Application Note

Protocol Parameter via Modbus

Configuration netIC

Hilscher Gesellschaft für Systemautomation mbH

www.hilscher.com

DOC110503AN02EN | Revision 2 | English | 2012-01 | Released | Public

Table of Contents

1 Introduction

1.1 About this Document

After installation of the netIC, it needs to be configured in order to be able to use it. There are both protocol-independent and protocol-dependent settings to be made.

In general, there are two different ways to accomplish this:

- Via Modbus RTU or
- Via netX Configuration Tool, which is the easiest way.

The latter way of configuration via the X Configuration Tool is described in a separate document (see *Operating Instruction Manual, netX Configuration Tool for netIC*) while this document concentrates on describing in detail the configuration via Modbus RTU.

1.2 List of Revisions

Rev	Date	Name	Chapter	Revision
1	2011-11-02	RG	All	Created
2	2012-01-18	HH	-	Componet removed

Table 1: List of Revisions

1.3 Terms, Abbreviations and Definitions

Term	Description
BootP	Boot Protocol (Internet)
CRC	Cyclic Redundancy Check
DAP	Device Access Point (PROFINET IO)
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name Service
EtherCAT	Ethernet Control Automation Technology
GSD	Generic Station Description
IP	Internet Protocol (Internet)
LSB	Least significant bit
MSB	Most significant bit
PDO	Process Data Object (CANopen, EtherCAT and Powerlink)
SHIF	Serial Host Interface
SII	Slave Information Interface (EtherCAT)
SSIO	Synchronous Serial Input Output
TCP	Transmission Control Protocol (Internet)
VARAN	Versatile Automation Random Access Network
XML	Extensible markup language

Table 2: Terms, Abbreviations and Definitions

1.4 References

The following list provides an overview on available documentation.

Manual	Contents	File Name of the Document
Protocol-independent documents		
User Manual and Design Guide, NIC 50 Real Time Ethernet and Fieldbus Gateways	Installation, Commissioning, Operation and Hardware Description	netIC_usermanual_designguide_en.pdf (English version) netIC_Benutzerhandbuch_Designguide_de.doc (German version)
Protocol-dependent documents		
EtherCAT Slave	Description of EtherCAT Slave Protocol API	EtherCAT Slave Protocol API XX EN.pdf (English Version)
EtherNet/IP Adapter	Description of EtherNet/IP Adapter Protocol API	EtherNetIP Adapter Protocol API XX EN.pdf (English Version)
Open Modbus/TCP	Description of Open Modbus/TCP Protocol API	OpenModbusTCP Protocol API XX EN.pdf (English Version)
Powerlink Controlled Node	Description of Powerlink Controlled Node Protocol API	Powerlink Controlled Node Protocol API XX EN.pdf (English Version)
PROFINET IO RT IRT Device	Description of PROFINET IO RT IRT Device Protocol API (V3)	PROFINET IO RT IRT Device Protocol API XX EN.pdf (English Version)
VARAN Client	Description of VARAN Protocol API	VARAN Client (Slave) Protocol API XX EN.pdf (English Version)
CANopen Slave	Description of CANopen Slave Protocol API	CANopen Slave Protocol API XX EN.pdf (English Version)
CC-Link Slave	Description of CC-Link Slave Protocol API	CC-Link Slave Protocol API XX EN.pdf (English Version)
DeviceNet Slave	Description of DeviceNet Slave Protocol API	DeviceNet Slave Protocol API XX EN.pdf (English Version)
PROFIBUS DP Slave	Description of PROFIBUS DP Slave Protocol API	PROFIBUS-DP Slave Protocol API XX EN.pdf (English Version)

Table 3: References

1.5 Legal Notes

1.5.1 Copyright

1.5.2 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.

1.5.3 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.

1.5.4 Export

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.

1.5.5 Registered Trademarks

Windows® 2000, Windows® XP, Windows® Vista und Windows® 7 are registered trademarks of Microsoft Corporation.

Adobe-Acrobat® is a registered trademark of the Adobe Systems Incorporated.

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

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

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

SERCOS interface® is a registered trademark of SERCOS International e. V., Suessen, Germany.

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

2 Setting Parameters via Modbus RTU

If you want to configure the netIC device by writing parameter data into the register area you can use Modbus RTU commands. Modbus function code 16 is used for writing and function code 3 for reading parameter data.

Note: Modbus RTU Parameters can only be set when the netIC operates as Modbus RTU Slave.

Note: Configuration via Modbus RTU is not possible for the NIC50-RE/S3S as it is not supported by the SERCOS III Slave Protocol Stack V3.

Register Areas

There are 2 possible register areas necessary for configuration via Modbus RTU. See *Figure 1: Register Area* on page 10 showing these register areas.

- Parameters of the System Configuration Block
  Register addresses 100 to 199
  These contain SSIO, Modbus RTU and Mapping.
- Parameters of the Network Configuration Data
  Register addresses 300 to 987
  These contain the protocol-specific parameter.

Order of Data

Modbus RTU transfers 16-bit values (registers) in the Motorola format („Big Endian“): First the high byte, then the low byte is transferred. netIC, however, uses the Intel format („Little Endian“). Here first the low byte, then the high byte of a 16 bit word is stored. Therefore, the Modbus parameter „swap“ is by default set to „1“ causing an internal swap of low and high bytes.

At parameters that contain 2 registers, the low-order part (low word) of the parameter value is stored first. The high-order part (high word) of the parameter value is stored on the following register.

Example:

A parameter is located on register addresses 311 and 312. Then the low-order part (low word) of the parameter value is located at register address 311 and the high-order part (high word) of the parameter value is located on register address 312.

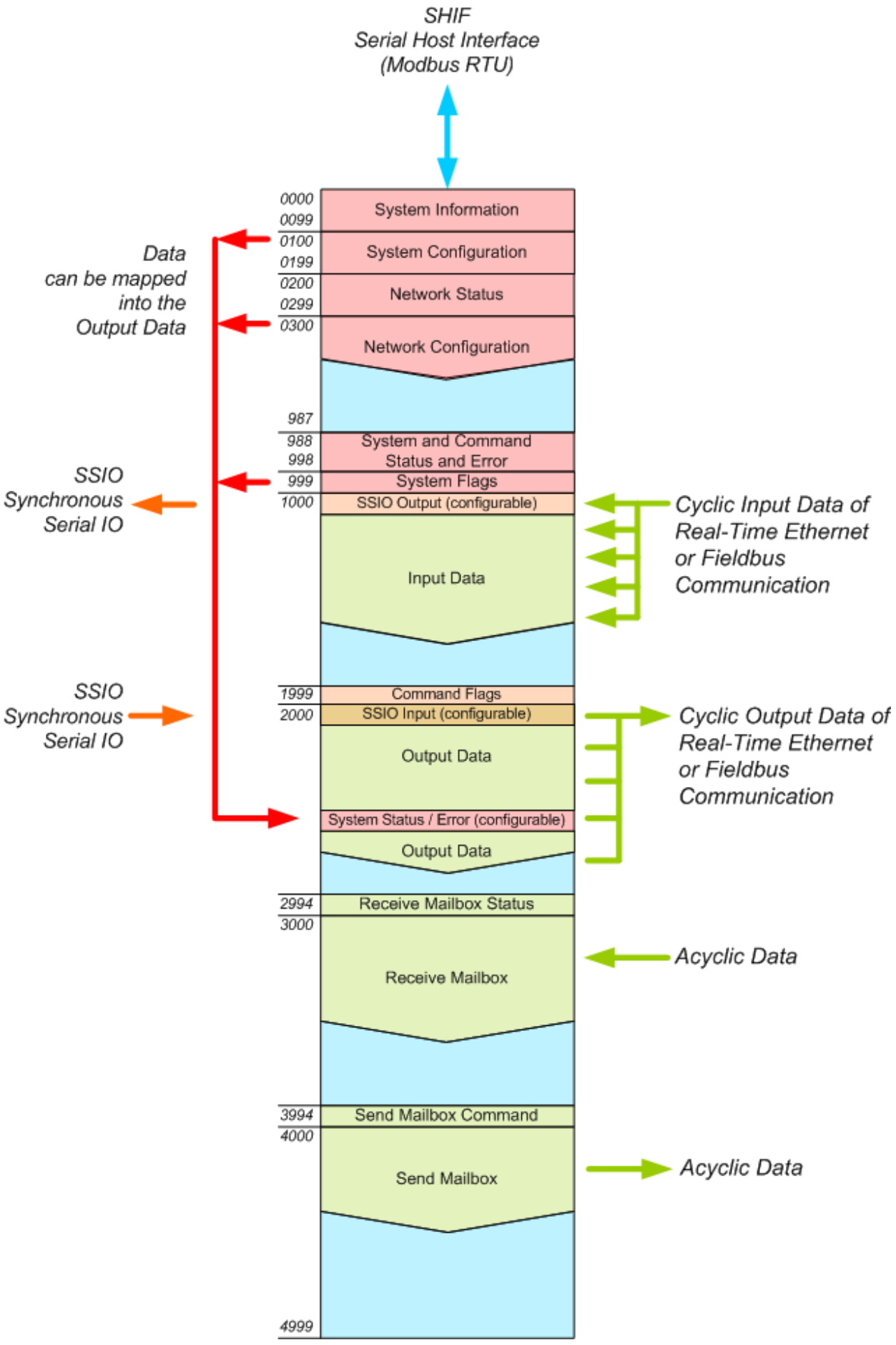

Figure 1: Register Area

General Procedure

In order to configure the netIC via Modbus RTU, proceed as follows:

- Write all configuration parameters to the corresponding registers using function code 16.
- Set the INIT flag within the command register at register address 1999 (See section *"Command Flags"* of the netIC User Manual and Design Guide.) .

Note: After a reset of the netIC device, the configuration parameters are read from the Flash memory. To store values into the flash memory see below.

The netIC starts with the new configuration parameters.

Procedure to store Configuration Parameters in the Flash Memory

To configure the netIC via Modbus RTU permanently so that it will continue to work with the same parameters after a reset, you need to store the configuration parameters in the flash memory. In order to do so, proceed as follows:

1. Write all configuration parameters to the corresponding registers using function code 16.
2. Set the CLR_CFG flag at register address 1999 within the command register to clear the configuration parameter in the Flash memory
3. Check status register 999 until the flag FLS_CFG has been cleared. This is necessary as erasing the configuration may last for some time.
4. Set the STR_CFG flag at register address 1999 within the command register to store the configuration parameter in the Flash memory
5. Check status register 999 until the flag FLS_CFG has been set again. This is necessary as storing the configuration may last for some time.
6. Set the INIT flag at register address 1999 within the command register.
7. The netIC starts with the new configuration parameters.

For detailed information about the flags within the command register at register address 1999, see section *"Command Flags"* of the netIC User Manual and Design Guide.).

2.1 Examples

Two examples may clarify the relationship between the parameters of the netX Configuration Tool and those available for Modbus RTU configuration:

- SSIO parameter (Parameters of the System Configuration Block)

If you want to access the number of input and output bytes for SSIO, you find these in the **Operating Instruction Manual netX Configuration Tool for netIC 50**: Access register 104 for *SSIO Number of Input Byte*s and register 105 for *SSIO Number of Output Bytes*.

- Parameters of the Network Configuration Data

If you want to access the length of the input data bytes of PROFINET IO device you find these in the **Operating Instruction Manual netX Configuration Tool for netIC 50**. Access register 311 (Low word). Register 312 (High word) has always value zero.

2.2 Configuration Process at Power-Up

When the netIC power is switched on, the firmware checks whether the Flash memory contains configuration parameters.

When configuration parameters are available in the Flash memory, then these are read from the Flash memory and are stored in the corresponding registers as default value. (Register addresses 100 to 199 and 300 to 987). This allows the user to read back the currently active configuration from the corresponding registers. When the user wants to modify only one particular network or any system parameter he just has to modify the corresponding register (e.g. the watchdog time or the PROFIBUS slave address) and reinitialize the netIC via Modbus RTU by setting the INIT flag within the command register. For more information, see section *"Command Flags"* of the netIC User Manual and Design Guide.

Note: After a reset of the netIC device, the configuration parameters are read from the Flash Memory (similarly to netIC power on).

For a description how to store values into the flash memory, see section „*Setting Parameters via Modbus RTU*“ from page 9.

When the firmware finds no configuration parameters in the Flash memory, then the corresponding registers are initialized with zero. The default value for the Modbus RTU baudrate is 9600 baud.

3 Configuring the Parameters of the System Configuration Block (Register Area 100 – 199)

This section explains the protocol-independent part of system configuration by accessing the registers of the System Configuration Block via Modbus RTU.

3.1 Synchronous Serial Input/Output (SSIO)

The following table shows the SSIO registers to be configured:

Register	Parameter	Meaning	Range of Value / Value
100	Mode	Operation mode of synchronous serial IO interface **This parameter is currently not supported!**	0 (fixed value, denotes Shift Register)
101	Address	For future use **This parameter is currently not supported!**	0
102-103	Baud Rate	Transmission speed of synchronous serial IO interface	0 (Automatic) 500: SPI not in use 100000: SPI baudrate = 100KHz 200000: SPI baudrate = 200KHz 500000: SPI baudrate = 500KHz 1000000: SPI baudrate = 1MHz 2000000: SPI baudrate = 2MHz 5000000: SPI baudrate = 5MHz
104	Output Bytes	Number of output bytes in synchronous serial IO	0 … 256
105	Input Bytes	Number of input bytes in synchronous serial IO	0 … 256

Table 4: Synchronous Serial Input/Output Parameters

Note on NIC 50-REFO: At the NIC 50-REFO there is no synchronous serial input-/ output interface. The parameters addressed via register 100-105 are not applicable in this gateway.

3.2 Serial Host Interface (SHIF)

The following table shows the Serial Host Interface registers to be configured:

Register	Parameter	Meaning	Range of Value / Value
106	SHIF Type	0 = Modbus RTU / UART 1 = Modbus RTU / SPI All other values are reserved.	0 … 1
107	SHIF Baudrate	With SHIF Type = Modbus RTU / UART: - Modbus RTU Baudrate With SHIF Type = Modbus RTU / SPI: - Reserved.	With SHIF Type = Modbus RTU / UART: - See Table 6: Contents of Baudrate Register below With SHIF Type = Modbus RTU / SPI: - Set to 0.
108	SHIF Address	Defines the own Modbus RTU address of the netIC Gateway device.	1 … 247, Default: 2
109	SHIF Configuration flags	With SHIF Type = Modbus RTU / UART: - PARITY_NONE 0x00000000 - PARITY_EVEN 0x00000001 - PARITY_ODD 0x00000002 - RTS_ON 0x00000004 - ENABLE_SWAP 0x00000010 With SHIF Type = Modbus RTU / SPI: - INCLUDE_CRC_AND_ADDR 0x00000020	Flags

Table 5: Serial Host Interface Parameters

3.3 Switch Modbus RTU Baudrate via Modbus RTU

The baudrate for Modbus RTU of the netIC device can be set (switched) by the Modbus RTU Master. Therefore the netIC has to run in Modbus RTU Slave mode.

Register

Register 107 contains the value for the baudrate and contains:

Bit	Description
0 .. 14	Value for the baudrate (x 100) 12 = 1200 Baud 24 = 2400 Baud 48 = 4800 Baud 96 = 9600 Baud 192 = 19200 Baud 384 = 38400 Baud 576 = 57600 Baud 1152 = 115200 Baud
15	Write protection flag To avoid that the baudrate can be changed unintendedly during runtime operation, a write protection flag is available. 0 - Write protection off 1 - Write protection on (active)

Table 6: Contents of Baudrate Register

Process

1. Write with function code 16 the new baudrate into register 107.
2. The netIC switches the baudrate to the new value.
3. Set the new baudrate at the Modbus RTU Master.
4. The baudrate switch must be confirmed by the Modbus RTU Master by rewriting register 107 within 2000 ms with the same value as in step 1.

Note to step 4: The confirmation by rewriting register 107 must be done within 2000 milliseconds after netIC switched to the new baudrate. When the confirmation of the new baudrate in not done by the Modbus RTU Master within 2000 ms, then the netIC device will switch back to its old baudrate! If no baudrate had been initialized, the netIC will switch to the default baudrate of 9600 Baud.

5. Afterwards, communication via Modbus RTU is performed with the new baudrate.

3.4 Data Mapping

The following table shows the data mapping registers to be configured:

The mapping parameters are:

Register	Parameter	Meaning	Range of Value / Value
SSIO-Mapping			
110	No. Of Config Register	Width of config register specified as number of bytes	0 …10 Default: 0
111	No. Of Status Register	Width of status register specified as number of bytes	0 …10 Default: 0
112	Offset Add. I_Data	Offset Address Input Data	0 … 1800 Default: 0
113	Offset Addr. O_Data	Offset Address Output Data	0 … 1800 Default: 0
114	SSIO Watchdog Time	Watchdog Time for SSIO	
115-119	Reserved	Reserved for future use	0
Diag-Mapping			
120	Offset Addr. Diag_Data	Offset Address Diagnostic Data	0 … 1800 Default: 0
121	No. Mapping Data	Number of Mapping Data	0..5
122-199	Mapping Data	Mapping Data: ID1, Length1, ID2, Length 2, …	

Table 7: Data-Mapping Parameter

4 Configuring the Parameters of the Network Configuration Data (Register Area 300 - 987)

This section explains the protocol-dependent part of system configuration by accessing the registers of the Network Configuration Block via Modbus RTU.

The following protocols are covered:

Table 8: Supported Protocols at Configuration via Modbus RTU

4.1 CANopen Slave

Register	Parameter	Meaning	Range of Value/Value
300	Network Configuration Data Length	Total length (specified in number of bytes) of the following configuration data. If this value does not match the actual size of the configuration, the configuration will be rejected!	
301-302 Bit 0	Bus Startup	Communication start application controlled or automatic	0: Automatic (Default) 1: Application controlled
303-304	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
301-302 Bits 1 and 2	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)	0: None, (1: 1 Byte, 2: 4 Byte) Default: 0 (None)
305-306	Node Address	Node ID of CANopen slave	1 … 127, Default: 2
307-308	Baudrate	Baud rate of CANopen connection	0: 1 MBaud, 1: 800 kBaud, 2: 500 kBaud, 3: 250 kBaud, 4: 125 kBaud, 5: 100 kBaud, 6: 50 kBaud, 7: 20 kBaud, 8: 10 kBaud, Default: NIC50-COS. 0=1 MBaud
309-310	CANopen Flags		
309-310 Bit 0:	29-BIT IDENTIFIER DISABLED/ ENABLED	29-BIT IDENTIFIER DISABLED/ ENABLED (Currently not supported)	0: 11 bit (default)
309-310 Bit 1- 3	(Reserved)	Reserved for future use, set to 0.	0
309-310 Bit 4	Vendor ID DISABLED/ENABLED	This flag determines whether the Vendor ID parameter (Register 311-312) is enabled or disabled. 0: Vendor ID DISABLED: For the Vendor ID, the default value (identification of Hilscher, 0x44) is used. The parameter “Vendor ID” is not evaluated at all. 1: Vendor ID ENABLED: The parameter “Vendor ID” is evaluated.	0: Vendor ID DISABLED 1: Vendor ID ENABLED
309-310 Bit 5	Product Code DISABLED/ENABLED	This flag determines whether the Product Code parameter (Register 313-314) is enabled or disabled. 0: Product Code DISABLED For the Product Code, the default value is used. The parameter “Product Code” is not evaluated at all. 1: Product Code ENABLED: The parameter “Product Code” is evaluated.	0: Product Code DISABLED 1: Product Code ENABLED

For more see next page

309-310 Bit 6	Serial Number DISABLED/ENABLED	This flag determines whether the Serial Number parameter (Register 315-316) is enabled or disabled. 0: Serial Number DISABLED: For the Serial Number, the default value is used. The parameter “Serial Number” is not evaluated at all. 1: Serial Number ENABLED: The parameter “Serial Number” is evaluated.	0: Serial Number DISABLED 1: Serial Number ENABLED
309-310 Bit 7	Revision Number DISABLED/ENABLED	This flag determines whether the Revision Number parameter (Register 317-318) is enabled or disabled. 0: Revision Number DISABLED: For the Revision Number, the default value is used. The parameter “Revision Number” is not evaluated at all. 1: Revision Number ENABLED: The parameter “Revision Number” is evaluated.	0: Revision Number DISABLED 1: Revision Number ENABLED
309-310 Bit 8-31	(Reserved)	Reserved for future use, set to 0.	0
311-312	Vendor ID	Identification number of the manufacturer Only applicable if Vendor ID ENABLED (Register 309 Bit 4 = 1).	0x00000000 ... 0x0000FFFF (hex), Hilscher: 0x00000044 (hex)
313-314	Product Code	Product code of the device as specified by the manufacturer Only applicable if Product Code ENABLED (Register 309 Bit 5 = 1).	0x00000000 ... 0xFFFFFFFF (hex), Default: NIC 50-COS: 0x001785A4 (hex)
317-318	Revision Number	Revision number of the device as specified by the manufacturer Only applicable if Revision Number ENABLED (Register 309 Bit 7 = 1).	0x00000000 ... 0xFFFFFFFF (hex), Default: 0x00020000 (hex)
315-316	Serial Number	Serial number of the device Only applicable if Serial Number ENABLED (Register 309 Bit 6 = 1).	0x00000000 ... 0xFFFFFFFF (hex)

Table 9: CANopen-Slave-Parameter

Note: To configure the Master which is connected to the NIC50-COS, 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 each. The number of output data bytes and input data bytes is not adjustable in the Slave.

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 sets the number of output data bytes and input data bytes at this same time.

For more information refer to the CANopen Slave Protocol API Manual.

4.2 CC-Link Slave

Register	Parameter	Meaning	Range of Value/Value
300	Network Configuration Data Length	Total length (specified in number of bytes) of the following configuration data. If this value does not match the actual size of the configuration, the configuration will be rejected!	
301-302 Bit 0	Bus Startup	Communication start application controlled or automatic	0: Automatic (Default) 1: Application controlled
303-304	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
301-302 Bits 1 and 2	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)	0: None, (1: 1 Byte, 2: 4 Byte) Default: 0: None
305-306	CC-Link Flags		
305-306 Bit 0	CONFIG VENDOR CODE DISABLED/ENABLED	This flag determines whether the parameter “Vendor Code”(Register 323-324) is enabled or disabled. 0: Default Vendor Code is used 1: Value from parameter ‘VendorCode’ is used	0,1
305-306 Bit 1	CONFIG MODEL TYPE DISABLED/ENABLED	This flag determines whether the parameter “Model Type” (Register 325-326) is enabled or disabled. 0: Default Model Type is used 1: Value from parameter ‘ModelType’ is used	0,1
305-306 Bit 2	CONFIG SW VERSION DISABLED/ENABLED	This flag determines whether the parameter “Software-Version” (Register 327-328) is enabled or disabled. 0: Default Software Version is used 1: Value from parameter ‘Software Version’ is used	0,1
307-308	Slave 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
309-310	Baud Rate	Network transmission rate	0: 156 kBaud 1: 625 kBaud 2: 2500 kBaud 3: 5 MBaud 4: 10 MBaud Default: 0 (156 kBaud)
311-312	Station Type	Type of CC-Link station 0: Remote I/O Station: 1: Remote Device Station	0,1 Default: 0 (Remote I/O Station)
313-314	Number of Stations	Number of occupied stations Remote I/O Station: Remote Device Station:	 1 (Default) 1 ... 4

For more see next page

315-316	CC-Link Version	CC-Link Version 1: CC-Link Version 1 2: CC-Link Version 2	1,2 Default: 1 (CC-Link Version 1)
317-318	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
319-320	I/O Types and Points	I/O types and points	0..16
321-322 321 Bit 0	Hold last received Output Data	Hold Clear Mode; Behavior in case of bus error Clear output data (0) Hold last received output data (1)	Default: 0 (Clear output data)
323-324	Vendor Code	Code for the Vendor Only applicable if CONFIG VENDOR CODE ENABLED (Register 305 Bit 0 = 1)	0 ... 65535 bzw. 0x00000000 ... 0x0000FFFF (hex), Hilscher: 0x00000352 (hex)
325-326	Model Type	Model type Only applicable if CONFIG MODEL TYPE ENABLED (Register 305 Bit 1 = 1)	0 ... 255, Default: 2
327-328	Software Version	Software version Only applicable if CONFIG SW VERSION CODE ENABLED (Register 305 Bit 2 = 1)	0 ... 63, Default NIC 50-CCS: 2

Table 10: CC-Link Slave Parameters

The parameter *I/O Types and Points* can be used to adjust:

- the total number of I/O points
- the I/O types

Depending on the number of occupied stations, the total number of I/O points can have the following values:

- 8 points
- 16 points
- 32 points

The following: I/O types are available:

- Mixed
- Input
- Output
- Composite

Mixed in this context applies to the situation when both input and output exist on the same module. The same I/O numbers are used, see CC-Link specification.

Composite means a device that doesn't use the same numbers for input and output.

The following values are possible for parameter `ulIoTypesPoints`:

Code	Value	total number of I/O points	I/O type
CCLINK_SLAVE_IO_TYPES_POINTS_DEFAULT	0x0000L	Use this value to work with default settings	
CCLINK_SLAVE_IO_TYPES_POINTS_MIXED_DEP_ON_STATION	0x0001L	*	*Mixed*
CCLINK_SLAVE_IO_TYPES_POINTS_MIXED_8POINTS	0x0002L	8	*Mixed*
CCLINK_SLAVE_IO_TYPES_POINTS_MIXED_16POINTS	0x0003L	16	*Mixed*
CCLINK_SLAVE_IO_TYPES_POINTS_MIXED_32POINTS	0x0004L	32	*Mixed*
CCLINK_SLAVE_IO_TYPES_POINTS_INPUT_DEP_ON_STATION	0x0005L	*	*Input*
CCLINK_SLAVE_IO_TYPES_POINTS_INPUT_8POINTS	0x0006L	8	*Input*
CCLINK_SLAVE_IO_TYPES_POINTS_INPUT_16POINTS	0x0007L	16	*Input*
CCLINK_SLAVE_IO_TYPES_POINTS_INPUT_32POINTS	0x0008L	32	*Input*
CCLINK_SLAVE_IO_TYPES_POINTS_OUTPUT_DEP_ON_STATION	0x0009L	*	*Output*
CCLINK_SLAVE_IO_TYPES_POINTS_OUTPUT_8POINTS	0x000AL	8	*Output*
CCLINK_SLAVE_IO_TYPES_POINTS_OUTPUT_16POINTS	0x000BL	16	*Output*
CCLINK_SLAVE_IO_TYPES_POINTS_OUTPUT_32POINTS	0x000CL	32	*Output*
CCLINK_SLAVE_IO_TYPES_POINTS_COMPOSITE_DEP_ON_STATION	0x000DL	*	*Composite*
CCLINK_SLAVE_IO_TYPES_POINTS_COMPOSITE_8POINTS	0x000EL	8	*Composite*
CCLINK_SLAVE_IO_TYPES_POINTS_COMPOSITE_16POINTS	0x000FL	16	*Composite*
CCLINK_SLAVE_IO_TYPES_POINTS_COMPOSITE_32POINTS	0x0010L	32	*Composite*

Table 11: Possible Values for Parameter `ulIoTypesPoints`

* means the total number of I/O points depends on the number of occupied stations.

Number of 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: 12 … 368 Bytes
output data: 12 … 368 Bytes

Firmware/stack works according to CC-Link Version 1.11

input data: 4 … 48 Bytes
output data: 4 … 48 Bytes

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.

For more information refer to the CC Link Slave Protocol API Manual.

4.3 DeviceNet Slave

Register	Parameter	Meaning	Range of Value / Value
Interface			
300	Network Configuration Data Length	Total length (specified in number of bytes) of the following configuration data. If this value does not match the actual size of the configuration, the configuration will be rejected!	
301-302 Bit 0	Bus Startup	Communication start application controlled or automatic	0: Automatic 1: Application controlled Default: 0 (Automatic)
303-304	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
301-302 Bits 1 and 2	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)	0: None, (1: 1 Byte, 2: 4 Byte) Default: 0 (None)
Bus			
305-306	MAC ID	This parameter defines the DeviceNet address of the device within the network.	0 … 63, Default: 2
307-308	Baudrate	Baud rate of DeviceNet connection (meaning of values see *Table 13: Available Baud Rate Values*)	0…2 Default: 0: 500 kBaud
Data			
309-310	Prod. Data Length	Produced data length sets the number of send bytes.	0 … 255, Default: 8
311-312	Cons. Data Length	Consumed data length sets the number of receive bytes.	0 … 255, Default: 8
Ident			
313-314	ConfigFlags	The variable ConfigFlags defines configuration parameters, see below. Detailed description of flags see below!	0 … 255 Default: 0
315-316	EnableFlags	The variable *EnableFlags* indicates to the DNS-Task of the DeviceNet Slave Stack if the following identity variables ProductType, ProductCode, MinorRev, MajorRev and the ProdName are filled with user application specific data. Detailed description of flags see below!	0x00000000 - 0x0000003F 0 denotes: Settings from stack are applied
317	Vendor ID	Identification number of the manufacturer Only applicable, if EnableFlag `MSK_DNS_ENABLE_VENDORID` has been set (Register 315 Bit 0 = 1)	0x00000000 ... 0x0000FFFF (hex), Default: Hilscher (Changeable): 0x00000011B (hex)

Register	Parameter	Meaning	Range of Value / Value
318	Product Type	Communication Adapter Only applicable, if EnableFlag MSK_DNS_ENABLE_PRODUCTTYPE has been set (Register 315 Bit 1 = 1)	0x00000000 ... 0x0000FFFF (hex), Default: 0x00000000C (hex)
319	Product Code	Product code of the device Only applicable, if EnableFlag MSK_DNS_ENABLE_PRODUCTCODE has been set (Register 315 Bit 2 = 1)	0x00000000 ... 0xFFFFFFFF (hex), Default NIC 50-DNS: 0x00000023(hex)
320 Low byte	Major Rev	Major Revision Only applicable, if EnableFlag MSK_DNS_ENABLE_MAJORMINORREV has been set (Register 315 Bit 3 = 1)	1 ... 255, Default: 1
320 High byte	Minor Rev	Minor Revision Only applicable, if EnableFlag MSK_DNS_ENABLE_MAJORMINORREV has been set (Register 315 Bit 3 = 1)	1 ... 255, Default: 1
321-322	Serial Number	Serial number of the device Only applicable, if EnableFlag MSK_DNS_ENABLE_SERIALNR has been set (Register 315 Bit 4 = 1)	0x00000000 ... 0xFFFFFFFF (hex)
323 Low byte	Reserved area	Reserved	
323 High byte	Reserved area/ Length	Lower 8 bits: reserved Upper 8 bits: contains length of subsequent variable Product Name (max. 32) Only applicable, if EnableFlag MSK_DNS_ENABLE_PRODUCTNAME has been set (Register 315 Bit 5 = 1)	
324-339	Product Name	The variable *Product Name* is a text string that should represent a short description of the product/product family. Only applicable, if EnableFlag MSK_DNS_ENABLE_PRODUCTNAME has been set (Register 315 Bit 5 = 1)	0 ... 31 ASCII Characters

Table 12: DeviceNet Slave Parameters

Additional explanations for the single parameters:

Watchdog time

The watchdog time parameter specifies the time in multiples of 1msec. the device has to supervise the host program if it has started the host-watchdog functionality once. Read in manual 'Toolkit General definitions' how to activate and deactivate the device and host supervision.

MAC ID (Node ID)

The MAC ID parameter defines the device's DeviceNet address within the network. Valid choices are in the range from 0 to 63, other values are not allowed by the DeviceNet specification. Thus they are rejected by the DeviceNet Slave Protocol Stack.

Baudrate

The applicable baud rates can be coded according to the values specified in the following table:

Baud rate	Value
Baud rate 500 kBit/s	0
Baud rate 250 kBit/s	1
Baud rate 125 kBit/s	2

Table 13: Available Baud Rate Values

ProducedSize

The ProducedSize parameter specifies the number of input bytes the device shall produce in the view of a master for each established connection. The bytes which shall be produced then must be handed over in the send data area of the dual-port memory.

ConsumedSize

The ConsumedSize parameter specifies the number of output bytes the device shall consume in the view of a master for each established connection. The bytes which are received are handed over in the receive data area of the dual-port memory.

ConfigFlags Byte

The following table shows the meaning of the ConfigFlags byte:

ConfigFlags Byte - Bits D0-D8 (Register 313 Low Byte, D8 High Byte)		
Bit	**Name**	**Description**
D8	MSK_DNS_CFG_FLAG_ENABLE_SET_BAUDRATE_REMOTE	Enable Remote Setting of Baudrate
D7	MSK_DNS_CFG_FLAG_ENABLE_SET_MACID_REMOTE	Enable Remote Setting of MAC ID
D6	MSK_DNS_CFG_FLAG_ENABLE_SET_PRODCONS_SIZE_REMOTE	Enable Remote Setting of Producer Size and Consumer Size
D5	MSK_DNS_CFG_FLAG_24VDCINVERT	Inverted handling of 24 V Network Power
D4	MSK_DNS_CFG_FLAG_RECVIDLE_USER_DATA	User data on receive idle
D3	MSK_DNS_CFG_FLAG_RECVIDLE_CLEAR_DATA	Clear data on receive idle
D2	MSK_DNS_CFG_FLAG_CONTINUE_ON_LOSS_NP	Continue on loss of NP
D1	MSK_DNS_CFG_FLAG_CONTINUE_ON_BUSOFF	Continue on Bus-off
D0	MSK_DNS_CFG_FLAG_IGNORE_ADDR_SWITCH	Ignore Address Switch

Table 14: Meaning of ConfigFlags Byte

If these bits are set to 1, the appropriate item will be activated, otherwise deactivated.

The single bits of the flags variable have the following meaning:

Bit ***0*** Ignore Address Switch (`MSK_DNS_CFG_FLAG_IGNORE_ADDR_SWITCH`):

This flag indicates whether the address switch is enabled.

If set to 1, the DeviceNet MAC-ID provided by the hardware address switches will be ignored. Instead the MAC-ID will be read from the bOwnMacId configuration variable after a warmstart. This applies to devices which feature hardware address switches, only. All other devices will ignore the `MSK_DNS_CFG_FLAG_IGNORE_ADDR_SWITCH` flag.

Otherwise, hardware address switches will not be ignored if present.

Bit 1 Continue on Bus-off (`MSK_DNS_CFG_FLAG_CONTINUE_ON_BUSOFF`):

This flag defines the behavior of the device in case a 'bus off' indication from the CAN controller occurs. If the `MSK_DNS_CFG_FLAG_CONTINUE_ON_BUSOFF` flag is set to the value 1 the device will continue its operation. Otherwise, the device will stop operation.

Bit 2 Continue on loss of NP (`MSK_DNS_CFG_FLAG_CONTINUE_ON_LOSS_NP`):

Currently not supported.

Bit 3 Clear data on receive idle (`MSK_DNS_CFG_FLAG_RECVIDLE_CLEAR_DATA`):

This flag defines the behavior of the device in case the master sends receive_idle telegrams in an established poll connection. Each master has the possibility to send no input data to the slave with a special poll commands. The slave on the other hand is still forced to send back its latest output data and keep the master updated.

This is procedure defined in the DeviceNet specification.

If the `MSK_DNS_CFG_FLAG_RECVIDLE_CLEAR_DATA` flag is set to the value 0 the device will hold the last received input data in the input process image.

If the value is set to 1 the device will clear the input area in the process image with the value 0.

Bit 4 User data on receive idle (MSK_DNS_CFG_FLAG_RECVIDLE_USER_DATA):

Currently not supported.

Bit 5 Inverted handling of Network Power (MSK_DNS_CFG_FLAG_24VDCINVERT):

Handles the 24V Network Signal inverted, if the user hardware does not follow the standard or have hardware reference of Hilscher NXSB-100 board.

Bit 6 Enable Remote Setting of Producer Size and Consumer Size:

Via Remote Service, the DeviceNet Master may request a change of the Producer Size or Consumer Size. This flag allows to enable (flag set) or disable (flag cleared) such changes. When the flag is set, then set_attribute_single request from client to attributes 7 and 8 (produced connection size and consumed connection size) of connection object will be automatically handled by stack application task. When the flag is not set and if connection class is not registered by second-level application task then the set_attribute_single request from client will be answered with general code: service not supported (0x08). When the flag is not set and if connection class is already registered by second-level application task using DNS_FAL_CMD_REGISTER_CLASS_REQ then the set_attribute_single request from client will be forward to second-level application task.

Bit 7 Enable Remote Setting of MAC ID:

Via Remote Service, the DeviceNet Master may request a change of the MAC ID. This flag allows to enable (flag set) or disable (flag cleared) such changes. When the flag is set, then set_attribute_single request from client to attributes 1 (MAC ID) of DeviceNet object will be automatically handled by stack application task. When the flag is not set and if DeviceNet class is not registered by second-level application task then the set_attribute_single request from client will be answered with general code: service not supported (0x08). When the flag is not set and if connection class is already registered by second-level application task using DNS_FAL_CMD_REGISTER_CLASS_REQ then the set_attribute_single request from client will be forward to second-level application task.

Bit 8 Enable Remote Setting of Baudrate:

Via Remote Service, the DeviceNet Master may request a change of the Baudrate. This flag allows to enable (flag set) or disable (flag cleared) such changes. When the flag is set, then set_attribute_single request from client to attributes 2 (Baudrate) of DeviceNet object will be automatically handled by stack application task. When the flag is not set and if DeviceNet class is not registered by second-level application task then the set_attribute_single request from client will be answered with general code: service not supported (0x08). When the flag is not set and if connection class is already registered by second-level application task using DNS_FAL_CMD_REGISTER_CLASS_REQ then the set_attribute_single request from client will be forward to second-level application task.

EnableFlags Byte

The EnableFlags byte decides whether the following identity variables ProductType, ProductCode, MinorRev, MajorRev and the ProductName are filled up with valid data, or not.

The following table shows the meaning of the EnableFlags byte:

EnableFlags Byte - Bits D0-D7 (Register 315 Low Byte)		
Bit	**Name**	**Description**
D7	-	**-**
D6	-	-
D5	MSK_DNS_ENABLE_PRODUCTNAME	Enable Flag for ProductName
D4	MSK_DNS_ENABLE_SERIALNR	Enable Flag for Serial Number
D3	MSK_DNS_ENABLE_MAJORMINORREV	Enable Flag for MajorRev and Minorrev
D2	MSK_DNS_ENABLE_PRODUCTCODE	Enable Flag for ProductCode
D1	MSK_DNS_ENABLE_PRODUCTTYPE	Enable Flag for ProductType
D0	MSK_DNS_ENABLE_VENDORID	Enable Flag for Vendor ID

Table 15: Meaning of EnableFlags Byte

If the corresponding bit is set to 1, the variable contains valid data and is enabled, i.e. the appropriate item (Vendor ID, product type, product code, minor and major revision together, serial number and product name) can be set by the user, otherwise this is not possible.

ProductType

The variable ProductType is the indication of the general type of product.. The Hilscher standard value for this is 12 which is a Communications Adapter.

ProductCode

The variable ProductCode is the identification of a particular product within a device type.

MinorRev

The variable MinorRev is one part of the revision which identifies the revision of the device. The revision attribute consists of Major and Minor Revisions and they are typically displayed as *major.minor*.

MajorRev

The variable MajorRev is the second part of the revision. The Major Revision attribute is limited to 7 bits. The eighth bit is reserved by DeviceNet and must have a default value of zero.

ProductName

The variable ProductName is a text string that should represent a short description of the product/product family. The maximum number of characters in this string is 32. The number of characters must be set in the variable `bLength` which is the first byte in the structure `tProduct`.

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.

For more information refer to the DeviceNet Slave Protocol API Manual.

4.4 EtherCAT Slave

Register	Parameter	Meaning	Range of Value / Value
Interface			
300	Network Configuration Data Length	Total length (specified in number of bytes) of the following configuration data. If this value does not match the actual size of the configuration, the configuration will be rejected!	
301-302 Bit 0	Bus Startup	Communication start application controlled or automatic	0: Automatic 1: Application controlled (Default for NIC 50-RE/ECS)
303-304	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
301-302 Bits 1 and 2	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)	0: None, (1: 1 Byte, 2: 4 Byte) Default: 0: None
Ident			
305-306	Vendor ID	Identification number of the manufacturer	0x00000000 ... 0xFFFFFFFF (hex) Primary Hilscher Vendor ID: 0x00000044 (hex) Default: Secondary Hilscher Vendor ID: NIC 50-RE/ECS: 0xE0000044 (hex)
307-308	Product Code	Product code of the device as specified by the manufacturer	0x00000000 ... 0xFFFFFFFF (hex), Default: NIC 50-RE/ECS: 0x0000000B (hex)
309-310	Revision Number	Revision number of the device as specified by the manufacturer	0x00000000 ... 0xFFFFFFFF (hex), Default: NIC 50-RE/ECS: 0x00000000 (hex)
311-312	Serial Number	Serial number of the device	0x00000000 ... 0xFFFFFFFF (hex)

Data			
313-314	Output Data Bytes	Length of the output data in Byte	0 … 1024 Byte Default: 4 Byte
315-316	Input Data Bytes	Length of the input data in Byte	0 … 1024 Byte Default: 4 Byte
317-318	Stack Configuration Flags	Stack Configuration Flags, see detailed description below	
319-320	SII Configuration Flags	SII Configuration Flags, see detailed description below	
321 Low byte	Sync PDI Configuration	Sync PDI configuration, see detailed description below	0…255
321 High byte	Sync Impulse Length	Sync impulse length (in units of 10 ns) , see detailed description below	0…65535 Default: 1000
322 Low byte			
322 High byte	Device type	Device type in object dictionary entry 0x1000	

Table 16: EtherCAT-Slave-Parameter

Stack configuration flags (Register 317-318)

The following flags deactivate the host-controlled update at the DPM firmware:

Stack Configuration Flags - Bits D0-D7 (Register 317 Low Byte)		
Bit	**Name**	**Description**
D7	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_SM7_NO_HOST_UPDATE	Reserved for future use
D6	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_SM6_NO_HOST_UPDATE	Reserved for future use
D5	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_SM5_NO_HOST_UPDATE	Reserved for future use
D4	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_SM4_NO_HOST_UPDATE	Reserved for future use
D3	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_SM3_NO_HOST_UPDATE	Deactivate SM3 (Host - triggered update)
D2	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_SM2_NO_HOST_UPDATE	Deactivate SM2 (Host-triggered update)
D1	Unused	Unused
D0	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_ENABLE_SECOND_FLAG_SET	Enable second flag set for object dictionary, SDO, AP, IDN dictionary, SoE dictionary configuration flags

Table 17: Stack Configuration Flags - Bits D0-D7

The following flags deactivate the bus-controlled update at the DPM firmware:

Stack Configuration Flags - Bits D8-D15 (Register 317 High Byte)		
Bit	**Name**	**Description**
D15	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_SET_FREERUN_ON_SM7	Reserved for future use
D14	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_SET_FREERUN_ON_SM6	Reserved for future use
D13	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_SET_FREERUN_ON_SM5	Reserved for future use
D12	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_SET_FREERUN_ON_SM4	Reserved for future use
D11	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_SET_FREERUN_ON_SM3	Set bus- triggered update of SM3 to free run.
D10	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_SET_FREERUN_ON_SM2	Set bus-triggered update of SM2 to free run.
D9	Unused	Unused
D8	Unused	Unused

Table 18: Stack Configuration Flags - Bits D8-D15

The following flags set the bus-controlled update at the IRQ associated with the sync manager:

Stack Configuration Flags - Bits D16-D23 (Register 318 Low Byte)		
Bit	**Name**	**Description**
D23	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_SET_SELF_UPDATE_ON_SM7	Reserved for future use
D22	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_SET_SELF_UPDATE_ON_SM6	Reserved for future use
D21	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_SET_SELF_UPDATE_ON_SM5	Reserved for future use
D20	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_SET_SELF_UPDATE_ON_SM4	Reserved for future use
D19	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_SET_SELF_UPDATE_ON_SM3	Set bus-controlled update of SM3 to self update.
D18	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_SET_SELF_UPDATE_ON_SM2	Set bus-controlled update of SM2 to self update
D17	Unused	Unused
D16	Unused	Unused

Table 19: Stack Configuration Flags - Bits D16-D23

Miscellaneous flags:

Stack Configuration Flags - Bits D24-D31 (Miscellaneous flags) (Register 318 High Byte)		
Bit	Name	Description
D31	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_ENABLE_DC_MODE_4	Reserved for future use
D30	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_ENABLE_DC_MODE_3	Reserved for future use
D29	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_ENABLE_DC_MODE_2	Reserved for future use
D28	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_ENABLE_DC_MODE_1	Reserved for future use

D27	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_CONFIGURE_BUS_SYNCHRONOUS_MODE	Enables the threshold variables
		`ulSm2ErrorThreshold`
		`ulSm3ErrorThreshold`
		`ulSyncFlagErrorThreshold`

D26	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_ENABLE_SYNC_OUTPUT_CONFIG	Activate sync output reconfiguration
D25	MSK_ECAT_DPM_WARMSTART_STACK_CFG_DO_NOT_CREATE_DEFAULT_OBJECTS	Create the default object set according to the configuration if not already set (default: not set)
D24	MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_CLEAR_APPLICATION_OBJECTS	Clear all application-specific objects (default: not set)

Table 20: Stack Configuration Flags - Bits D24-D31

Logic of Sync Manager Update Triggering

The following rules apply for setting the stack configuration flags:

1. For each available sync manager (SM2 or SM3) host-triggered update and bus-triggered update free run option exclude each other and thus cannot be activated at the same time. This means:

   For Sync Manager 2:

   If bit D2 is set (D2=1, host-triggered update selected), then D10 must be cleared (D10=0, no bus-triggered update free run)

   This also applies vice versa:

   If bit D10 is set (D10=1, bus-triggered update selected free run), then D2 must be cleared (D2=0, no host -triggered update)

   For Sync Manager 3:

   If bit D3 is set (D3=1, host-triggered update selected), then D11 must be cleared (D11=0, no bus-triggered update)

   Vice versa:

   If bit D11 is set (D11=1, bus-triggered update selected), then D3 must be cleared (D3=0, no host-triggered update)

2. For each available sync manager (SM2 or SM3) the self update option and free run option of bus-triggered update exclude each other and thus cannot be activated at the same time. This means:

   For Sync Manager 2:

   If bit D18 is set (D18=1, host-triggered update selected), then D10 must be cleared (D10=0, no bus-triggered update free run)

This also applies vice versa:

If bit D10 is set (D10=1, bus-triggered update selected free run), then D18 must be cleared (D18=0, no host -triggered update)

For Sync Manager 3:

If bit D19 is set (D19=1, host-triggered update selected), then D11 must be cleared (D11=0, no bus-triggered update)

Vice versa:

If bit D11 is set (D11=1, bus-triggered update selected), then D19 must be cleared (D19=0, no host-triggered update)

3. Set bits D24 (`MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_CLEAR_APPLICATION_OBJECTS`) and D25 (`MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_DO_NOT_CREATE_DEFAULT_OBJECTS`) to 1, if it is intended to create own objects (for an own CoE profile). This prevents the protocol stack from updating the SII memory (D24=1) and creating the default object set (D25=1). Otherwise set both to 0 in order to work with the default object set.

   If you intend to activate slave-oriented dynamic PDO mapping for the RxPDO, you need to set bit D25 to 1.

4. Set bit D26

   (`MSK_ECAT_DPM_SET_CONFIG_STACK_CFG_ENABLE_SYNC_OUTPUT_CONFIG`) to 1, if the values for Sync PDI Config and Sync Impulse Length should be taken over. Setting bit D26 to 0 means, that these values are not taken care of.

   If you intend to activate slave-oriented dynamic PDO mapping for the TxPDO, you need to set bit D26 to 1.

SII Configuration Flags (Register 319-320)

Within the SII configuration flags, only the highest significant byte is currently used. See the table below:

SII Configuration Flags		
Bit	Name	Description
D31	Unused	Unused
D30		Unused
D29		Unused
D28		Unused
D27		Unused
D26	MSK_ECAT_DPM_WARMSTART_SII_CFG_DO_NOT_BUILD_TXPDO_INFO	Suppress generation of TXPDO data block in SII
D25	MSK_ECAT_DPM_WARMSTART_SII_CFG_DO_NOT_BUILD_TXPDO_INFO	Suppress generation of RXPDO data block in SII
D24	MSK_ECAT_DPM_WARMSTART_SII_CFG_DO_NOT_UPDATE	Suppress initialization of SII

Table 21: SII Configuration Flags

How to configure Sync0 and Sync1 Signals?

The following variables in the configuration packet affects the Sync signals:

-Bit D26 within `ulStackConfigurationFlags` (see above): This bit shall be set to 1 if user parameters shall be applied.

-`usSyncImpulseLength` (see below): the length of the sync signals is configured here. It is a multiple of 10 ns (value of 10 results in a pulse length of 100ns)

-`bSyncPdiConfig` (see below): this bit field holds the configuration for both Sync signals

Keep in mind that the Distributed Clocks Feature (including the Sync0/1 settings) must be enabled and configured explicit in the configuration of the EtherCAT Master.

Sync PDI Configuration (Register 321 Low byte)

Bit No.	Description
0	SYNC0 Output type: 0 - Push Pull 1 - OpenDrain/OpenSource (depends on bit 1) Note: netX100/500 firmware ignores this bit. They always work as “Push Pull”.
1	SYNC0 Polarity: 0 - low active 1 - high active
2	SYNC0 Output enable/disable: 0 - disabled 1 - enabled
3	SYNC0 mapped to PDI-IRQ: 0 - disabled 1 - enabled
4	SYNC1 Output type: 0 - Push Pull 1 - OpenDrain/OpenSource (depends on bit 5) Note: netX100/500 firmware ignores this bit. They always work as “Push Pull”.
5	SYNC1 Polarity: 0 - low active 1 - high active
6	SYNC1 Output enable/disable: 0 - disabled 1 - enabled
7	SYNC1 mapped to PDI-IRQ: 0 - disabled 1 - enabled

Table 22: Description of the variable `bSyncPdiConfig`

The current settings are mapped into the register 0x151 of the EtherCAT Slave Controller.

Sync Impulse Length

The sync impulse length is specified in units of 10 ns. The default value is 1000.

General Notes:

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 NIC 50 RE ECS V2.**2**.x.xml* is used/updated, the firmware with the version *2.**2**.x* must be use/updated.

Important: If *Application controlled (1)* is chosen and a watchdog error occurs, the stack will not be able to reach the OPERATIONAL or the SAFE_OPERATIONAL state. In this case, a channel reset is required.

4.5 EtherNet/IP Adapter (Slave)

Register	Parameter	Meaning	Range of Value / Value
Interface			
300	Network Configuration Data Length	Total length (specified in number of bytes) of the following configuration data. If this value does not match the actual size of the configuration, the configuration will be rejected!	
301-302 Bit 0	Bus Startup	Communication start application controlled or automatic	0: Automatic 1: Application controlled (Default for NIC 50-RE/EIS)
303-304	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
301-302 Bits 1 and 2	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)	0: None, (1: 1 Byte, 2: 4 Byte) Default: 0: None
Data			
305-306	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 NIC 50-RE/EIS: Default: 32 Byte
307-308	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 NIC 50-RE/EIS: Default: 32 Byte
Bus			
309 Bit 0	IP address available	If this flag is set, the content of the IP-Address parameter (Register 311-312) will be evaluated.	0: IP address not available 1: IP address available Default: 0
309 Bit1	Netmask available	If this flag is set, the content of the NetMask parameter (Register 313-314) will be evaluated. If the flag is not set the device will assume to be an isolated host which is not connected to any network. The Gateway parameter will be ignored in this case.	0: Netmask not available 1: Netmask available Default: 0
309 Bit 2	Gateway available	If this flag is set, the content of the Gateway parameter (Register 315-316) will be evaluated. If the flag is not set the device will assume that there exists no gateway.	0: Gateway not available: 1: Gateway available Default: 0
309 Bit 3	Enable BOOTP	If this flag is set, the device obtains its IP Address, Netmask and Gateway Address from a BOOTP server.	0: No configuration from BOOTP Server 1: Configuration from BOOTP Server Default: 0

For more see next page

309 Bit 4	Enable DHCP	If this flag is set, the device obtains its IP Address, Netmask and Gateway Address from a DHCP server.	0: No configuration from DHCP Server 1: Configuration from DHCP Server Default:1
309 Bit 5..9	Reserved	Reserved for future use	0
309 Bit 10	Auto-Negotiation (Ethernet Port 1)	If this flag is set, the device will auto-negotiate link parameters with the remote hub or switch. This flag will override the Duplex Operation flag (Bit 11) and the Speed Selection flag (Bit 12).	0: No Auto-Negotiation: 1: Auto-Negotiation:
309 Bit 11	Duplex Operation: (Ethernet Port 1)	If this flag is set, full-duplex operation will be enabled. 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.	0: Half Duplex-Operation 1: Full Duplex-Operation
309 Bit 12	Speed Selection (Ethernet Port 1)	If set, the device will operate at 100 MBit/s, else at 10 MBit/s. This parameter will not be in effect, when auto-negotiation (bit 10) is active.	0: 10 MBit/s, 1: 100 MBit/s,
309 Bit 13..14	Reserved	Reserved for future use	
309 Bit 15	Extended Flag	This flag can be used if the device has two Ethernet ports. In that case the two ports can be configured differently regarding “Speed Selection”, “Duplex Operation” and “Auto-Negotiation” If not set, both ports are configured with the same parameters using the bits 10 to 12 of Register 309. If set, port 1 is configured using bits 10 to 12. Port 2 is configured using the bits 10 to 12 of Register 310.	0: Same parameter set 1: Different parameter set
310 Bit 0	Reserved	Reserved for future use	
310 Bit 1..9	Reserved	Reserved for future use	
310 Bit 10	Auto-Negotiation (Ethernet Port 2):	Only evaluated if Register 309 Bit 15 is set. Behaves the same as Register 309 Bit 10.	0: No Auto-Negotiation: 1: Auto-Negotiation:
310 Bit 11	Duplex Operation: (Ethernet Port 2):	Only evaluated if Register 309 Bit 15 is set. Behaves the same as Register 309 Bit 11.	0: Half Duplex-Operation 1: Full Duplex-Operation
310 Bit 12	Speed Selection (Ethernet Port 2):	Only evaluated if Register 309 Bit 15 is set. Behaves the same as Register 309 Bit 12.	0: 10 MBit/s, 1: 100 MBit/s,
310 Bit 13..15	Reserved	Reserved for future use	
311-312	IP Address	Valid IP address for the device If ‘Enabled’ is not set (Default setting), the device obtains its IP Address from a DHCP server or also from a BOOTP server, if this one is set. If ‘Enabled’ is set, the device uses the manually entered value.	Valid IP address Default value: 0.0.0.0
313-314	Netmask	Valid Network mask for the device If ‘Enabled’ is not set (Default setting), the device obtains its Netmask from a DHCP server or also from a BOOTP server, if this one is set. If ‘Enabled’ is set, the device uses the manually entered value.	Valid network mask Default value: 0.0.0.0
315-316	Gateway	Valid Gateway address for the device If ‘Enabled’ is not set (Default setting), the device obtains its Gateway Address from a DHCP server or also from a BOOTP server, if this one is set. If ‘Enabled’ is set, the device uses the manually entered value.	Valid gateway address Default value: 0.0.0.0

For more see next page

Register	Parameter	Meaning	Range of Value / Value
Ident			
317	Vendor ID	Identification number of the manufacturer	0x00000000 ... 0x0000FFFF (hex), Hilscher 0x00000011B (hex)
318	Product Type	Communication Adapter	0x00000000 ... 0x0000FFFF (hex), Default: 0x00000000C (hex)
319	Product Code	Product code of the device as specified by the manufacturer	0x00000000 ... 0x0000FFFF (hex), Default NIC 50-RE/EIS: 0x00000010D (hex)
320-321	Serial number	Serial number of the device as specified by the manufacturer	
322 Low byte	Minor-Rev	Minor Revision	0 ... 255, Default: 1
322 High byte	Major-Rev	Major Revision	0 ... 255, Default: 1
323-337	Device name	Device name of the device station as character string, e.g. EtherNet/IP Adapter (Slave). 1st Byte contains length of device name in characters, Bytes 1-31 the device name itself.	0 - 31 ASCII characters

Table 23: EtherNet/IP Adapter Parameters

The following rules apply for vendor IDs applicable to EtherNet/IP devices:

1. The value must be contained in the interval [0x0000..0xFFFF].
2. The MSB of the vendor ID must not be 0.
3. The LSB of the vendor ID must not be 0.
4. The MSB of the vendor ID must not be 0xFF.

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 concerning IP Address, Netmask and Gateway

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 set (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 set (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.

Default: Booting from DHCP is activated

4.6 Open Modbus/TCP

Register	Parameter	Meaning	Range of Value / Value
Interface			
300	Network Configuration Data Length	Total length (specified in number of bytes) of the following configuration data. If this value does not match the actual size of the configuration, the configuration will be rejected!	
301-302 Bit 0	Bus Startup	Communication start application controlled or automatic	0: Automatic (Default) 1: Application controlled
303-304	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
OMB-related parameters			
305-306	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.	0..16 (Sockets), Default: 4 (Sockets)
307-308	Response Timeout	Telegram Timeout(in units of 100 ms) 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.	1 ... 60,000 (100 ... 6,000,000 ms) Default:20 (=2,000 ms)
309-310	Client Connection Watchdog Time	Connection remain open time (in units of 100 ms) 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	1 ... 60,000 (100 ... 6,000,000 ms) Default: 10 (=1000 ms)
311-312	Protocol Mode	Mode of data exchange: I/O Server. Decides between operation in Message and IO-Mode Use for netIC devices only locally.	0: Message (packet) Mode 1: IO Server Mode Default: 1 (I/O Server Mode)
313-314	Send Acknowledgement 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 seconds is used.	0 ... 2,000,000,000 ms, Default: 31,000 ms
315-316	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 is used.	0 ... 2,000,000,000 Default: 31,000 (=31 seconds)

For more see next page

Register	Parameter	Meaning	Range of Value / Value
317-318	Close Acknowledgement 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 is used.	0 ... 2,000,000,000 ms, Default: 13,000 (=13 seconds)
319-320	Data Swap	Data-storage mode: 0: Data will not be swapped 1: Data will be swapped	0, 1, Default: 1
TCP/IP related Parameters			
321-322	Flags word	Flags area, see detailed description below	
321 Bit 0		IP address available: If this flag is set (Register 321 Bit 0=1), the content of the *IP Address* parameter (Register 323-324) will be evaluated.	0,1 Default: 0 (not set.)
321 Bit 1		Netmask available: If this flag is set (Register 321 Bit 1=1), the content of the *Net Mask* parameter (register 325-326) will be evaluated. If the flag is not set (=0) the stack will assume to be an isolated host which is not connected to any network. The *Gateway* parameter will be ignored in this case.	0,1 Default: 0 (not set.)
321 Bit 2		Gateway available: If this flag is set (Register 321 Bit 2=1), the content of the *Gateway* parameter (register 327-328) will be evaluated. If the flag is not set (=0), the stack will assume that there is no gateway	0,1 Default: 0 (not set.)
321 Bit 3		BootP If set (Register 321 Bit 3 =1), the device obtains its IP Address, Netmask, Gateway Address from a BOOTP server.	0,1 Default: 0 (not set.)
321 Bit 4		DHCP If set (Register 321 Bit 4 =1), the device obtains its IP Address, Netmask, Gateway Address from a DHCP server.	0,1 Default: 0 (not set.)
321 Bit 5		Set Ethernet address (MAC address): If set (Register 321 Bit 5 =1), the *Ethernet Address* area will be evaluated.	0,1
321 Bit 6 ... 15, 322		Reserved for future use	-
323-324	IP Address	Valid IP address for the device If 'IP address available' is not set (Register 321 Bit 0=0), the device obtains its IP Address from a DHCP or BOOTP server. If 'IP address available' is set(Register 321 Bit 0=1), the device uses the manually entered value.	Valid IP address
325-326	Net Mask	Valid Network mask for the device If 'Netmask available' is not set (Register 321 Bit 1=0), the device obtains its Netmask from a DHCP or BOOTP server. If 'Netmask available' is set(Register 321 Bit 1=1), the device uses the manually entered value.	Valid network mask
327-328	Gateway	Valid IP Address of Gateway for the device If 'Gateway available' is not set (Register 321 Bit 2=0), the device obtains its Gateway Address from a DHCP or BOOTP server. If 'Gateway available' is set(Register 321 Bit 2=1), the device uses the manually entered value.	Valid IP address
329-331	Ethernet Address	Ethernet address	Valid Ethernet address
332 Bit 0 and 1	Map FC1 and FC3 Flag	If not set, data are read from the input area using FC1, FC3 and FC23. If set, 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.	set, not set, Default: not set

Table 24: Open Modbus/TCP Parameters

Note concerning IP Address, Netmask and Gateway

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 set (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 set (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.

Additional explanations for OpenModbus-related Parameters:

Bus Startup (SystemFlags)

This parameter contains the system flags area. Currently this value may only have the values 0 or 1. The start of the device can be performed either application controlled or automatically:

- Automatic (0):

Network connections are opened automatically without taking care of the state of the host application. Communication with a remote controller after a device start is allowed without BUS_ON flag, but the communication will be interrupted if the BUS_ON flag changes state to 0

- Application controlled (1):

The channel firmware is forced to wait for the host application to wait for the Application Ready flag in the communication change of state register (see section 3.2.5.1 of the *netX DPM Interface Manual*). Communication with controller is allowed only with the BUS_ON flag set. Setting the bus on flag can be accomplished by using the RCX_START_STOP_COMM_REQ/CNF packet provided by the OMB_AP –Task and described in section 6.1.8 of the OpenModbus/TCP Protocol API Manual.

The default value is 0 (Automatic). For more information concerning device start-up see section 4.4.1 “*Controlled or Automatic Start*” of the *netX DPM Interface Manual*.

Watchdog Time

This parameter contains the time interval for the supervision of data transfer by the internal watchdog timer. The value must be either the value 0 or a number between 20 and 65535.

If the value 0 is specified, this indicates the watchdog timer has been switched off. Otherwise, the watchdog timer interval is specified in units of milliseconds.

Provided Server Connections (OpenServerSockets)

This parameter describes the number of sockets to provide for server requests.

A value of 0 would mean that the Open Modbus/TCP task exclusively works as a client, while a value of 16 means that the Open Modbus/TCP task exclusively works as server in message-mode. The values 1 … 15 means, that the Open Modbus/TCP task could work as a client and server simultaneous.

Response Timeout (AnswerTimeout)

This parameter describes the telegram timeout

This parameter is only relevant for client jobs in message-mode. After expiration of this time, the job will be canceled and an error is send to the application. Value is multiplied with 100 ms.

Note: This timeout starts after command is send to the destination device via TCP.

Client Connection Watchdog Time (OmbOpenTime)

This parameter describes the connection remain open time. This parameter is only for client jobs in message-mode. The connection to the destination-device stays open, until timeout is expired. The value given in the parameter is multiplied with 100 milliseconds to determine the time value to be applied.

Note: This timeout starts after receiving the answer to a command.

Protocol Mode

This parameter describes the currently active mode of data exchange, either message mode or I/O mode. The possible values are:

Value	Mode
0	Message-Mode
1	IO-Mode

The three following parameters *Send Acknowledgement Timeout, Connect Acknowledgement Timeout* and *Close Acknowledgement Timeout* decide about the timeout between the Open Modbus Task and the TCP Task.

Send Acknowledgement Timeout

This parameter describes the parameter for the TCP task (in milliseconds) . Used OMB task internally. It specifies the timeout value for trying to send messages via TCP/IP

0 represents the default value of 31000 milliseconds.

Connect Acknowledgement Timeout

This parameter describes the parameter for the TCP task (in milliseconds) . Used OMB task internally. It specifies the timeout value for trying to establish a connection with the TCP task.

0 represents the default value of 31000 milliseconds.

Close Acknowledgement Timeout

This parameter describes the parameter for the TCP task (in milliseconds) . Used OMB task internally. It specifies the timeout value for trying to close a connection with the TCP task.

0 represents the default value of 13000 milliseconds.

Data Swap

This parameter decides whether data will be swapped(`Swap = 1`) or not(`Swap = 0`).

Additional explanations for TCP/IP-related Parameters

Flags Word

The *Flags Word* contains bit-oriented flags according to the following table as described in *Table 24: Open Modbus/TCP Parameters*.

Please note, that a fallback procedure between the different configuration methods is active, if more than one choice is enabled in the *Flags Word* parameter. If enabled, the stack will first try to configure using DHCP. If DHCP configuration fails, the stack will fall back to BOOTP, if this is enabled. In case of a BOOTP failure, the values found in the *IP Address*, *Net Mask* and *Gateway* parameters will be used, if enabled in the *Flags Word*. If none of these configuration mechanisms succeed, the stack will report an error.

IP Address

The stacks IP address can be configured using the *IP Address* parameter. This parameter is only effective, if bit 0 of register 321 is set. Otherwise, the values configured via DHCP or BOOTP are applied.

NetMask

The *Net Mask* parameter contains the Netmask for the subnet the device is connected to. T This parameter is only effective, if bit 1 of register 321 is set. Otherwise, the values configured via DHCP or BOOTP are applied.

Gateway

The *Gateway* parameter stores the IP address of the default gateway. This parameter is only effective, if bit 2 of register 321 is set. Otherwise, the values configured via DHCP or BOOTP are applied. If no such values have been configured, it is assumed that there is no gateway present.

Ethernet Address

The *Ethernet Address* area can be used to overwrite the device's default Ethernet address (MAC address). This parameter is only effective, if bit 5 of register 321 is set.

Map FC1 and FC3 Flag

This parameter is an Open Modbus/TCP stack related parameter. It holds bit-oriented flags according to the following table:

Bits	Name (Bit mask)	Description
31 ... 2	Reserved	Reserved for future use
1	OMB_OMBTASK_CFG_FLAG_ TCPIP_NO_CONFIG	Configuration of TCP/IP stack: If set, the TCP/IP stack is **not** configured. Use this flag only for special cases! This means, in case that someone else configure the TCP/IP stack!
0	OMB_OMBTASK_CFG_FLAG_ FC1_FC3_OUTPUT	Alternative mapping in IO mode: If set, the Function codes FC1 and FC3 are mapped to the Output Data image of dual-port memory (abPd0Output[]).Needed e.g. for Clients without FC2, FC4 support. See also chapter 5.1.1 ‘Reading and Writing Data ‘

Table 25: Flags

4.7 POWERLINK Controlled Node

Register	Parameter	Meaning	Range of Value / Value
Interface			
300	Network Configuration Data Length	Total length (specified in number of bytes) of the following configuration data. If this value does not match the actual size of the configuration, the configuration will be rejected!	
301-302 Bit 0	Bus Startup	Communication start application controlled or automatic	0: Automatic 1: Application controlled (Default for NIC 50-RE/PLS)
301-302 Bits 1 and 2	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)	0: None, (1: 1 Byte, 2: 4 Byte) Default: 0: None
303-304	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
Ident			
305-306	Vendor ID	Identification number of the manufacturer	0x00000000 ... 0xFFFFFFFF (hex), Hilscher: 0x00000044
307-308	Product Code	Product code of the device as specified by the manufacturer	0x00000000 ... 0xFFFFFFFF (hex), Default NIC 50-RE/PLS: 0x00000008 (hex)
309-310	Revision Number	Revision number of the device as specified by the manufacturer	0x00000000 ... 0xFFFFFFFF (hex), Default NIC 50-RE/PLS: 0x00000000 (hex)
311-312	Serial Number	Serial number of the device	0x00000000 ... 0xFFFFFFFF (hex), Default NIC 50-RE/PLS: 0x00000000
Interface			
313-314	Stack Configuration Flags		
313 Bit 0	Disable Host-Triggered Input Data Exchange	Decides, whether host-triggered update for input data exchange is enabled (hook set) or disabled (hook not set).	0: not set 1: set Default: 0 (not set.)
313 Bit 1	Disable Host-Triggered Output Data Exchange	Decides, whether host-triggered update for output data exchange is enabled (hook set) or disabled (hook not set).	0: not set 1: set Default: 0 (not set.)

Register	Parameter	Meaning	Range of Value / Value
313 Bit 2	Configure Default Objects	Decides, whether a set of default objects are created within the object dictionary (=1, hook set) or not (=0, hook not set). If the objects will be created, the old set of previously existing objects will be cleared	0: Do not create standard objects 1: Create standard objects Default: 1 (Create standard objects)
313 Bit 3	Delete Application Objects	Decides, whether application objects are deleted (hook set) or not (hook not set). When 'Configure Default Objects' is set, then the firmware automatically does a 'Delete Application Objects'.	0: not set 1: set Default: 1 (set)
313 Bit 4	Disable PDO Mapping Version Check	Decides, whether PDO mapping version is set (hook set) or not set (hook not set).	0: not set 1: set Default: 1 (set)
315-316	Threshold Disable Flags		
315 Bit 0	Disable Loss SoC Threshold	Bit 0 of the threshold disable flags controls whether the Loss SoC error detection is to be disabled: 0 = Do not set Loss SoC error threshold to 0 1 = Set Loss SoC error threshold to 0. Register 317-318 will not be evaluated in that case.	0: not set 1: set Default: 0 (not set.)
315 Bit 1	Disable Loss PReq Threshold	Bit 1 of the threshold disable flags controls whether the Loss SoC error detection is to be disabled: 0 = Do not set Loss PReq error threshold to 0 1 = Set Loss PReq error threshold to 0. Register 319-320 will not be evaluated in that case.	0: not set 1: set Default: 0 (not set.)
315 Bit 2	Disable Loss SoA Threshold	Bit 2 of the threshold disable flags controls whether the Loss SoA error detection is to be disabled: 0 = Do not set Loss SoA error threshold to 0 1 = Set Loss SoA error threshold to 0. Register 321-322 will not be evaluated in that case.	0: not set 1: set Default: 0 (not set.)
315 Bit 3	Disable SoC Jitter Threshold	Bit 3 of the threshold disable flags controls whether the SoC Jitter error detection is to be disabled: 0 = Do not set SoC Jitter error threshold to 0 1 = Set SoC Jitter error threshold to 0. Register 323-324 will not be evaluated in that case.	0: not set 1: set Default: 1 (set)
315 Bit 4	Disable Collision Threshold	Bit 4 of the threshold disable flags controls whether the collision detection error threshold is to be disabled: 0 = Do not set collision error threshold to 0 1 = Set collision error threshold to 0. Register 325-326 will not be evaluated in that case.	0: not set 1: set Default: 0 (not set.)
315 Bit 5	Disable CRC Error Threshold	Bit 5 of the threshold disable flags controls whether the CRC error threshold is to be disabled: 0 = Do not set CRC error threshold to 0 1 = Set CRC error threshold to 0. Register 327-328 will not be evaluated in that case.	0: not set 1: set Default: 0 (not set.)
317-318	Loss SoC Threshold	specifies the Loss SoC Threshold if the threshold is enabled by register 315 , bit 0 = 0 (Disable Loss SoC Threshold) A value of 0 indicates to use the last known value or, if this does not exist, the internal default value. A value different from 0 indicates to use the specified value for the threshold.	0...2^{32}-1 Default: 15

For more see next page

319-320	Loss PReq Threshold	specifies the Loss PReq Threshold if the threshold is enabled b register 315 , bit 1 = 0y (Disable Loss PReq Threshold) A value of 0 indicates to use the last known value or, if this does not exist, the internal default value. A value different from 0 indicates to use the specified value for the threshold.	0...2^{32}-1 Default: 15
321-322	Loss SoA Threshold	specifies the Loss SoA Threshold if the threshold is enabled by register 315 , bit 2 = 0 (Disable Loss SoA Threshold) A value of 0 indicates to use the last known value or, if this does not exist, the internal default value. A value different from 0 indicates to use the specified value for the threshold.	0...2^{32}-1 Default: 15
323-324	SoC Jitter Threshold	specifies the SoC Jitter Threshold if the threshold is enabled by register 315 , bit 3 = 0 (Disable SoC Jitter Threshold) A value of 0 indicates to use the last known value or, if this does not exist, the internal default value. A value different from 0 indicates to use the specified value for the threshold.	0...2^{32}-1 Default: 0
325-326	Collision Threshold	specifies the Collision Threshold if the threshold is enabled by register 315 , bit 4 = 0 (Disable Collision Threshold) A value of 0 indicates to use the last known value or, if this does not exist, the internal default value of 15. A value different from 0 indicates to use the specified value for the threshold.	0...2^{32}-1 Default: 15
327-328	CRC Error Threshold	specifies the CRC Error Threshold if the threshold is enabled by register 315 , bit 5 = 0 (Disable CRC Error Threshold) A value of 0 indicates to use the last known value or, if this does not exist, the internal default value of 15. A value different from 0 indicates to use the specified value for the threshold.	0...2^{32}-1
329-330	Cycle Length	specifies the current cycle length in microseconds to be set. 0 = Ignore configuration parameter 0...2^{31}-1 = Change to new cycle length value	0...2^{31}-1 Default: 1000 (1 millisecond)
331-332	SoC Jitter Range	specifies the maximum allowed jitter in nanoseconds on two consecutive SoC frames: 0 = Ignore configuration parameter 0...2^{31}-1 = Change to SoC jitter range value	0...2^{31}-1 Default: 2000 (2000 nanoseconds)

For more see next page

Data			
333	Output Data Bytes	Length of the output data in byte	1… 1490 Byte, Default: 4 Byte
334	Input Data Bytes	Length of the input data in byte	1… 1490 Byte, Default: 4 Byte
Bus			
335-350	DNS Node Name	DNS-compatible name of the Powerlink Controlled Node / Slave (up to 32 characters)	
351-352	Gateway Address	Gateway address for IP stack The default gateway address is for NIC 50-RE/PLS: `192.168.100.254`	Valid address
353 Low byte	Node Id	EPL Node ID (EPL = Ethernet Powerlink)	1...239, Default NIC 50-RE/PLS: 1
353 High byte	SoC Trigger Config	Configuration of the SoC Trigger output 0,2 = Output disabled 1 = Output is low-active 3 = Output is high-active	Default: Output disabled
354-355	SoC Trigger Delay	Delay of SoC Trigger Impulse in units of 10 nanoseconds Choice of 0 disables SoC Trigger generation	0-10000000 (0-1 second)
356-357	SoC Trigger Length	Length of SoC Trigger Impulse in units of 10 nanoseconds This value must be smaller than the CycleLength.	100-10000000 (1microsecond –1 second)
358 Low byte	PReq Mapping Version	Mapping version of the PReq to be expected on receive	0...255 Default: 0
358 High byte	PRes Mapping Version	Mapping version of the PRes frame to be sent	0...255 Default: 0
359 Low byte	Number of Status Entries	Number of status entries to be kept in the status response	0...32 Default: 0
359 High byte	PReqErrorThreshold (lower 8 bits)	PReq Bus-Synchronous error threshold	$0...2^{32}-1$
360-361 Low byte	PReqErrorThreshold (upper 24 bits)	PReq Bus-Synchronous error threshold	$0...2^{32}-1$
361 High byte	PResErrorThreshold (lower 8 bits)	PRes Bus-Synchronous error threshold	$0...2^{32}-1$
362-363 Low byte	PResErrorThreshold (upper 24 bits)	PRes Bus-Synchronous error threshold	$0...2^{32}-1$
363 High byte	SyncFlagErrorThreshold (lower 8 bits)	Sync Flag error threshold	$0...2^{32}-1$
364-365 Low byte	SyncFlagErrorThreshold (upper 24 bits)	Sync Flag error threshold	$0...2^{32}-1$

Table 26: Powerlink Controlled Node / Slave Parameters

The 6 related threshold parameters

LossSoC Threshold

LossPReq Threshold

LossSoA Threshold

SoCJitter Threshold

Collision Threshold

CrcError Threshold

have a common logic which can be described as follows:

To each of these parameters there is an associated "*Set to zero*" bit:

Parameter	**Associated "*Set to zero*" bit**
LossSoC Threshold	Bit 0 of Register 315-316 (Threshold Disable Flags)
LossPReq Threshold	Bit 1 of Register 315-316 (Threshold Disable Flags)
LossSoA Threshold	Bit 2 of Register 315-316 (Threshold Disable Flags)
SoCJitter Threshold	Bit 3 of Register 315-316 (Threshold Disable Flags)
Collision Threshold	Bit 4 of Register 315-316 (Threshold Disable Flags)
CrcError Threshold	Bit 5 of Register 315-316 (Threshold Disable Flags)

For each of these parameters the consequence specified in the following table will happen depending on the value and the contents of the associated "*Set to zero*" bit:

"Set to zero" bit	**Value**	**Consequence**
0	0	Threshold remains unchanged
0	Value	Threshold is set to value
1	Does not matter	Threshold is set 0

The default value for LossSoC Threshold can be calculated according to the following formula:

*LossSoC = 1000 * CycleLength + LossOfFrameTolerance*

LossSoC is activated per default.

The parameters

- LossPReq Threshold
- LossSoA Threshold
- SoCJitter Threshold
- Collision Threshold
- CrcError Threshold

are deactivated per default.

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.

4.8 PROFIBUS DP Slave

Register	Parameter	Meaning	Range of Value / Value
Interface			
300	Network Configuration Data Length	Total length (specified in number of bytes) of the following configuration data. If this value does not match the actual size of the configuration, the configuration will be rejected!	
301-302 Bit 0	Bus Startup	Communication start application controlled or automatic	0: Automatic 1: Application controlled Default: 0 (Automatic)
303-304	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
301-302 Bits 1 and 2	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)	0: None, (1: 1 Byte, 2: 4 Byte) Default: 0 (None)
Ident			
305	Ident Number	PROFIBUS-system specific identification number	0x00000000 ... 0x0000FFFF (hex), Default NIC 50-DPS: 0x000000C10 (hex)
Bus			
306 Lowbyte	Station Address	PROFIBUS address of the device	0 … 126
306 Highbyte	Baudrate	Baud rate of PROFIBUS connection. See *Table 28: Available Baud Rate Values* below.	0:..11: See *Table 28* 15: Auto-Detect Default: 15: Auto-Detect
307 Bit 0	DPV1 Enable	WRMSTRT_FLG_DPV1_ENABLE This flag indicates whether DPV1 is supported, or not. If set, DPV1 functions are activated. Otherwise, DPV1 functions will not be available.	0: DPV1 disabled, 1: DPV1 enabled Default: 1
307 Bit 1	Sync supported	WRMSTRT_FLG_SYNC_SUPP This flag indicates if set to 1 that the slave stack shall support the SYNC command and the SYNC mode is activated. Otherwise, if set to FALSE(0), the slave stack will not support the SYNC command.	0: Sync not supported, 1: Sync supported Default: 1
307 Bit 2	Freeze supported	WRMSTRT_FLG_FREEZE_SUPP This flag indicates if set to TRUE(1) that the slave stack shall support the FREEZE command and the FREEZE mode is activated. Otherwise, if set to FALSE(0), the slave stack will not support the FREEZE command.	0: Freeze not supported, 1: Freeze supported Default: 1
307 Bit 3	‘Fail safe’ supported	WRMSTRT_FLG_FAILSAFE_SUPP This flag indicates whether ‘Fail safe’ operation is supported. If set, FAILSAFE mode is activated. Otherwise, FAILSAFE mode will not be available.	0: ‘Fail safe’ not supported, 1: ‘Fail safe’ supported Default: 1

307 Bit 4	Alarm SAP 50 deactivate	WRMSTRT_FLG_NO_ALARM_SUPP This flag indicates if set to TRUE (1) that the alarm SAP 50 is. If the flag is set to FALSE(0), the stack supports the alarm SAP 50.	0: SAP 50 supported, 1: SAP 50 deactivated Default: 1
307 Bit 5	I/O data swap	WRMSTRT_FLG_IO_SWAP This flag indicates if the I/O Data at the Dual Port Memory is shown at Motorola or Intel format.	0: No swap, 1: Swap Default: 1
307 Bit 6	Auto configuration	WRMSTRT_FLG_AUTOCONFIG This flag indicates if set to TRUE(1) that the PROFIBUS DP slave stack requests the host application for check configuration and user parameter data. If set to FALSE(0), the stack handles configuration and parameter data.	0: Auto-config off, 1: Auto-config on Default: 1
307 Bit 7	Address change not allowed	WRMSTRT_FLG_NO_ADDR_CHANGE This flag indicates if set to TRUE(1) that the slave stack does not support the “Set Slave Address” command. If set to FALSE(0), changing the bus address via the master is activated and the slave stack does support the “Set Slave Address” command	0: “Set Slave Address” supported, 1: “Set Slave Address” not supported Default: 1
Data			
308 (Highbyte)	Length of Configuration Data	Number of bytes following containing configuration data	2 … 244
309..430 (each) 309..430 (each)	Each of the registers 309 to 430 may contain a parameter block for slave configuration data. Such a parameter block consists of an identifier byte (in the low byte, may be specified either in the general or in the special identifier format) and a length byte (in the high byte) The length of the complete check configuration data block must not exceed 244 bytes, however, it is recommended not to exceed an amount of 32 bytes otherwise some restrictions apply, see the IEC 61158 or EN 50170 specification.		
	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.	

Table 27: Parameters - PROFIBUS-DP Slave

The applicable baud rates (for register 306 high byte) can be coded with the values given in the following table:

Baud rate	Symbolic Constant	Value
Baud rate 9,6 kBit/s	PROFIBUS_DL_DATA_RATE_96	0
Baud rate 19,2 kBit/s	PROFIBUS_DL_DATA_RATE_19_2	1
Baud rate 93,75 kBit/s	PROFIBUS_DL_DATA_RATE_93_75	2
Baud rate 187,5 kBit/s	PROFIBUS_DL_DATA_RATE_187_5	3
Baud rate 500 kBit/s	PROFIBUS_DL_DATA_RATE_500	4
Baud rate 1,5 MBit/s	PROFIBUS_DL_DATA_RATE_1500	6
Baud rate 3 MBit/s	PROFIBUS_DL_DATA_RATE_3000	7
Baud rate 6 MBit/s	PROFIBUS_DL_DATA_RATE_6000	8
Baud rate 12 MBit/s	PROFIBUS_DL_DATA_RATE_12000	9
Baud rate 31,25 kBit/s	PROFIBUS_DL_DATA_RATE_31_25	10
Baud rate 45,45 kBit/s	PROFIBUS_DL_DATA_RATE_45_45	11
Baud rate Auto detect	PROFIBUS_DL_DATA_RATE_AUTO	15

Table 28: Available Baud Rate Values

For registers 309 to 430:

Identifier Byte for the General Format

The identifier byte can be specified in the general (also called compact) format or the special format. In the general format, the meaning of the single bits is defined as follows:

General Identifier Format of Identifier Byte (according to IEC 61158/EN 50170 Specification)				
Bit	**Name**	**Value**	**Description**	
D7	Consistency	Consistency extends over		
		0	Byte or word	
		1	Whole length	
D6	Length format	**Length format**		
		0	Byte structure	
		1	Word structure	
D5.. D4	Signification	Input/Output/Special identifier format		
		D5	D4	
		0	0	This combination signifies the special identifier format, see below.
		0	1	Input
		1	0	Output
		1	1	Input- Output
D3D0	Length of data	0	1 Byte/Word	
		1	2 Bytes/Words	
		…		
		15	16 Bytes/Words	
		(choice of byte or word depends on length format bit)		

Table 29: General Identifier Format of Identifier Byte according to IEC 61158/EN 50170 Specification

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

Consistency in this context means whether the whole data need to be interpreted as an entity or each byte/word may separately be interpreted by the Profibus DP Master.

Note: When transferring data in word mode, the high byte is transferred first by Profibus DP, then the low-byte. However, the Profibus DP Master has the possibility to swap this sequence of the bytes within the word, if required by the target system.

Identifier Byte for the Special Format

To allow extended configurations and to increase flexibility, a special extension of the identifier system described above is also supported by Profibus DP. The main advantages of this format are:

- It is possible to determine the number of input and output bytes associated to the defined identifier.
- User specific data can be added.

This format is called the special identifier format and signified by the combination of bit 4 and 5 both being zero as already described above in the discussion of the general identifier format.

Special Identifier Format of Identifier Byte (according to IEC 61158/EN 50170 Specification)				
Bit	**Name**	**Value**	**Description**	
D7..D6	Consistency/ Length format (used for Input/Output)	Input/Output		
		D7	D6	
		0	0	Free area
		0	1	1 length byte for inputs follows
		1	0	1 length byte for outputs follows
		1	1	1 length byte for outputs and 1 length byte for inputs follows
D5.. D4	Signification	Signification of Special identifier format		
		D5	D4	(No other combinations allowed in Special Identifier Format)
		0	0	This combination signifies the special identifier format, see below.
D3	Data Length	Length of manufacturer specific data		
		0	No manufacturer specific data follow; no data in Real_Cfg_Data.	
		1-14	Manufacturer specific data of the length specified in the following byte(s) follow, these should be equal to those in Real_Cfg_Data.	
		15	In case of Check_Cfg: No manufacturer specific data follow, verification may be omitted.	

Table 30: Special Identifier Format of Identifier Byte according to IEC 61158/EN 50170 Specification

Length Byte

The length bytes following the special identifier format bytes are organized as described in the table below:

Structure of Length Byte in the Special Identifier Format of the Identifier Byte according to IEC 61158/EN 50170 Specification			
Bit	**Name**	**Value**	**Description**
D7	Consistency	Consistency extends over	
		0	Byte or word
		1	Whole length
D6	Length format	Length format	
		0	Byte structure
		1	Word structure
D5.. D0	Length of data	0	1 Byte/Word
		1	2 Bytes/Words
		…	
		63	64 Bytes/Words

Table 31: Structure of Length Byte in the Special Identifier Format of the Identifier Byte according to IEC 61158/EN 50170 Specification

General note:

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).

For more information refer to the PROFIBUS-DP Slave Protocol API Manual.

4.9 PROFINET IO Device

For PROFINET IO Device, you should first check, whether or not configuration via Modbus makes sense in your case of application due to the following reasons:

1. The configuration for PROFINET IO Device via Modbus is more sophisticated than the ones for other communication systems.
2. PROFINET IO Device requires additional configuration information which cannot be supplied via Modbus.
3. Large configurations exceed the amount of available registers in the virtual dual-port memory of the netIC and cannot be managed at all. For details of this limitation read below.
4. If little-endian byte order is used, an additional optional structure (`tSignalConfig`) is required even more enlarging the complexity and reducing the total number of submodules available. For detail, see PROFINET IO Device RT IRT Protocol API Manual.

Important limitation: At maximum, there are only 688 registers available for configuration purposes. Of these, at maximum 392 registers are available to be filled with API and submodule structures.

Configurations with a large number of submodules (>19) may exceed this limit. In this case, configuration via Modbus is not possible.

If big-endian byte order is used, the following formula allows to calculate the total number of submodules available in dependence from the number of APIs:

$$n_{SM} = (392-4*n_{API})/20$$

where n_{SM} represents the total number of submodules and n_{API} represents the number of APIs

Omit the fractional part of the result, then you will get the number of total number of submodules available.

Consequently, this excludes:

- Using more than 16 APIs within the entire configuration (even if each of these would only consist of one single submodule).
- Using more than 19 submodules within a single API.
- Using more than totally 16 to 19 submodules within the entire configuration (the exact value depends on the result of the formula above). Anyway, this is by far lower than the amount of submodules that the stack can handle (1000 submodules) and may be considered as a severe restriction.

The following *Table 32: PROFINET IO-Device Parameters V3* describes the configuration parameters to apply for PROFINET IO Device

Register	Parameter	Meaning	Value Range
Interface			
300	Network Configuration Data Length	Total length (specified in number of bytes) of the following configuration data. If this value does not match the actual size of the configuration, the configuration will be rejected!	
301-302 Bit 0	Bus Startup	Communication start application controlled or automatic	0: Automatic 1: Application controlled Default for NIC 50-RE/PNS: 1: Application controlled
301-302 Bits 1 and 2	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)	0: None, (1: 1 Byte, 2: 4 Byte) Default: 0: None
303--304	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			
305-306	Vendor ID	Identification number of the manufacturer, assigned by PROFIBUS Nutzerorganisation e. V. All Hilscher products use the value 286 (0x011E).	0x0000 ... 0xFEFF (hex), Hilscher: 0x011E (hex)
307-308	Device ID	Identification number of the device, freely eligible by the manufacturer, fixed for every device. NIC 50-RE/PNS uses the value 266 (0x0010A).	0x00000000 ... 0x0000FFFF (hex), NIC 50-RE/PNS: 0x00000010A (hex)
309-310	MaxAR	Currently not used. Set to 0.	0
Data			
311-312	Input bytes (Complete Input Size)	Maximum amount of allowed input data. The sum of data of all submodules configured by the user must not exceed this value. This field references input data as data received by the IO-Device.	Default value:128 Allowed values: 0.. 1024 Bytes
313-314	Output bytes (Complete Output Size)	Maximum amount of allowed output data. The sum of data of all submodules configured by the user must not exceed this value. This field references output data as data sent by the IO-Device.	Default value:128 Allowed values: 0.. 1024 Bytes
Ident			
Length: 315-316 Data 317-436	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.	Character string, 1 ... 240 characters, Default: See GSDML-File
Length: 437-438 Data 439-558	Type of Station	Type name of the PROFINET station; name can be assigned freely.	Character string, 1 ... 240 characters, Default: See GSDML-File
559-572	Device Type	Description of the device type, freely eligible	Character string, 0 ... 25 characters

For more see next page

<table>
<tr><td>573-582</td><td>Order ID</td><td>Hilscher device number (e. g. 1541 000) or order description of the customer for its device</td><td>Character string,
0 ... 20 characters</td></tr>
<tr><td></td><td>Device-Access-Point (DAP)</td><td>The Device Access Point (DAP) must be selected according to the firmware version. According to the version of the PROFINET IO Slave device firmware the Device Access Point sets, which properties the device has. The device access point (DAP) is a module of the GSDML to describe the device parameters device specific.
The following table shows what DAP must be set at which firmware version.
NIC 50-RE/PNS:
<table>
<tr><th>Firmware Version</th><th>DAP</th><th>Supported Features</th></tr>
<tr><td>1.2.11 or older</td><td>1</td><td>RT Communication</td></tr>
<tr><td>1.2.12 - 1.4.3</td><td>2</td><td>FastStartup,
RT Communication,
IRT Communication</td></tr>
<tr><td>1.4.4 or newer</td><td>3</td><td>FastStartup,
Identification & Maintenance 1-4,
RT Communication,
IRT Communication</td></tr>
</table></td><td>1, 2, 3</td></tr>
<tr><td colspan="4">Bus</td></tr>
<tr><td>583-584</td><td>IP Address</td><td>Valid IP address for the device</td><td>Valid IP address
Default: 0.0.0.0</td></tr>
<tr><td>585-586</td><td>NetMask</td><td>Valid network mask for the device</td><td>Valid network mask
Default: 0.0.0.0</td></tr>
<tr><td>587-588</td><td>Gateway Address</td><td>Valid gateway address for the device</td><td>Valid gateway address
Default: 0.0.0.0</td></tr>
<tr><td colspan="4">Ident</td></tr>
<tr><td>589</td><td>Hardware Revision</td><td>Hardware Revision</td><td>0..0xFFFF
Default: 0</td></tr>
<tr><td>590</td><td>Software Revision 1</td><td>Software Revision, Part 1</td><td>0..0xFFFF
Default: 0</td></tr>
<tr><td>591</td><td>Software Revision 2</td><td>Software Revision, Part 2</td><td>0..0xFFFF
Default: 0</td></tr>
<tr><td>592</td><td>Software Revision 3</td><td>Software Revision, Part 3</td><td>0..0xFFFF
Default: 0</td></tr>
<tr><td>593
High byte</td><td>Prefix/Reserved</td><td>Software Revision Prefix
Possible values and their meanings are::
‘V’: Edited Version
‘R’: Revision
‘P’: Prototype
‘U’: In field test
‘T’: Test device</td><td>‘V’, ‘R’, ‘P’, ‘U’, ‘T’</td></tr>
<tr><td>594</td><td>Maximum Diag Records</td><td>The number of diagnosis records the stack shall be able to handle in parallel.</td><td>1..0xFFFF
Default: 256</td></tr>
<tr><td>595</td><td>InstanceID</td><td>Instance ID. This parameter must match with <code>ObjectUUID_LocalIndex</code> in the GSDML file belonging to the IO-Device.</td><td>0..0xFFFF
Default: 1</td></tr>
<tr><td>596</td><td>Reserved</td><td>Set to zero</td><td>0</td></tr>
</table>

For more see next page

Data			
597-598	NumAPI	Number of APIs	>=1
599-600	API profile	Number of the API profile to be configured. Currently only the value 0 indicating "manufacturer specific " is supported!	0
601-602	NumSubmoduleItems	Number of following submodules contained in this API	1..n
603	Slot	The slot this submodule belongs to.	
604	Subslot	The subslot this submodule belongs to.	
605-606	Module ID	The ModuleID of the module this submodule belongs to.	
607-608	Submodule ID	The SubmoduleID of this submodule.	
609-610	ProvDataLen	Provider Data Length The length of data provided by this submodule. This length describes the data sent by IO-Device and received by IO-Controller.	0..1024
611-612	ConsDataLen	Consumer Data Length The length of data consumed by this submodule. This length describes the data sent by IO-Controller and received by IO-Device.	0..1024
613-614	DPM Offset In	Offset in DPM InputArea where consumed data for the submodule shall be copied to. This data is received by IO-Device and sent by IO-Controller. If the length of data in this direction is 0 or if DPM is not used this value shall be set to `0xFFFFFFFF`.	
615-616	DPM Offset Out	Offset in DPM OutputArea where provided data of the submodule shall be taken from. This data is sent by IO-Device and received by IO-Controller. If the length of data in this direction is 0 or if DPM is not used this value shall be set to `0xFFFFFFFF`.	
617	OffsetIOPSProvider	Offset IOPS Provider Offset where to put IOPS provider state for this submodule relative to beginning of IOPS block in DPM output area too	
618	OffsetIOPSConsumer	Offset IOPS Consumer Offset where to take IOPS provider state of this submodule relative to beginning of IOPS block in DPM input area from.	
619-622	Reserved	Reserved for future use. Set to zero.	
623…	Further submodule structure(s) **`PNS_IF_SUBMODULE_STRUCT_T`** (in case of `NumSubmoduleItems>1`**)** or API structure **`PNS_IF_API_STRUCT_T`** (in case of `NumSubmoduleItems=1`**)** follow, see below!		

Table 32: PROFINET IO-Device Parameters V3

Here only the beginning of the data area can be outlined. In any case, the data area begins with an API structure (`PNS_IF_API_STRUCT_T`). This is followed by a number of submodule structures (`PNS_IF_SUBMODULE_STRUCT_T`). The exact number of `PNS_IF_SUBMODULE_STRUCT_T` submodule structures can be taken from registers 601-602 for the first API.

If more than one API is configured, then again an API (`PNS_IF_API_STRUCT_T`) will follow and then `NumSubmoduleItems` times a submodule structure (`PNS_IF_SUBMODULE_STRUCT_T`) will follow for each additional API.

However, for any further API you have to calculate the register numbers to be used on your own. The following rules apply for the amount of registers required and the number of submodule structures in an API:

- Any `PNS_IF_API_STRUCT_T` requires 4 registers.
- Any `PNS_IF_SUBMODULE_STRUCT_T` requires 20 registers:
- There are as many submodule structures as specified in the APIs `NumSubmoduleItems` variable (stored as 32 bit value in registers 3 and 4).

For more details concerning the configuration mechanism see the PROFIBUS IO Device Protocol API Manual, section "Set Configuration Request".

Note: To configure the Controller, a GSDML file (device description file) is required. The settings in the used PROFINET IO 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 PROFINET IO Controller of this device. If no name chosen freely is used in the configuration file, then the name from the GSDML file is used.

4.10 VARAN Client

Register	Parameter	Meaning	Range of Value/Value
Interface			
300	Network Configuration Data Length	Total length (specified in number of bytes) of the following configuration data. If this value does not match the actual size of the configuration, the configuration will be rejected!	
301-302 Bit 0	Bus Startup	This parameter is represented by bit 0 of the system flags. The start of the device can be performed either application controlled or automatically: Automatic (0): Network connections are opened automatically without taking care of the state of the host application. Communication with a controller after a device start is allowed without BUS_ON flag, but the communication will be interrupted if the BUS_ON flag changes state to 0 Application controlled (1): The channel firmware is forced to wait for the host application to wait for the Application Ready flag in the communication change of state register (see section 3.2.5.1 of the netX DPM Interface Manual). Communication with controller is allowed only with the BUS_ON flag. For more information concerning this topic see section “Controlled or Automatic Start” of the netX DPM Interface Manual.	0: Automatic 1: Application controlled
303-304	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 = 100 (0x64), 0 = Off
Begin of VARAN-specific parameters			
305-306	Application mode	Application mode `0 = VARAN_CLIENT_APP_MODE_IO`	0
307-316	Reserved bytes	Set to 0	0
317-318	Vendor Identifier	This value is specific for every single vendor.	Allowed values: 0 … 0xFFFFFFFF Default value: 0x1D
319-320	Device Identifier	This value is specific for every single device type. In case of an unknown device Id the manager rejects the client	Allowed values: 0 … 0xFFFFFFFF Default value: 0x3FD
321-322	License Number	The license number of the device	Allowed values: 0 … 0xFFFFFFFF Default value: 0x0

For more see next page

323-324	Product Revision	The product revision number of the device	Allowed values: 0 … 0xFFFFFFFF Default value: 0x0
325-356	VendorName	This is a 64-byte ASCII array, which should contain the vendor name	Allowed values: ASCII characters range Default value: “Hilscher GmbH”
357-388	DeviceName	This is a 64-byte ASCII array, which should contain the device name	Allowed values: ASCII characters range Default value: “netX”
389-390	Serial Number	The serial number of the device	Allowed values: 0 … 0xFFFFFFFF Default value: 0
391-392	Order Number	The order number of the device	Allowed values: 0 … 0xFFFFFFFF Default value: 0
393-396	Reserved	Reserved	0
397-398	MemArea1ReadOffset	Reading offset of memory area 1	Allowed values: 0 … 0xFFFF Default value: 0x2000
399-400	MemArea1ReadSize	Reading size of memory area 1	Allowed values: 0 … 128 Default value: 128
401-402	MemArea1WriteOffset	Writing offset of memory area 1	Allowed values: 0 … 0xFFFF Default value: 0x2000
403-404	MemArea1WriteSize	Writing size of memory area 1	Allowed values: 0 … 128 Default value: 128
405-406	MemArea2ReadOffset (not supported, set to default value)	Reading offset of memory area 2	Allowed values: 0… 0xFFFF Default value: 0xFFFF
407-408	MemArea2ReadSize (not supported, set to default value)	Reading size of memory area 2	Allowed values: 0… 128 Default value: 0

For more see next page

409-410	MemArea2WriteOffset (not supported, set to default value)	Writing offset of memory area 2	Allowed values: 0 … 0xFFFF Default value: 0xFFFF
411-412	MemArea2WriteSize (not supported, set to default value)	Writing size of memory area 2	Allowed values: 0 … 128 Default value: 0
413-414	ConfigFlags (not supported, set to default value)	HAL configuration flags Bit 0: Enables EMAC (not supported yet) Bit 1: Enable Memory Area 2 (not supported yet)	Reserved bits: 0 … 0xFFFFFFFC Default value: 0
415-416	ClientWdgTime	Client watchdog time in ms. This parameter concerns the communication over the VARAN bus. (0 indicates watchdog off)	Allowed values: 0 … 130 0 – Disables watchdog triggering Default value: 130
417-432	Reserved	Reserved	
433-434	SyncOutPulsLen	Length of the sync out pulse in 10ns steps. (e.g. value 100 results as 10ns*100=1000ns=1µs pulse)	Allowed values: 10 … n Default value: 100
435-436	SyncOut0Mode *)	This parameter controls whether SYNC OUT 0 ID 0 – This output is not available 3 – Time for data IN valid 4 – Time for data OUT valid 5 – Time for data IN/OUT valid See note below	Allowed values: 0,3,4,5 Default value: 5
437-438	SyncOut0Flags	Sync out 0 flags: Bit 0: Enable/Disable output Bit 1: Polarity active high/low	Bit 0: 0 = Output disable Bit 0: 1 = Output enable Bit 1: 0 = active low Bit 1: 1 = active high Bit 2..31 reserved, set to 0 Default value: 0x00000001
439-440	SyncOut1Mode *)	This parameter controls whether SYNC OUT 1 ID 0 – This output is not available 3 – Time for data IN valid 4 – Time for data OUT valid 5 – Time for data IN/OUT valid See note below	Allowed values: 0,3,4,5 Default value: 0

441-442	SyncOut1Flags	Sync out 1 flags: Bit 0: Enable/Disable output Bit 1: Polarity active high/low	Bit 0: 0 = Output disable Bit 0: 1 = Output enable Bit 1: 0 = active low Bit 1: 1 = active high Bit 2..31 reserved, set to 0 Default value: 0x00000000

Table 33: VARAN Client Parameters

Note: The parameters besides the first two ones are VARAN specific. Some of them are stored into the RAM as an image, emulating the presence of a SPI memory. After a successful initialization, the Manager tries to read/write/erase this SPI image by sending specific VARAN commands. At this stage read only commands are processed.

The usage of memory area 2 is not implemented yet in the firmware layer.

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)

5 Lists

5.1 List of Figures

5.2 List of Tables

6 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.

BootP

Boot Protocol

Boot Protocol

A protocol for automatically assigning the IP address for a TCP/IP connection.

CANopen

CANopen is a high level communication protocol for the CAN Fieldbus system which is a serial bus system allowing multi master and real-time operation for networking with high-reliability. It works with twisted pair wires and is standardized by ISO 11898.

CAN and CANopen are especially used in the automotive industry and have been developed by Robert Bosch GmbH. The user organization supporting CANopen is CAN in Automation (CiA, http://www.can-cia.org/).

CC-Link

Control and Communication Link

A Fieldbus system for networking with transmission rates up to 10 Mbps which has been developed by Mitsubishi Electric Corporation and which is mainly used in Asia. The user organization supporting CC-Link is CC-Link Partner Association (CLPA, http://www.clpa.org/).

Coil

A coil (in the meaning defined by Modbus terminology) is a single bit in memory that can be accessed (i.e. read or write) via Modbus.

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).

DDF

Device_Description_File.

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, GSD files, GSDML files or XDD files.

DeviceNet

DeviceNet is a Fieldbus based on an own cabling system with two twisted pairs of wires with shielding, one for communication and one for 24 V power supply (maximum allowed current 8 A).

It is promoted by the Open DeviceNet Vendors Association (ODVA, http://www.odva.org/) and commonly used in the USA and in Asia.

DHCP

See Dynamic Host Configuration Protocol

Dynamic Host Configuration Protocol

A protocol for automatically assigning the IP address for a TCP/IP connection.

EDS file

A special kind of Device Description File used by CANopen, DeviceNet und EtherNet/IP.

EtherCAT

Ethernet Control Automation Technology

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 Powerlink

A communication system for industrial Ethernet designed and developed in Austria by B&R, Bernecker + Rainer Industrie-Elektronik Ges.m.b.H.

It partly uses CANopen technologies.

Flash Memory

A non-volatile memory circuit allowing to store data even if the power supply to the memory circuit is switched off during storage of data.

Function code

A function code (in the meaning defined by Modbus terminology) is a standardized method to access (i.e. read or write) coils or registers via Modbus.

Gateway

A device interfacing between two different communication standards.

Gateway Address

The IP Address of a gateway

GSD file

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

GSDML file

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

Industrial Ethernet

See Real-Time Ethernet

IP Address

A numeric value used for uniquely identifying a participant within the Internet. An IP address is set up of four numbers each in the range between 0 and 255 which are separated by points, for instance 192.168.2.100 would be an IP address.

An IP address can be static or can dynamically be assigned by special protocols such as BootP and DHCP.

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.

Modbus RTU

A standard for serial communication developed by Schneider Automation that is used for communication of the host with the NIC 50-RE. It uses the Modbus Data Model.

netX

networX on chip, next generation of communication controllers.

netX Configuration Tool

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

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.

Powerlink

See Ethernet Powerlink

PROFIBUS-DP

Process Field Bus – Decentralized Periphery

A Fieldbus system for connecting analog and digital inputs and outputs with intelligent controls which is used mainly in factory automation. It follows the standards in IEC 61158 and supports transmission speed up to 12 Mbps.

It has been developed by Siemens AG. The user organization supporting PROFIBUS is PROFIBUS & PROFINET International.

PROFINET-IO

A communication system for Industrial Ethernet designed and developed by PROFIBUS & PROFINET International under large participation of Siemens AG. It uses some mechanisms similar to those of the PROFIBUS field bus.

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
- Open Modbus/TCP
- Powerlink
- PROFINET-IO
- SERCOS III

Register

A register (in the meaning defined by Modbus terminology) 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.

SERCOS III

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

VARAN

Versatile Automation Random Access Network

A communication system for industrial Ethernet designed and developed by SIGMATEK.

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 a standard maintained by the W3C (World-wide web consortium). Device Description Files often use XML-based formats for storing the device-related data appropriately.

7 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 - 110 025
Phone: +91 11 40515640
E-Mail: info@hilscher.in

Italy
Hilscher Italia srl
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.
Suwon, 443-734
Phone: +82 (0) 31-695-5515
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