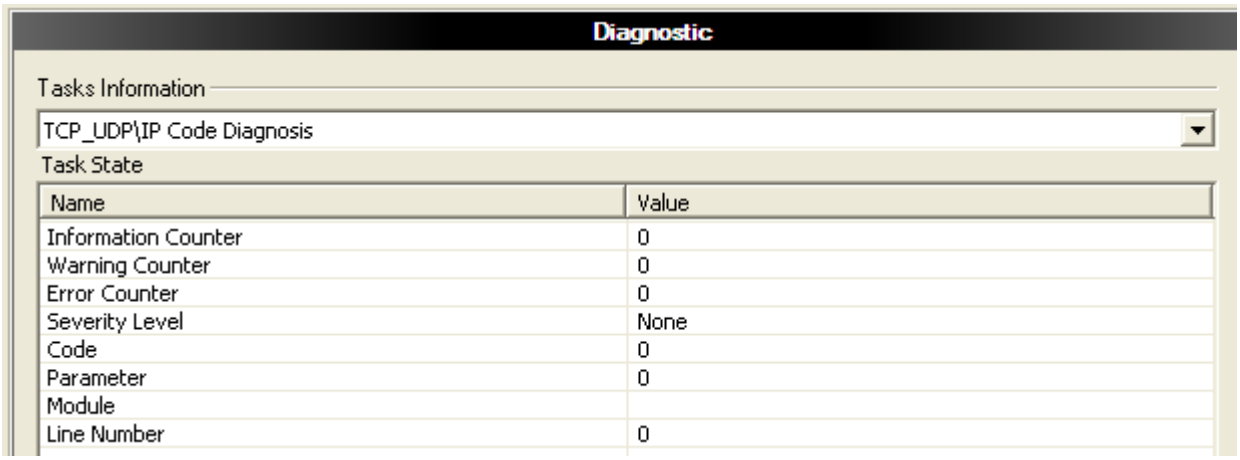


Figure 115: Extended Diagnosis > [Folder Name] > IP-Code Diagnosis, Example

Name	Description
Info Counter	Counter for information reports
Warning Counter	Counter for warning reports
Error Counter	Counter for errors
Severity Level	Level of the last occurred error
Code	Code of the last occurred error
Parameter	Additional information to the error
Module	Software module
Line number	Line number inside the software module

Table 116: Extended Diagnosis > [Folder Name] > IP-Code Diagnosis

6.16.9 TCPUDP/TCP_UDP Information

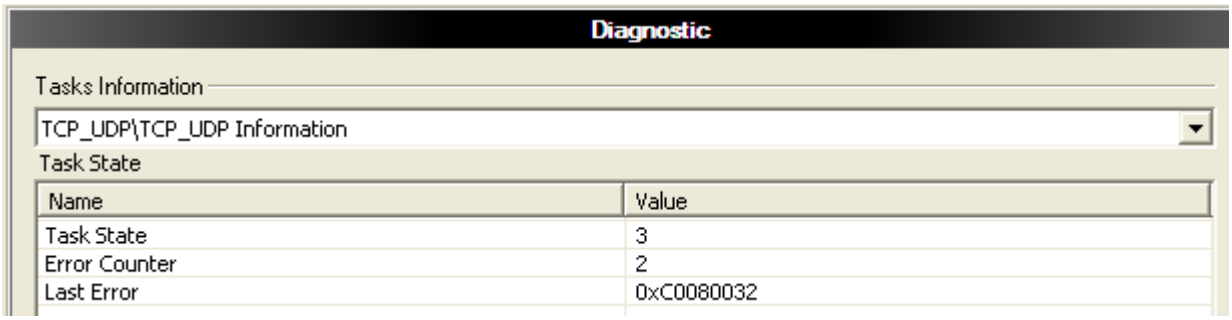


Figure 116: Extended Diagnosis > [Folder Name] > TCP_UDP-Information, Example

Name	Description
Task State	Actual state of the protocol process: 0 = Task not initialized 1 = Task is running 2 = Task initialized 3 = Initialization error
Error Count	Counter for errors
Last Error	Last occurred error (Description see appropriate manual)

Table 117: Extended Diagnosis > [Folder Name] > TCP_UDP-Information

6.16.10 TCPUDP/TCP_UDP Code Diagnosis

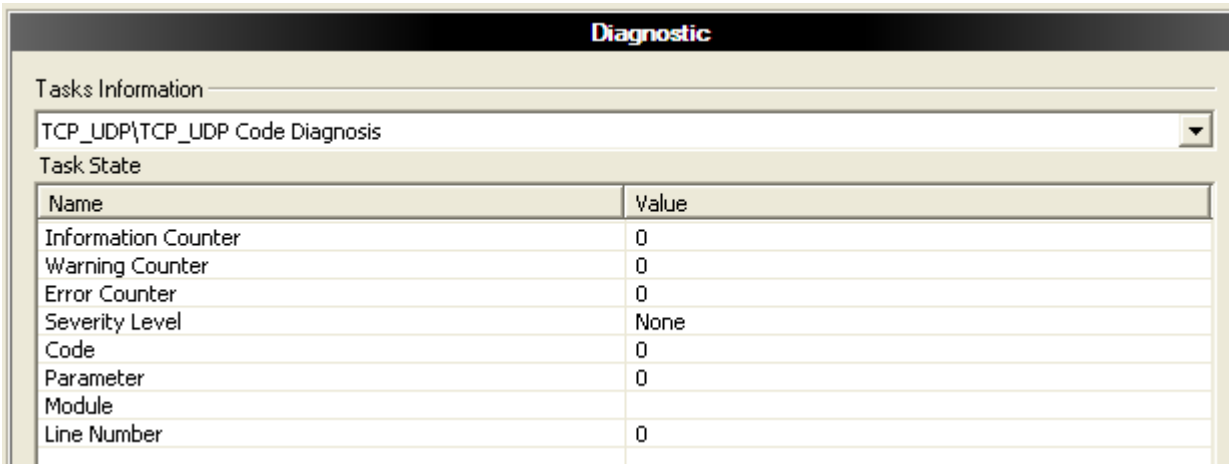


Figure 117: Extended Diagnosis > [Folder Name] > TCP_UDP Code Diagnosis Example, Display

Name	Description
Info Counter	Counter for information reports
Warning Counter	Counter for warning reports
Error Counter	Counter for errors
Severity Level	Level of the last occurred error
Code	Code of the last occurred error
Parameter	Additional information to the error
Module	Software module
Line number	Line number inside the software module

Table 118: Extended Diagnosis > [Folder Name] > TCP_UDP Code Diagnosis

7 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.

➢ Open the IO Monitor via navigation area > **IO Monitor**.

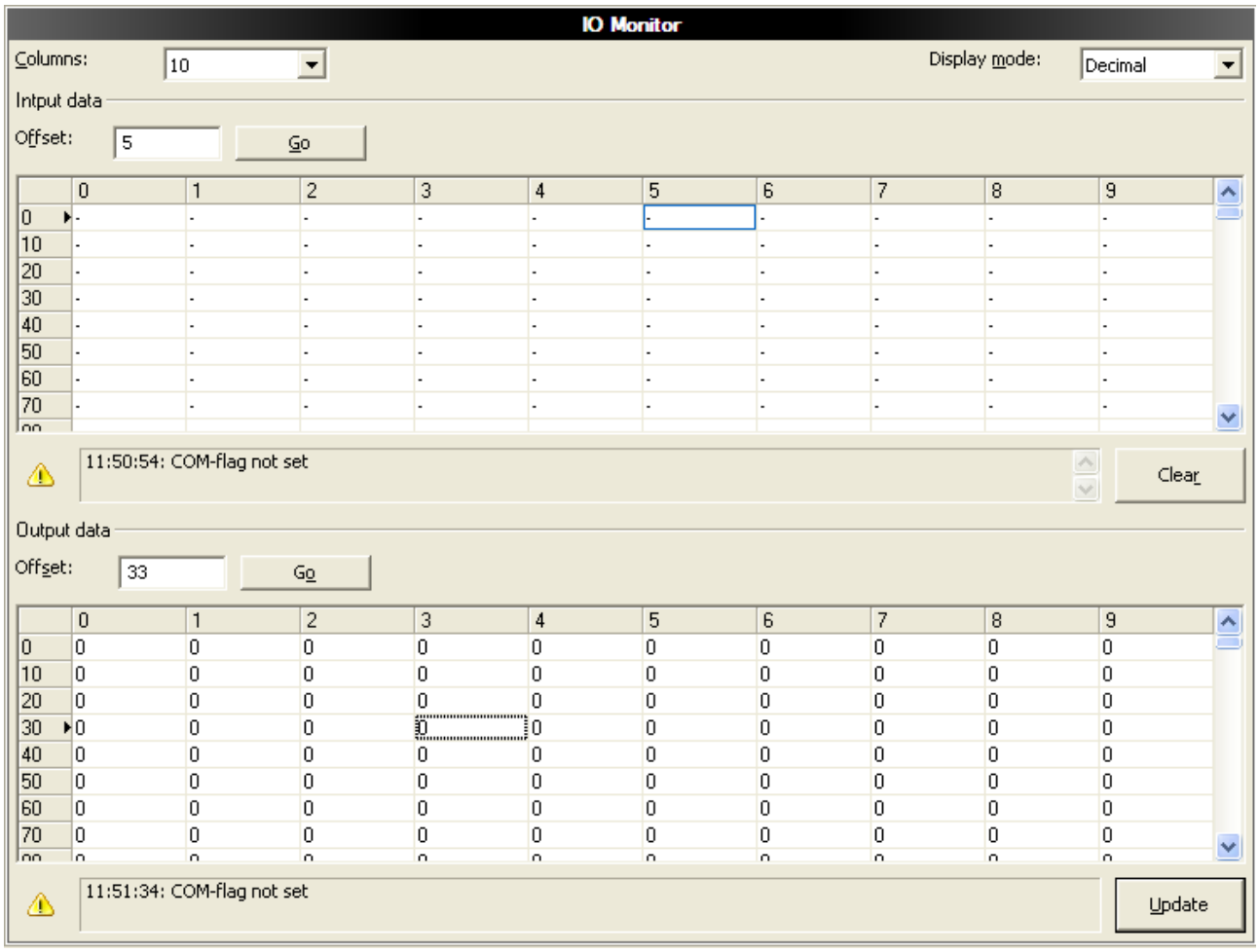


Figure 118: IO Monitor

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.

Clear clears displayed error messages.

➢ Enter the output value and select the button **Update**.

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

8 Annex

8.1 EtherCAT Summary over Vendor ID, Conformance test, Membership and Network Logo

8.1.1 Vendor ID

The communication interface product is shipped with Hilscher's secondary vendor ID, which has to be replaced by the Vendor ID of the company shipping end products with the integrated communication interface. End Users or Integrators may use the communication interface product without further modification if they re-distribute the interface product (e.g. PCI Interface card products) only as part of a machine or machine line or as spare part for such a machine. In case of questions, contact Hilscher and/or your nearest ETG representative. The ETG Vendor-ID policies apply.

8.1.2 Conformance

EtherCAT Devices have to conform to the EtherCAT specifications. The EtherCAT Conformance Test Policies apply, which can be obtained from the EtherCAT Technology Group (ETG, www.ethercat.org).

Hilscher range of embedded network interface products are conformance tested for network compliance. This simplifies conformance testing of the end product and can be used as a reference for the end product as a statement of network conformance (when used with standard operational settings). It must however be clearly stated in the product documentation that this applies to the network interface and not to the complete product.

Conformance Certificates can be obtained by passing the conformance test in an official EtherCAT Conformance Test lab. Conformance Certificates are not mandatory, but may be required by the end user.

8.1.3 Certified Product vs. Certified Network Interface

The EtherCAT implementation may in certain cases allow one to modify the behavior of the EtherCAT network interface device in ways which are not in line with EtherCAT conformance requirements. For example, certain communication parameters are set by a software stack, in which case the actual software implementation in the device application determines whether or not the network interface can pass the EtherCAT conformance test. In such cases, conformance test of the end product must be passed to ensure that the implementation does not affect network compliance.

Generally, implementations of this kind require in-depth knowledge in the operating fundamentals of EtherCAT. To find out whether or not a certain type of implementation can pass conformance testing and requires such testing, contact EtherCAT Technology Group (“ETG”, www.ethercat.org) and/or your nearest EtherCAT conformance test centre. EtherCAT may allow the combination of an untested end product with a conformant network interface. Although this may in some cases make it possible to sell the end product without having to perform network conformance tests, this approach is generally not endorsed by Hilscher. In case of questions, contact Hilscher and/or your nearest ETG representative.

8.1.4 Membership and Network Logo

Generally, membership in the network organization and a valid Vendor-ID are prerequisites in order to be able to test the end product for conformance. This also applies to the use of the EtherCAT name and logo, which is covered by the ETG marking rules.

Vendor ID Policy accepted by ETG Board of Directors, November 5, 2008

8.2 References

[1] Device Type Manager (DTM) Style Guide, Version 1.0 ; FDT-JIG - Order No. <0001-0008-000>

[2] EtherNetIP Adapter Protocol API Manual, Revision 12, Hilscher GmbH 2013

[3] Open Modbus/TCP Protocol API Manual, Revision 7, Hilscher GmbH 2013

[4] PROFINET IO-Device Protocol API Manual (V3.4), Revision 13, Hilscher GmbH 2013

PROFINET IO-Device Protocol API Manual (V3.5), Revision 3, Hilscher GmbH 2013

[5] sercos Slave Protocol API Manual (V3), Revision 11, Hilscher GmbH 2013

[6] PROFIBUS DP Slave Protocol API Manual, Revision 14, Hilscher GmbH 2013

[7] CC-Link Slave Protocol API Manual, Revision 8, Hilscher GmbH 2013

[8] CANopen Slave Protocol API Manual (V3), Revision 4, Hilscher GmbH 2013

[9] EtherCAT Slave Protocol API Manual, Revision 3 (V4), Hilscher GmbH 2013

EtherCAT Slave Protocol API Manual, Revision 20 (V2), Hilscher GmbH 2013

[10] POWERLINK Controlled Node/Slave Protocol API Manual, Revision 11, Hilscher GmbH 2013

[11] DeviceNet Slave Protocol API Manual, Revision 12, Hilscher GmbH 2013

[12] CompoNet Slave Protocol API Manual, Revision 2, Hilscher GmbH 2010

[13] 3964R Slave Protocol API Manual, Revision 1, Hilscher GmbH 2010

[14] Modbus RTU Protocol API Manual, Revision 1, Hilscher GmbH 2010

[15] VARAN Client Protocol API Manual, Revision 3, Hilscher GmbH 2013

[16] PROFIBUS MPI Protocol API Manual, Revision 4, Hilscher GmbH 2011

[17] Dual-Port Memory Interface Manual, Revision 12, Hilscher GmbH 2012

8.3 List of Figures

8.4 List of Tables

8.5 Glossary

Auto-Negotiation

Auto-Negotiation is a feature of an interface: An interface with Auto-Negotiation will automatically determine a set of correct communication parameters.

Baud rate

Data transmission speed of a communication channel or interface.

cifX

Communication **I**nter**F**ace based on net**X**

Coil

A coil is a single bit in the memory that can be accessed using Modbus: read or write access with FC 1, 5, 15. Depending on the used Modbus function code a single coil or several coils lying in succession can be accessed.

comX

Real-time Ethernet and fieldbus modules

Communication modules of the comX family of Hilscher on the basis of the communication controller netX 10, 50 and 100.

CRC

Cyclic Redundancy Check

A mathematic procedure for calculating checksums based on polynomial division in order to detect data transmission errors. For a more detailed description see the Wikipedia article

(http://en.wikipedia.org/wiki/Cyclic_redundancy_check).

CSP

electronic device data sheet, required for each CC-Link device

Device Description File

A file containing configuration information about a device being a part of a network that can be read out by masters for system configuration. Device Description Files use various formats which depend on the communication system. Often these formats are based on XML such as EDS_files or GSDML_files. Contains configuration information

DHCP

Dynamic Host Configuration Protocol

This is a protocol simplifying the configuration of IP networks by automatically assigning IP addresses.

Discrete Input

A “Discrete Input” (as defined in the Modbus terminology) is a single bit in the memory which can be accessed using Modbus (read with FC 2).

DP

Decentral Periphery

DPM

Dual-Port Memory

EDS

Electronic Data Sheet

EDS file

A special kind of Device Description File used for example by EtherNet/IP.

EtherCAT

A communication system for industrial Ethernet designed and developed by Beckhoff Automation GmbH.

Ethernet

A networking technology used both for office and industrial communication via electrical or optical connections. It has been developed and specified by the Intel, DEC and XEROX. It provides data transmission with collision control and allows various protocols. As Ethernet is not necessarily capable for real-time application, various real-time extensions have been developed, see Real-Time Ethernet.

EtherNet/IP

A communication system for industrial Ethernet designed and developed by Rockwell. It partly uses the CIP (Common Industrial Protocol).

EtherNet/IP

A communication system for industrial Ethernet designed and developed by Rockwell. It partly uses the CIP (Common Industrial Protocol).

EtherNet/IP Scanner

A Scanner exchanges real-time I/O data with Adapters and Scanners. This type of node can respond to connection requests and can also initiate connections on its own.

Ethernet POWERLINK

A communication system for industrial Ethernet designed and developed by B&R. It partly uses CANopen technologies.

FDL

Fieldbus Data Link defines the PROFIBUS communication on layer 2, identical for DP and FMS

Full duplex

Full duplex denominates a telecommunication system between two communication partners which allows simultaneous communication in both directions is called a full-duplex telecommunication system. At such a system, it is possible to transmit data even if currently data are received. Full-duplex is the opposite of Half duplex.

Function code

A function code (FC) is a standardized method to access, i. e. read or write on coils (Bits) or registers via Modbus.

Modbus function codes are elements of Modbus request/reply telegrams.

Gateway

A device interfacing between two different communication standards.

GSD

Generic Station Description, Device description file

GSD file

A special kind of Device Description File used by PROFIBUS (GSD = Generic Station Description).

GSDML

Generic Station Description Markup Language

XML based device description file.

GSDML file

A special kind of XML-based Device Description File used by PROFINET.

Half duplex

Half duplex denominates a telecommunication system between two communication partners which does not allow simultaneous, but alternating, communication in both directions is called a half-duplex telecommunication system. At such a system, receiving data inhibits the transmission of data. Half-duplex is the opposite of _Full_duplex.

Hub

A network component connecting multiple communication partners with each other. A hub does not provide own intelligence, thus it does not analyze the data traffic and sends received data to all connected communication partners. A hub can be used for setting up a star topology.

Industrial Ethernet

See Real-Time Ethernet

Modbus Data Model

The data model distinguishes four basic types of data areas:

- Discrete Inputs (inputs) = FC 2 (Read)
- Coils (outputs) = FC 1, 5, 15 (Write and Read back)
- Input register (input data) = FC 4 (Read)
- Holding register (output data) = FC 3, 6, 16, 23 (Write and Read back).

It should be noted, however, that depending on the device manufacturer and device type:

- the data area in the device may be present or not,
- and two data areas can be combined into one data region. For example, discrete inputs and input registers can be a common data area, which can be accessed with read-FC 2 and FC 4.
- Further FC 1 and FC 3 are used instead of reading back the inputs to read the outputs.

MPI

Multi Point Interface

MPI ist eine proprietäre Schnittstelle der SIMATIC® S7® Serie von speicherprogrammierbaren Steuerungen. MPI ist PROFIBUS-kompatibel, basiert auf RS-485 und arbeitet gewöhnlich mit einer Datenübertragungs-rate von 187,5 kBaud.

netJACK

Real-time Ethernet and fieldbus modules

Communication modules of the netJACK family of Hilscher on the basis of the communication controller netX 10, 50 and 100 with either PCI Express or 8/16 bit dual port memory interface.

netX

networX on chip, next generation of communication controllers

netX Configuration Tool

The netX Configuration Tool allows users to operate netX based devices in different networks. Its graphical user interface serves as a configuration tool for the installation, configuration and Diagnostic of the devices.

Object Dictionary

An object dictionary is a storage area for device parameter data structures. It is accessed in standardized manner.

Open Modbus/TCP

A communication system for Industrial Ethernet designed and developed by Schneider Automation and maintained by the Modbus-IDA organization based on the Modbus protocols for serial communication.

PC Cards cifX

Communication Interfaces of the cifX product family of Hilscher on the basis of the communication controller netX 100:

for the Real-Time Ethernet systems

- EtherCAT
- EtherNet/IP
- Open-Modbus/TCP
- POWERLINK
- PROFINET IO
- sercos
- VARAN

and for the fieldbus systems

- PROFIBUS DP
- PROFIBUS MPI
- CANopen
- DeviceNet
- AS-Interface
- CompoNet
- CC-Link

as Communication Interface netX with PCI Bus

- PCI (CIFX50),
- PCI Express (CIFX 50E),
- Low Profile PCI Express (CIFX 70E, CIFX 100EH-RE\CUBE*),
- Compact PCI (CIFX80),
- Mini PCI (CIFX90),
- Mini PCI Express (CIFX 90E),
- PCI-104 (CIFX 104C)

and as Communication Interface netX with ISA Bus

- PC/104 (CIFX 104).

*only Real-Time Ethernet

PROFINET

A communication system for Industrial Ethernet designed and developed by PROFIBUS International. It uses some mechanisms similar to those of the PROFIBUS field bus.

PROFINET IO Controller

A PROFINET control unit responsible for the defined run-up of an I/O subsystem and the cyclic or acyclic data exchange.

PROFINET IO Device

A PROFINET field device that cyclically receives output data from its IO-Controller and responds with its input data.

RE

RE stands for Real Time Ethernet

Real-Time Ethernet

Real-Time Ethernet (Industrial Ethernet) is an extension of the Ethernet networking technology for industrial purposes with very good Real-Time features and performance. There is a variety of different Real-Time Ethernet systems on the market which are incompatible with each other. The most important systems of these are

- EtherCAT
- EtherNet/IP
- Ethernet POWERLINK
- Open Modbus/TCP
- PROFINET
- Sercos
- VARAN

Register

A register is a 16-bit wide storage area for data which can be accessed and addressed as a unit by some of the Modbus Function Codes.

Depending on the used Modbus function code a single register or multiple registers sequentially located can be accessed.

Modbus differs Input Registers (FC 4) and Holding Registers (FC 3, 6, 16, 23).

RS232

An interfacing standard for serial communication on data lines defined by EIA (Electronic Industries Alliance) in *ANSI/EIA/TIA-232-F-1997.*

SCP

sercos Communication Profile

sercos

A communication system for industrial Ethernet designed and developed by Bosch-Rexroth and supported by sercos International.

SPI

SPI means Serial Peripheral Interface. SPI is a bus system for a synchronous serial data bus which has been developed by Motorola. SPI makes use of the master Slave-principle. It requires at least three lines used for data input, data output and clock and works in full duplex mode.

Switch

A network component connecting multiple communication partners (or even entire branches of a network) with each other. A switch is an intelligent network component which analyzes network traffic in order to decide on its own. For the connected communication partners a switch behaves transparently.

SYNC

Sychronisation cycle of the master

TCP/IP

Transport Control Protocol/Internet Protocol connection-orientated, secure transfer protocol as basis for the Internet-protocols

UART

UART means Universal Asynchronous Receiver Transmitter. It is a special kind of electronic circuit which is used for transmitting data serially with a fixed frame consisting of one start bit, five to nine data bits, an optional parity bit for the detection of transmission errors and one stop bit. Working asynchronously, it does not use an explicit clock signal.

UCMM

Unconnected Message Manager

VARAN

Versatile **A**utomation **R**andom **A**ccess **N**etwork

A communication system for industrial Ethernet based on the DIAS-BUS developed by Sigmatek. The system is supported by the VARAN-BUS-NUTZERORGANISATION (VNO).

Warmstart

A part of the initialization process of netX-controlled communication systems. During warmstart the netX-controlled system is adjusted to the intended parameters of operation. These parameters are supplied by a special message, the warmstart message which is transferred to the netX within the warmstart packet.

Watchdog Timer

A watchdog timer provides an internal supervision mechanism of a communication system. It supervises that an important event happens within a given timeframe (the watchdog time which can be adjusted accordingly, for instance by a parameter in the Warmstart message) and causes an alarm otherwise (usually this is accomplished by changing the operational state of the communication system to a more safe state).

XDD file

A special kind of Device Description file used by Ethernet POWERLINK.

XML

XML means Extended Markup Language. It is a symbolic language for structuring data systematically. XML is standard maintained by the W3C (World-wide web consortium). Device Description Files often use XML-based formats for storing the device-related data appropriately.

8.6 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.
New Delhi - 110 065
Phone: +91 11 26915430
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