

FNL Modbus TCP Interface

Users Manual

V0.1

17.06.2009

Project No.: 5304
Doc-ID.: FNL Modbus TCP Interface-UM-V0.1
Status: Released

Revision History

Version	Date	Description
V0.1	17.06.2009	Initial Version

COMSOFT GmbH
Wachhausstraße 5a
76227 Karlsruhe, Germany
Phone +49 721 9497 - 0
Fax +49 721 9497 - 129

Copyright © 2009 by COMSOFT GmbH

This document is protected by copyright. Reproduction, duplication, publishing, transfer or disclosure of the contents of this document are only permitted after prior written agreement has been obtained from COMSOFT GmbH.

Contents

1	Introduction	1
2	Installation	1
2.1	Installation of the Software	1
2.2	FNL - Ethernet configuration	2
2.2.1	Adjustment of the FNL TCP/IP-address	3
2.2.2	Adjustment of the FNL operation mode	4
2.2.3	Modbus TCP Server	6
2.2.3.1	Supported functions	6
2.2.3.2	Modbus TCP Server address mapping	7
2.3	Operation as PROFIBUS DP Master	9
2.3.1	Configuration as PROFIBUS DP Master	10
2.3.2	PROFIBUS DP configuration download	11
2.3.3	Monitor/Modify mode	13
2.3.4	Modbus TCP Server with DP Master Mode	14
2.3.4.1	Configuration of the PROFIBUS DP Master by control register1 (8193)	14
2.3.4.2	Mapping of the DP Input/Output data	15
2.3.4.3	Indication of changing Input data	18
2.3.4.4	Watchdog Timer	19
2.3.4.5	PROFIBUS DP Life List and Status Information	20
2.3.4.6	Status codes	20
2.3.4.7	DP Slave Summary State	21
2.4	Operation as PROFIBUS DP Slave	22
2.4.1	Configuration of FNL as DP Slave	22
2.4.1.1	Configuration of the DP Master	22
2.4.1.2	Configuration of FNL AUTO SLAVE Mode	22
2.4.2	Modbus TCP Server as DP Slave	24
2.4.2.1	Status of the FNL DP Slave in control register2 (8194)	24
2.4.2.2	Mapping of the FNL DP Slave Input/Output data	26
2.4.2.3	Connection monitoring	28
2.4.2.3.1	PROFIBUS DP communication	28
2.4.2.3.2	Start up and Monitoring of the Modbus TCP Client connection	28

List of Figures

Figure 1: COMSOFT Network Configurator	2
Figure 2: Adjustment of the TCP/IP-address.....	3
Figure 3: Adjustment of the FNL operation mode	4
Figure 4: FNL with changed parameters.....	5
Figure 5: FNL – Operation as PROFIBUS DP Master.....	9
Figure 6: PROFIBUS configuration with CONFIGURATOR II	10
Figure 7: Driver selection.....	11
Figure 8: Installed FNL devices	11
Figure 9: Download symbol	12
Figure 10: Download of PROFIBUS configuration	12
Figure 11: Monitor/Modify symbol.....	13
Figure 12: Monitor/Modify mode	13
Figure 13: Format of control register1.....	14
Figure 14: FNL – Operation as PROFIBUS DP Slave/Modbus TCP Server.....	22
Figure 15: Format control register2.....	24

List of Tables

Table 1: Supported Modbus TCP Server functions.....	6
Table 2: Modbus TCP adress mapping	8
Table 3: Control register1	14
Table 4: Example PROFIBUS_to_MODBUS Buffer.....	15
Table 5: Example MODBUS_to_PROFIBUS Buffer.....	17
Table 6: Life list with DP Slave status.....	20
Table 7: Status codes DP Slave	20
Table 8: Life list and DP Slave Status.....	21
Table 9: Status bits DP Slave Mode	25
Table 10: FNL DP Slave status codes	25
Table 11: Input data FNL DP Slave	26
Table 12: Output data FNL DP Slave	27

This page is intentionally left blank !

1 Introduction

FNL is a Ethernet based PROFIBUS DP Gateway, which can be operated as DP Master or DP Slave. FNL is implemented as MODBUS TCP Server according to MODBUS TCP protocol standard V1.1 and supports a subset of MODBUS TCP compatible functions to access input- and output buffers.

Due to the Ethernet based MODBUS TCP interface FNL can communicate with every common MODBUS TCP client system independent of the used hardware platform and operating system. Only for configuration purposes a windows based PC is necessary.

2 Installation

Install FNL according to the Installation Instructions (document FNL_e.pdf), by connecting FNL to a power supply (24 Volts DC) as well as to an Ethernet network. The PC used for the FNL configuration must be connected to the same Ethernet network as well. A direct connection between the PC and FNL via an Ethernet cross cable is also possible.

2.1 Installation of the Software

The delivered software on the CD needs to be installed on a Windows-based PC. Start the setup program and follow the instructions to install the software. The setup program copies all necessary components to the PC.

2.2 FNL - Ethernet configuration

Start the program "COMSOFT Network Configurator" from the Windows Start Menu "*Comsoft GmbH/Network Configurator*".

The COMSOFT Network Configurator scans automatically the network and displays all detected FNL devices:

- For scanning the network again, click the "search"-symbol on top-left.

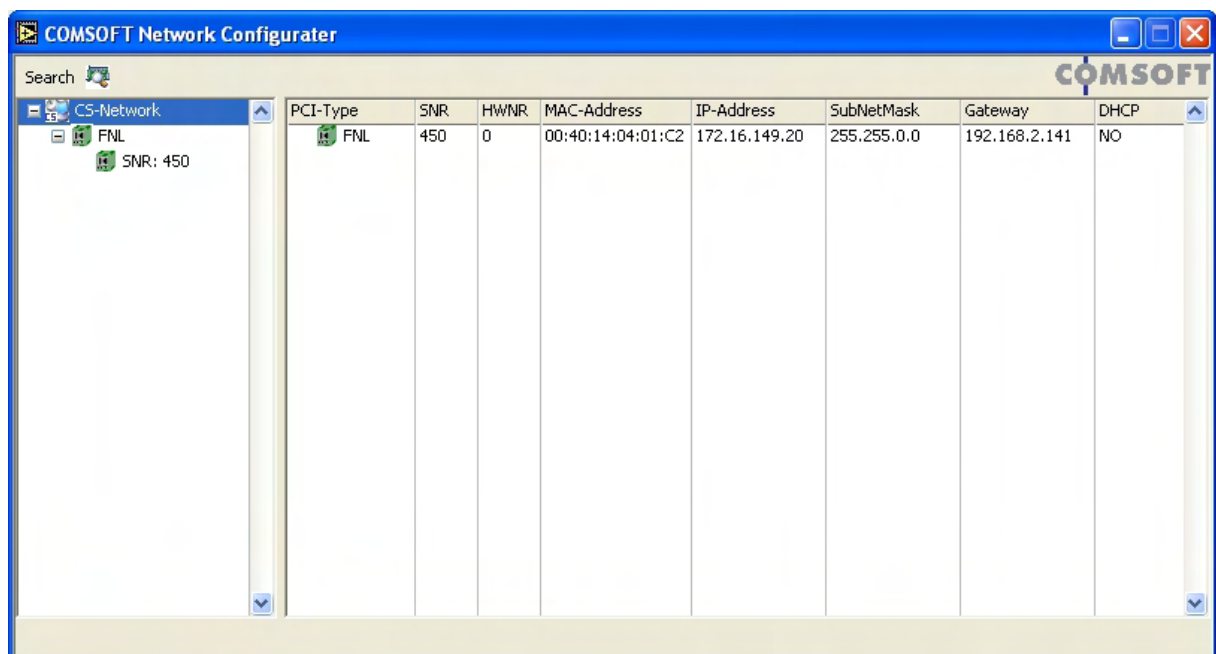


Figure 1: COMSOFT Network Configurator

2.2.1 Adjustment of the FNL TCP/IP-address

- Double click the FNL to be configured and adjust the **IP address**, **Subnet mask** and **Gateway**. Using DHCP to obtain the IP address automatically is possible as well.

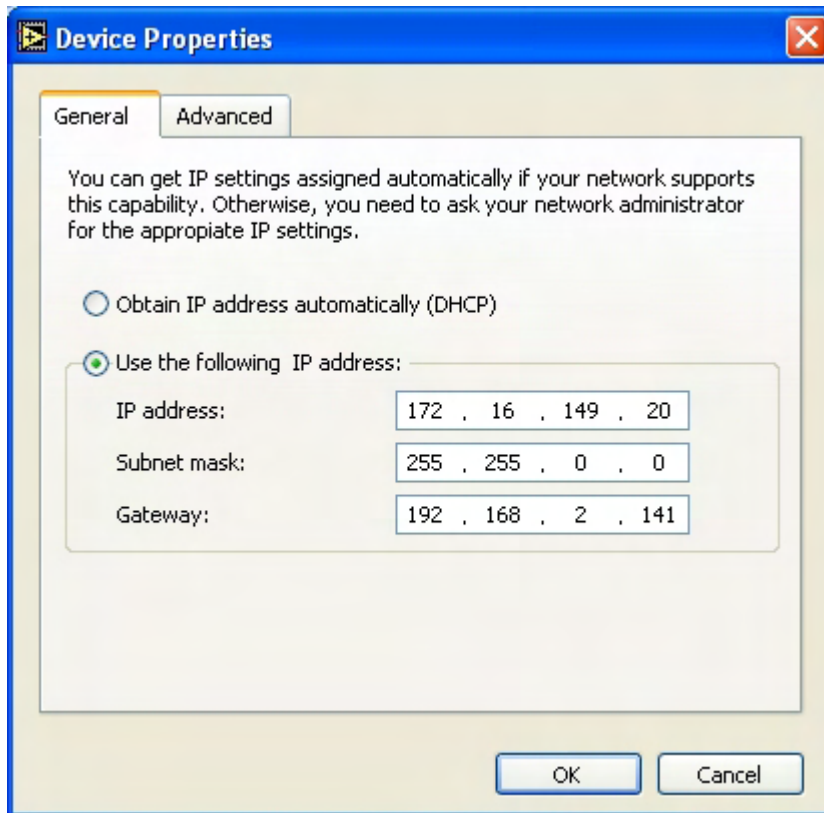


Figure 2: Adjustment of the TCP/IP-address

2.2.2 Adjustment of the FNL operation mode

FNL supports 2 operation modes to guarantee the downwards compatibility to existing applications.

- Select the tab **Advanced** and adjust the parameter **FNL-Mode** to "COMSOFT interface":

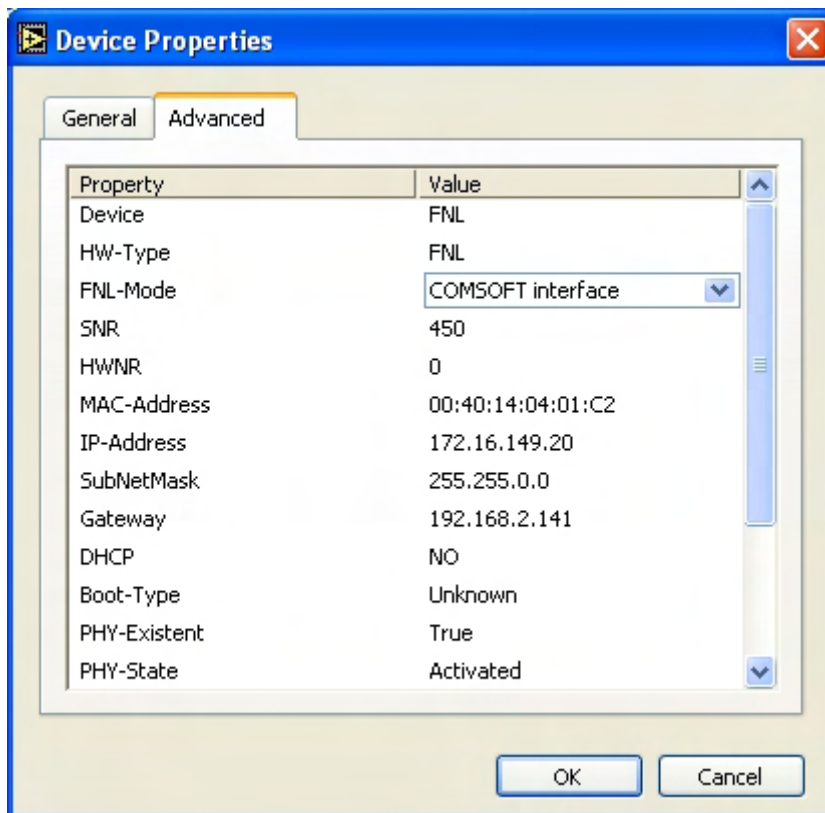
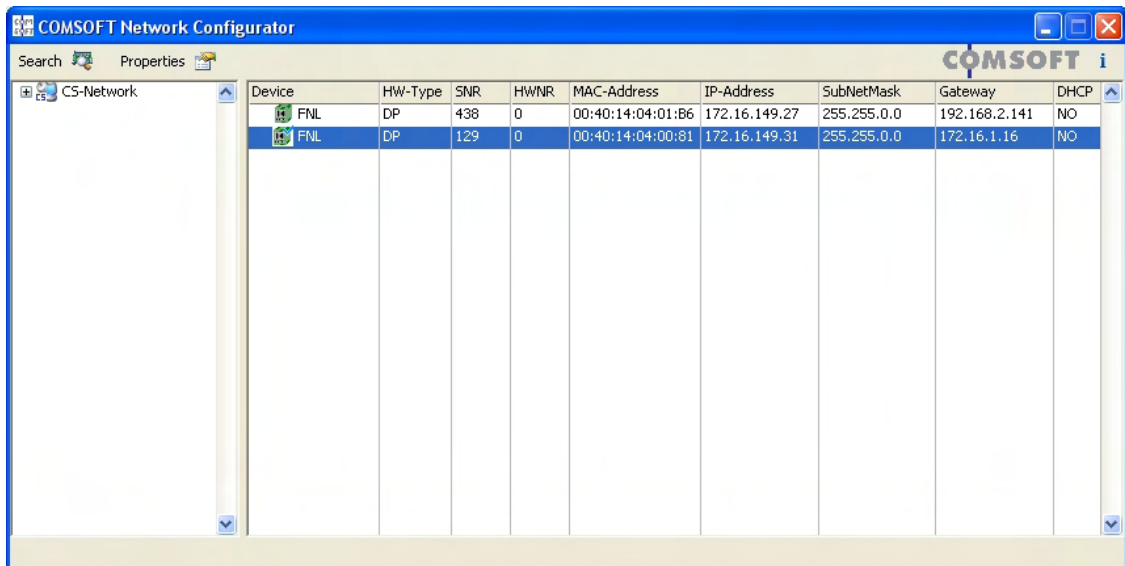


Figure 3: Adjustment of the FNL operation mode

- Click the **OK** button to close the "device properties" dialogue box. "Network Configurator" activates the changed parameters on the related FNL device and displays it:



The screenshot shows the COMSOFT Network Configurator interface. On the left, a tree view shows 'CS-Network'. The main area displays a table with the following columns: Device, HW-Type, SNR, HWNR, MAC-Address, IP-Address, SubNetMask, Gateway, and DHCP. Two rows of FNL devices are listed, with the second row highlighted in blue.

Device	HW-Type	SNR	HWNR	MAC-Address	IP-Address	SubNetMask	Gateway	DHCP
FNL	DP	438	0	00:40:14:04:01:B6	172.16.149.27	255.255.0.0	192.168.2.141	NO
FNL	DP	129	0	00:40:14:04:00:81	172.16.149.31	255.255.0.0	172.16.1.16	NO

Figure 4: FNL with changed parameters

- Close the "Network Configurator"

2.2.3 Modbus TCP Server

All process data are mapped to the buffers of the Modbus TCP Server.

Properties:

Port Number (Not configurable)	0x502
Max. connections	8

Table 1: MODBUS TCP Properties

2.2.3.1 Supported functions

Modbus Function	Function Code	Direction	Buffer
Read Discrete Inputs	2	PROFIBUS → Modbus	PROFIBUS_to_MODBUS Buffer
Read Coils	1	PROFIBUS → Modbus	MODBUS_to_PROFIBUS Buffer
Read Input Registers	4	PROFIBUS → Modbus	All buffers
Read Holding Registers	3	PROFIBUS → Modbus	MODBUS_to_PROFIBUS Buffer
Write Coil	5	Modbus → PROFIBUS	MODBUS_to_PROFIBUS Buffer
Force Multiple Coils	15	Modbus → PROFIBUS	MODBUS_to_PROFIBUS Buffer
Force Multiple Registers	16	Modbus → PROFIBUS	MODBUS_to_PROFIBUS Buffer Control register 1 + 2
Write Single Register	6	Modbus → PROFIBUS	MODBUS_to_PROFIBUS Buffer Control register 1 + 2
Mask Write Registers	22	Modbus → PROFIBUS	MODBUS_to_PROFIBUS Buffer Control register 1 + 2
Read/Write Registers	23	Modbus ↔ PROFIBUS	All buffers

Table 2: Supported Modbus TCP Server functions

2.2.3.2 Modbus TCP Server address mapping

For the data exchange between Modbus and PROFIBUS the Modbus TCP server supports 2 different buffers.

PROFIBUS_to_MODBUS:

The buffer stores all data, which FNL reads from PROFIBUS and writes to Modbus TCP. The buffer can only be read by the Modbus TCP client. According to the FNL operation mode the buffer stores different data:

DP Master Operation mode:

The buffer stores all data which FNL reads as DP Master from the DP Slaves.

DP Slave Operation mode:

The buffer stores all data which FNL reads as DP Slave from the DP Master.

MODBUS_to_PROFIBUS:

The buffer stores all data, which FNL reads from Modbus TCP and writes to PROFIBUS. The buffer can be read and written by the Modbus TCP client. According to the FNL operation mode the buffer stores different data:

DP Master Operation mode:

The buffer stores all data which FNL writes as DP Master to the DP Slaves.

DP Slave Operation mode:

The buffer stores all data which FNL writes as DP Slave to the DP Master

Additionally different control registers for configuration- and status requests are available.

Layout:

Buffer	Register	Discrete Inputs	Buffer offset	Possible functions
PROFIBUS_to_MODBUS	1	1...16	0x0000...0x0001	read discrete Inputs
	2	17...32	0x0002...0x0003	read Input Registers
	.			read holding Registers
	.		.	read/write Registers(r)
	.		.	
	4096	65521...65536	0x1998...0x1999	
		Coil-Nr.		
MODBUS_to_PROFIBUS	4097	1...16	0x0000...0x0001	read coils
	4098	17...32		read Input Registers
	.			read holding Registers
	.			
	.			write coil
				force multiple coils
				force multiple Registers
				mask write Registers
				read/write Registers(rw)
	8192	65521...65536	0x1998...0x1999	Write single Register
Control register1	8193		0x0000...0x0001	read Input Registers
Control register2	8194		0x0002...0x0003	read holding Registers
				force multiple Registers
				mask write Registers
				read/write Registers(rw)
		Slave Status Info		
Life list (only DP Master operation mode)	8579	0...1	0x0000...0x0001	read/write Registers(r)
	8580	2...3	0x0002...0x0003	
	8581	4...5	0x0004...0x0005	
	.	.		
	.	.		
	.	.		
	8642	125...126	0x007D...0x007E	
Output data event counter (only DP Master Operation mode)	9005-9007		0x0000...0x0003	read/write Registers(r)
DP Slave summary state (only DP Master Operation mode)	9011		0x0000...0x0001	read/write Registers(r)

Table 3: Modbus TCP address mapping

The mapping of the PROFIBUS DP data as well as the meaning and structure of the registers depends on the FNL operation mode and is described in detail in the chapters below.

2.3 Operation as PROFIBUS DP Master

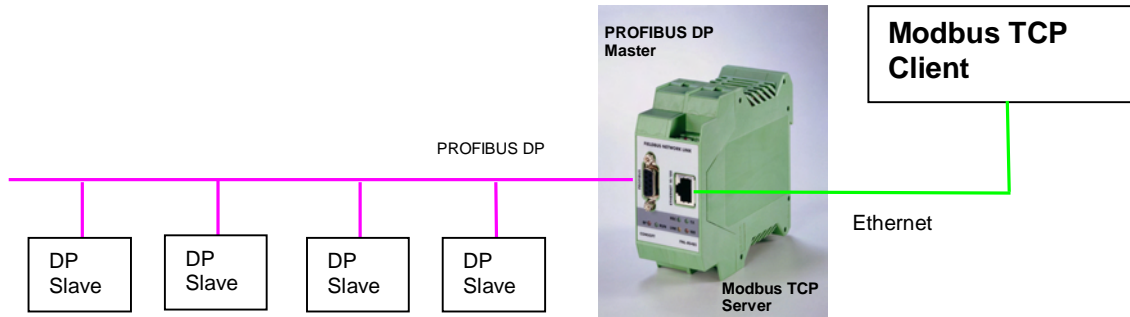


Figure 5: FNL – Operation as PROFIBUS DP Master

FNL works as PROFIBUS DP Master Class 0 and maps all DP Slave dependant input-output- and status data to the integrated Modbus TCP server.

2.3.1 Configuration as PROFIBUS DP Master

The PROFIBUS configuration is performed with the tool "COMSOFT *Configurator II*". "*Configurator II*" is a powerful tool to create PROFIBUS configurations. After creation and download of the PROFIBUS configuration you can test it immediately with the integrated monitor/modify mode and check for possible errors.

A detailed description how to get up and running can be found in the online help system of the "Configurator II".

- Start the program "*Configurator II*" from the Windows Start Menu "*Comsoft GmbH/Profibus Configurator*".
- Create the PROFIBUS configuration, download it to the FNL and test it with the "monitor/modify"-mode. FNL stores the PROFIBUS configuration in its Flash Type Memory, so it will be available at any time.

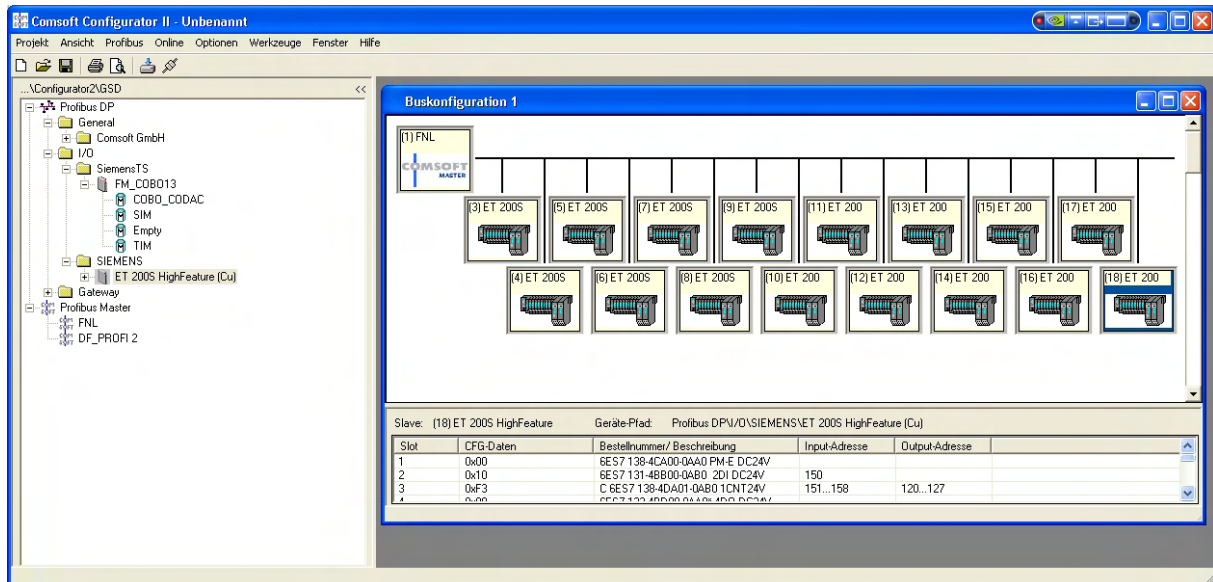


Figure 6: PROFIBUS configuration with CONFIGURATOR II

2.3.2 PROFIBUS DP configuration download

CONFIGURATOR II provides an integrated download function to flash the PROFIBUS DP Configuration into the FNL device.

Proceed the following steps:

- Select **Driver Selection** from the Online Menu:

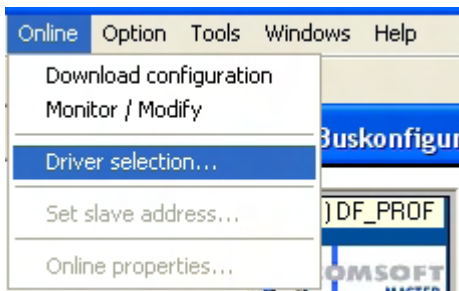


Figure 7: Driver selection

The installed FNL devices are displayed:

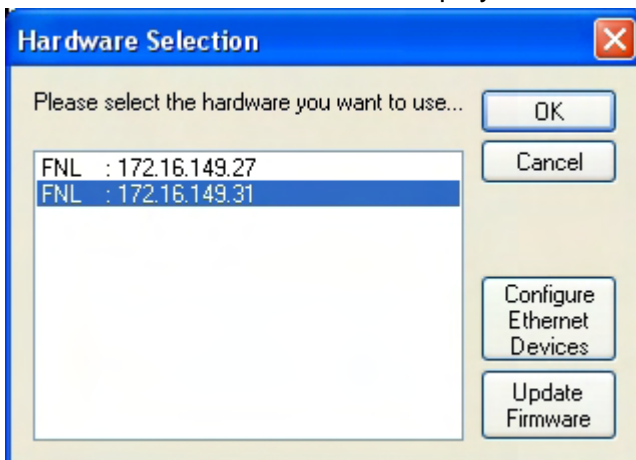


Figure 8: Installed FNL devices

Select a FNL device and click the **OK** button. If nothing is displayed, click the **Configure Ethernet Devices** button to rescan the available FNL devices. If still no FNL is found check your installation.

Refer to the online help system of CONFIGURATOR II for details of the driver selection.

- **Download** the PROFIBUS DP configuration

Click the **download** symbol in the toolbar of "Configurator II":

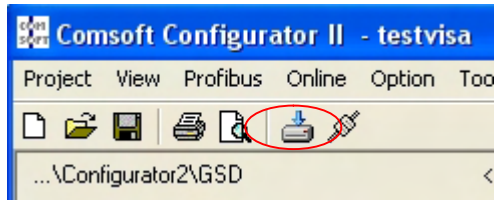


Figure 9: Download symbol

The PROFIBUS DP configuration is downloaded to the FNL device:

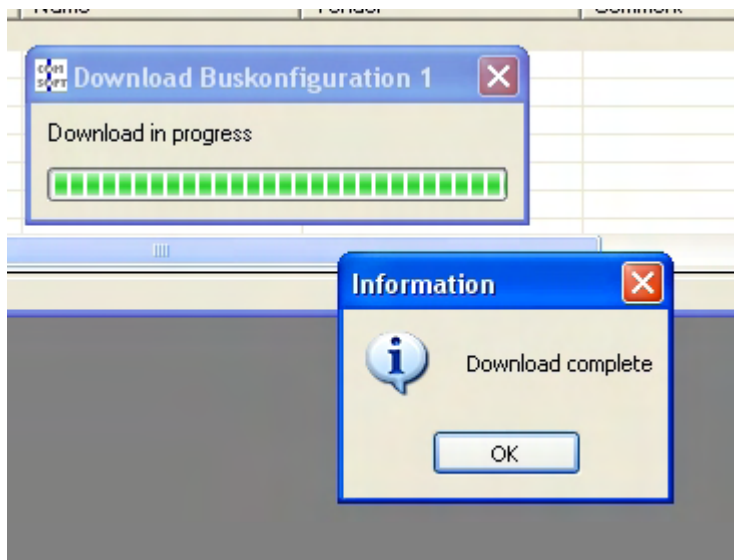


Figure 10: Download of PROFIBUS configuration

2.3.4.2 Mapping of the DP Input/Output data

The input data of the DP Slaves are mapped in consecutive order starting with the lowest DP Slave address to the PROFIBUS_to_MODBUS buffer.

The output data of the DP Slaves are mapped in consecutive order starting with the lowest DP Slave address to the MODBUS_to_PROFIBUS buffer.

The consecutive numbering of Discrete Inputs or Coils within the register starts with the lowest significant Bit (LSB).

The Modbus TCP Server does not combine input- or output data of different DP Slaves in a single Modbus register. In that case a DP Slave supports an odd number of input- or output bytes a pad byte is filled in the appropriate register.

Example configuration

DP Slave address	Number of input bytes	Number of output bytes
1	8	8
3	16	8
4	0	31
16	8	8

PROFIBUS_to_MODBUS Buffer

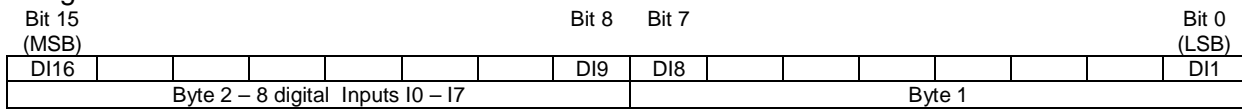
DP Slave	Register	Discrete Input	Buffer offset	DP Slave Input Bytes
1	1	1...16	0x0000...0x0001	Byte 1 and 2
	2	17...32	0x0002...0x0003	Byte 3 and 4
	3	33...48	0x0003...0x0004	Byte 5 and 6
	4	49...64	0x0004...0x0005	Byte 7 and 8
3	5	65...80	0x0005...0x0006	Byte 1 and 2
	6	81...96	0x0007...0x0008	Byte 3 and 4
	7	97...112	0x0009...0x000A	Byte 5 and 6
	8	113...128	0x000B...0x000D	Byte 7 and 8
	9	129...144	0x000E...0x000F	Byte 9 and 10
	10	145...160	0x0010...0x0011	Byte 11 and 12
	11	161...176	0x0012...0x0013	Byte 13 and 14
	12	177...192	0x0014...0x0015	Byte 15 and 16
16	13	193...208	0x0016...0x0017	Byte 1 and 2
	14	209...224	0x0018...0x0019	Byte 3 and 4
	15	225...240	0x001A...0x001B	Byte 5 and 6
	16	241...256	0x001C...0x001D	Byte 7 and 8

Table 5: Example PROFIBUS_to_MODBUS Buffer

Addressing of Discrete Inputs

DP Slave 1 installs a 8 digital input module (I0 – I7) on byte 2. This matches with the Discrete Inputs **9** (I0) – **16** (I7)

Register 1



MODBUS_to_PROFIBUS Buffer

DP Slave	Register	Coil	Buffer offset	DP Slave Output Bytes
1	4097	1...16	0x0000...0x0001	Byte 1 and 2
	4098	17...32	0x0002...0x0003	Byte 3 and 4
	4099	33...48	0x0003...0x0004	Byte 5 and 6
	4100	49...64	0x0004...0x0005	Byte 7 and 8
3	4101	65...80	0x0005...0x0006	Byte 1 and 2
	4102	81...96	0x0007...0x0008	Byte 3 and 4
	4103	97...112	0x0009...0x000A	Byte 5 and 6
	4104	113...128	0x000B...0x000D	Byte 7 and 8
4	4105	129...144	0x000E...0x000F	Byte 1 and 2
	4106	145...160	0x0010...0x0011	Byte 3 and 4
	4107	161...176	0x0012...0x0013	Byte 5 and 6
	4108	177...192	0x0014...0x0015	Byte 7 and 8
	4109	193...208	0x0016...0x0017	Byte 9 and 10
	4110	209...224	0x0018...0x0019	Byte 11 and 12
	4111	225...240	0x001A...0x001B	Byte 13 and 14
	4112	241...256	0x001C...0x001D	Byte 15 and 16
	4113	257...272	0x001E...0x001F	Byte 17 and 18
	4114	273...288	0x0020...0x0021	Byte 19 and 20
	4115	289...304	0x0022...0x0023	Byte 21 and 22
	4116	305...320	0x0024...0x0025	Byte 23 and 24
	4117	321...336	0x0026...0x0027	Byte 25 and 26
	4118	337...352	0x0028...0x0029	Byte 27 and 28
	4119	353...368	0x002A...0x002B	Byte 29 and 30
	4120	369...384	0x002C...0x002D	Byte 31 and Pad Byte
16	4121	385...400	0x002E...0x0017	Byte 1 and 2
	4122	401...416	0x0018...0x0019	Byte 3 and 4
	4123	417...432	0x001A...0x001B	Byte 5 and 6
	4124	433...448	0x001C...0x001D	Byte 7 and 8

Table 6: Example MODBUS_to_PROFIBUS Buffer

Addressing of Coils

DP Slave 4 installs on Byte 1 and Byte 2 a 16 digital output module O0 – O15. This matches in register 4105 with the coils **129** (O0) – **144** (O15).

Register 4105

Bit 15 (MSB)								Bit 8	Bit 7							Bit 0 (LSB)
C144								C137	C136							C129
Byte 2 – 8 digital outputs O8 – O15								Byte 1 – 8 digital outputs O0 – O7								

2.3.4.3 Indication of changing Input data

The Modbus registers 9005 and 9006 store a 32 bit event counter which is incremented with every change in the PROFIBUS_to_MODBUS Buffer what allows an optimized and more efficient access.

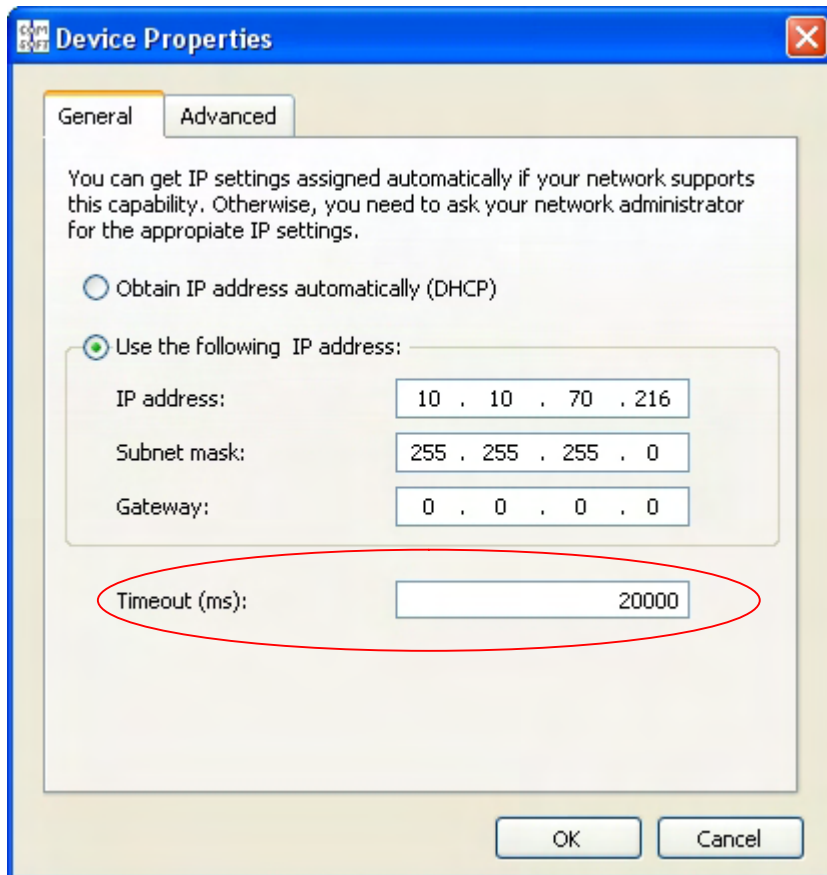
2.3.4.4 Watchdog Timer

FNL monitors the TCP/IP connection as well as the Modbus TCP Client activity. If the TCP/IP connection is disconnected or no more Modbus TCP Client requests are received and the watchdog timer expires, the PROFIBUS will be stopped and the FNL will reboot.

FNL accepts a reconnect on port 502 at every time and the watchdog timer will be automatically reactivated.

Setting the timeout value:

The timeout value is set in the **COMSOFT** Network Configurator (default: 20000 mS)



Note: The watchdog timer can be activated/deactivated via bit 1 in control register 1 (see chapter 2.3.4.1)

2.3.4.5 PROFIBUS DP Life List and Status Information

The life list stores status information for 127 DP Slaves (PROFIBUS address range 0 – 126) in the Modbus registers 8579 – 8642 starting with DP Slave address 0 in consecutive order, whereas 1 Modbus register stores the status information of 2 DP Slaves. The lower significant byte stores the lower PROFIBUS address and the higher significant byte stores the higher PROFIBUS address.

Example:

Register 8579 stores in the lower significant byte the status for DP Slave address 0 and in the higher significant byte the status for DP Slave address 1.

Register format:

15	8	7	0
Status for address 1		Status for address 0	

Buffer	Register	DP Slave status	Appropriate DP Slave addresses
Life list	8579	0xYY,0xYY	1,0
	8580	0xYY,0xYY	3,2
	8581	0xYY,0xYY	5,4
	.	.	
	.	.	
	.	.	
	8642	0xYY,0xYY	126,125

Table 7: Life list with DP Slave status

2.3.4.6 Status codes

Status Code	Meaning
0x40	DP Slave is in state data exchange, IO data are valid.
0x80	DP Slave is not in state data exchange and reports system diagnosis, IO data are invalid (i.e. DP Slave is disconnected or wrong configured).
0xC0	DP Slave is in state data exchange and reports external diagnosis, module related IO data may be invalid (i.e. wire break at analogue input module).
0x05	DP Slave is not configured and is not processed by the DP Master.

Table 8: Status codes DP Slave

If a different status code is indicated, please contact the **COMSOFT** Support (support@comsoft.de or by phone: +49 721 9497286 Monday-Friday 9.00 – 12.00 am and 01.00 – 04.00 pm)

Example configuration

DP Slave address	DP Slave status
0	Not configured
1	DP Slave is in state data exchange
2	Not configured
3	DP Slave is switched off
4	DP Slave is wrong configured
5 – 15	Not configured
16	DP Slave is in state data exchange and reports external diagnosis
17 – 124	Not configured
125	DP Slave is in state data exchange
126	Not configured

Life list:

Buffer	Register	Slave Status Info	Buffer offset
Life list	8579	0x40, 0x05	0x0000...0x0001
	8580	0x80, 0x05	0x0002...0x0003
	8581	0x05, 0x80	0x0004...0x0005
	8582	0x05, 0x05	
	8583	0x05, 0x05	
	8584	0x05, 0x05	
	8585	0x05, 0x05	
	8586	0x05, 0x05	
	8587	0x05, 0xC0	
	8588	0x05, 0x05	
	8589	0x05, 0x05	
	8590	0x05, 0x05	
	.	.	
	.	.	
	8642	0x05, 0x40	0x007D...0x007E

Table 9: Life list and DP Slave Status

Note: The PROFIBUS specific diagnostic information of the DP Slave is not available in the Modbus TCP Server.

2.3.4.7 DP Slave Summary State

To optimize the continuous check of every single DP Slave status, Modbus register 9011 stores a DP Slave status event counter which is incremented if the state of any DP Slave changes. This allows an optimized access to the life list based status information.

2.4 Operation as PROFIBUS DP Slave

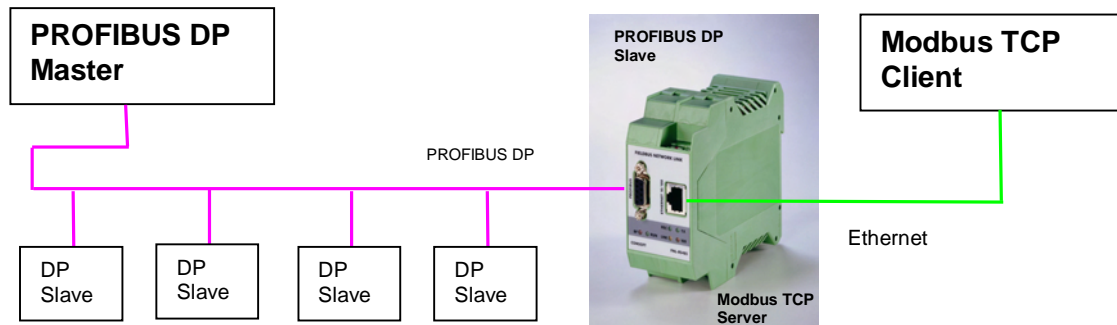


Figure 14: FNL – Operation as PROFIBUS DP Slave/Modbus TCP Server

FNL maps the input and output data exchanged with a DP Master and the communication status to the integrated Modbus TCP Server.

2.4.1 Configuration of FNL as DP Slave

2.4.1.1 Configuration of the DP Master

Configure FNL on the DP Master by the GSD file *COMS0A4B.gsd*. *COMS0A4B.gsd* is in scope of delivery and can be imported into the appropriate DP Master Configuration tool (i.e. Simatic S7® and HW-Konfig).

2.4.1.2 Configuration of FNL AUTO SLAVE Mode

In connection with the Modbus TCP Server FNL as DP Slave supports the AUTO Slave mode what means that only the configuration of a free PROFIBUS address is required. FNL automatically recognizes the current baud rate and the IO configuration from the DP Master.

To activate the AUTO SLAVE Mode and to configure the PROFIBUS address the **COMSOFT** Tool *DP Set Auto Slave.exe* is used:

```

DP Set Auto Slave v1.0 with TCP/IP access - Comsoft GmbH

|No| HW | Version | IP | Mac Address | AutoSlave | PB-Addr. |
|---|---|---|---|---|---|---|
| 0| FNL|U02.008 |192.168.20.51| 0-40-14-4-1-be| not supp. | not supp. |
| 1| FNL|U02.18.05|192.168.20.50| 0-40-14-4-2-2f| inactive  | 0 |

Please enter the No. of the target hardware : 1
The IpAddress of the selected target hardware is : 192.168.20.50
Create Socket...OK!
Connecting...OK!
Enter AutoSlave mode for the slave (0=inactive or 1=active) : 1
Enter Profibusaddress for the slave from 0 to 126 : 33
Set new values now. This may take a few seconds

|No| HW | Version | IP | Mac Address | AutoSlave | PB-Addr. |
|---|---|---|---|---|---|---|
| 0| FNL|U02.008 |192.168.20.51| 0-40-14-4-1-be| not supp. | not supp. |
| 1| FNL|U02.18.05|192.168.20.50| 0-40-14-4-2-2f| active | 33 |

Finished DP slave auto mode settings for FNL! Exit Application. Press key ...

```

1. Start DP Set Auto Slave.exe
2. Select the FNL to be configured from the displayed list by entering the appropriate number (No). If no FNL is displayed check the installation.
3. Activate the AUTO SLAVE Mode by entering the value 1
4. Enter a free PROFIBUS address between 0 and 126
5. Check the displayed response
6. Close the tool by pressing any key

2.4.2 Modbus TCP Server as DP Slave

2.4.2.1 Status of the FNL DP Slave in control register2 (8194)

In connection with AUTO SLAVE Mode no Modbus TCP Client based configuration is required. Control register 2 indicates the current status of FNL DP Slave.

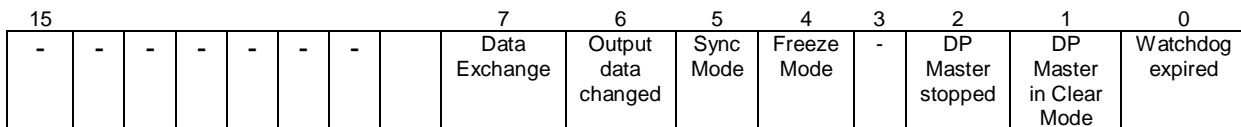


Figure 15: Format control register2

Note: Bits 0, 1, 2 and 6 are not automatically reset. The bits have to be reset by a Modbus TCP Client write register command to control register2.

Bit	Meaning	State	Comment
Bit 0	PROFIBUS Watchdog expired	0 = Watchdog active 1 = Watchdog expired	The DP Slave watchdog expires if the DP Master fails to repoll the DP Slave within the set watchdog time. The watchdog time is calculated by the DP Master configuration tool. The DP Masters sets the watchdog time on the DP Slave during the setup phase.
Bit 1	DP Master is in CLEAR state.	0 = CLEAR state inactive 1 = CLEAR state active	If the DP Master is in CLEAR state all output data transmitted to the FNL DP Slave are set to 0.
Bit 2	DP Master stopped	0 = DP Master active 1 = DP Master stopped	The bit is set if the DP Master stops polling the FNL DP Slave.
Bit 4	DP Master has activated the Freeze Mode	0 = Freeze Mode inactive 1 = Freeze Mode active	With Freeze Mode active the DP Slave does no longer update the input data transmitted to the DP Master until the DP Master deactivates the Freeze Mode. The Freeze Mode will be only activated/deactivated by the DP Master.

Bit	Meaning	State	Comment
Bit 5	DP Master has activated the Synch Mode	0 = Synch Mode inactive 1 = Synch Mode active	With Synch Mode active the DP Slave does no longer update the output data received from the DP Master until the DP Master deactivates the Sync Mode. The Sync Mode will be only activated/deactivated by the DP Master.
Bit 6	Output data changed	0 = no new output data available 1 = new output data available	Bit is set , if new output data received from the DP Master are available.
Bit 7	Data Exchange	0 = DP Slave is not in data exchange state 1 = data exchange state active	Only in state data exchange FNL DP Slave exchanges input- and output data with DP Master and the input- and output data are valid.

Table 10: Status bits DP Slave Mode

Examples:

Status code	Meaning
0x80	DP Slave is in state data exchange, IO data are valid
0x85	DP Slave is in state data exchange, IO data are valid. A DP Master restart happened (i.e. after reboot of FNL DP Slave or with a occasionally failing DP Master). Bit 0 and 2 can be reset manually by writing to control register2, to check for further DP Master fails.
0x00	No DP Master active after FNL DP Slave reboot
0x20	DP Master does not reach RUN state but stays in CLEAR mode. (Output data received from DP Master are always 0).
0xA0	DP Slave is in state data exchange, Synch Mode is activated by DP Master.
0x90	DP Slave is in state data exchange, Freeze Mode is activated by DP Master.
0xB0	DP Slave is in state data exchange, Freeze and Synch Mode are activated by DP Master.

Table 11: FNL DP Slave status codes

2.4.2.2 Mapping of the FNL DP Slave Input/Output data

Input data are transmitted from the FNL DP Slave to the DP Master. They are stored in the **MODBUS_to_PROFIBUS Buffer** and are written by the Modbus TCP Client application.

Output data are transmitted from the DP Master to the FNL DP Slave. They are stored in the **PROFIBUS_to_MODBUS Buffer** and are read by the Modbus TCP Client application.

The consecutive numbering of Discrete Inputs or Coils within the register starts with the lowest significant Bit (LSB).

Example configuration FNL DP Slave

DP Slave Address	Number of Input Bytes	Number of Output Bytes
16	8	16

MODBUS_to_PROFIBUS Buffer

Stores FNL DP Slave input data, i.e. the data transmitted from the FNL DP Slave to the DP Master.

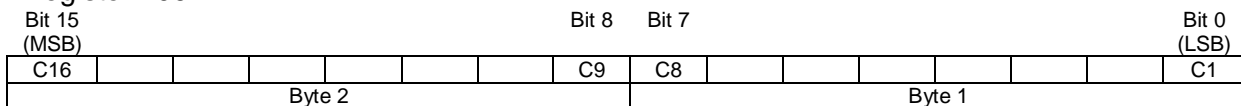
DP Slave	Register	Coil	Buffer offset	DP Slave Input Bytes
16	4097	1...16	0x0000...0x0001	Byte 1 and 2
	4098	17...32	0x0002...0x0003	Byte 3 and 4
	4099	33...48	0x0003...0x0004	Byte 5 and 6
	4100	49...64	0x0004...0x0005	Byte 7 and 8

Table 12: Input data FNL DP Slave

Addressing of Coils

16 Bits in register 4097 match with the coils **1 – 16**.

Register 4097



PROFIBUS_to_MODBUS Buffer

Stores FNL DP Slave output data, i.e. the data transmitted from the DP Master to the FNL DP Slave.

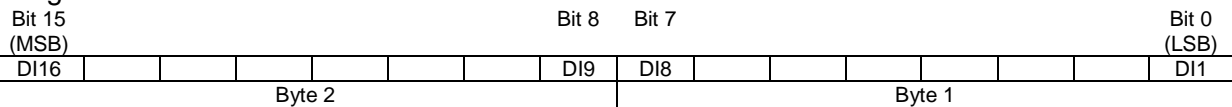
DP Slave	Register	Discrete Input	Buffer offset	DP Slave Output Bytes
16	1	1...16	0x0000...0x0001	Byte 1 and 2
	2	17...32	0x0002...0x0003	Byte 3 and 4
	3	33...48	0x0003...0x0004	Byte 5 and 6
	4	49...64	0x0004...0x0005	Byte 7 and 8
	5	65...80	0x0005...0x0006	Byte 9 and 10
	6	81...96	0x0007...0x0008	Byte 11 and 12
	7	97...112	0x0009...0x000A	Byte 13 and 14
	8	113...128	0x000B...0x000D	Byte 15 and 16

Table 13: Output data FNL DP Slave

Addressing of Discrete Inputs

16 Bits in register 1 match with the Discrete Inputs **1 – 16**

Register 1



2.4.2.3 Connection monitoring

2.4.2.3.1 PROFIBUS DP communication

The PROFIBUS DP Communication can be monitored by checking the status bits of control register2 (see chapter 2.4.2.1).

2.4.2.3.2 Start up and Monitoring of the Modbus TCP Client connection

In AUTO SLAVE Mode FNL DP Slave in principle does not monitor the Modbus TCP Client connection. After switch on the FNL activates the AUTO SLAVE Mode automatically and will be recognized by a DP Master as a properly working DP Slave despite if a Modbus TCP Client application is connected or not. This is necessary for example in connection with a PLC since the start up of a PLC is much more faster than that of a Modbus TCP Client system and the PLC would fall into maintenance due to the delayed start up of FNL DP Slave. Furthermore there is no disconnection of the PROFIBUS during normal operation if the Modbus TCP Client applications disconnects from the FNL DP Slave.

If the monitoring of the Modbus TCP Client application by the DP Master is required, this can be implemented by exchanging life information between the Modbus TCP Client and the DP Master on application level.