



DN-CBM-DP and **DN-DP**

PROFIBUS[®]-DP / DeviceNet[™]-Gateways

Software Manual

to Products: C.2846.02 and C.2930.02

NOTE

The information in this document has been carefully checked and is believed to be entirely reliable. **esd** makes no warranty of any kind with regard to the material in this document, and assumes no responsibility for any errors that may appear in this document. In particular descriptions and technical data specified in this document may not be constituted to be guaranteed product features in any legal sense.

esd reserves the right to make changes without notice to this, or any of its products, to improve reliability, performance or design.

All rights to this documentation are reserved by **esd**. Distribution to third parties and reproduction of this document in any form, whole or in part, are subject to **esd**'s written approval.

© 2020 esd electronics gmbh, Hannover

esd electronics gmbh

Vahrenwalder Str. 207
30165 Hannover
Germany

Phone: +49-511-372 98-0
Fax: +49-511-372 98-68
E-mail: info@esd.eu
Internet: www.esd.eu

Trademark Notices

PROFIBUS® is a registered trademark of PROFIBUS Nutzerorganisation e.V..

DeviceNet™ is a trademark of the Open DeviceNet Vendor Association, Inc (ODVA).

All other trademarks, product names, company names or company logos used in this manual are reserved by their respective owners.

Document File:	I:\Texte\Doku\MANUALS\CAN\DN-DP\Englisch\DN-DP_DN-CBM-DP_Software_en_13.wpd
Date of Print:	2020-07-16

Described software version:	DeviceNet Core: 101 DP/DeviceNet: 2.2.x
Hardware order no.:	DN-CBM-DP C.2846.02 DN-DP C.2930.02

Changes in the chapters

The changes in the document listed below affect changes in the hardware as well as changes in the description of the facts, only.

Version	Chapter	Changes as compared with previous version	Date
1.0	-	First issue of the English software manual of DN-CBM-DP <u>and</u> DN-DP	2007-10-31
1.1	-	Classification of notes inserted, notes revised	2016-01-19
	7.	New chapter: "DeviceNet Errata"	
1.2	1.4	New chapter: "Range of Application"	2017-03-28
1.3	1.4	Description modified	2020-07-16
	5.4.4	New Chapter: "Restrictions"	
	7.6	Chapter deleted	

Technical details are subject to change without further notice.

Classification of Notice Statements and Information

NOTICE

Notice statements are used to notify people on hazards that could result in things other than personal injury, like property damage or software error.



NOTICE

This NOTICE statement contains the general mandatory sign and gives information that must be heeded and complied with for a safe and error-free use.

INFORMATION



INFORMATION

Notes to point out something important or useful.

Typographical Conventions

Number Representation

All numbers in this document are base 10 unless designated otherwise. Hexadecimal numbers have a prefix of 0x, and binary numbers have a prefix of 0b. For example, 42 is represented as 0x2A in hexadecimal and 0b101010 in binary.

Contents

1. Overview	7
1.1 About this Manual	7
1.2 Introduction into Functionality of the Firmware	7
1.3 Configuration via PROFIBUS-DP	7
1.4 Range of Application	7
2. Functionality of the Local Firmware	8
2.1 PROFIBUS-Slave Address	8
2.2 User Data	9
2.3 Watchdog (Reaction Control)	9
2.4 Diagnostics	9
2.5 Parameter Telegram (DeviceNet Bit Rate)	9
2.6 Global-Control Services (FREEZE, SYNC, UNSYNC)	9
2.7 PROFIBUS-DP Profiles	9
3.2 Implementation	10
3.2.1 Strategy	10
3.2.2 Start-Up	10
3.2.3 Data Transfer	10
3.3 Diagnostics via LED Display	11
3.3.1 DN-CBM-DP Module	11
3.3.1.1 PROFIBUS LED (LED 3) of DN-CBM-DP	11
3.3.1.2 DeviceNet LEDs (LED1, LED2) of DN-CBM-DP	11
3.3.2 DN-DP Module	13
3.4 Slave Diagnostics	15
3.4.1 Diagnostic Bytes 0...5	15
3.4.1.1 Diagnostic Byte 0: Station Status 1	16
3.4.1.2 Diagnostic Byte 1: Station Status 2	17
3.4.1.3 Diagnostic Byte 2: Station Status 3	17
3.4.1.4 Diagnostic Byte 3: Master-PROFIBUS Address	17
3.4.1.5 Diagnostic Bytes 4 and 5: Manufacturer Identification	17
3.4.2 Extended (Module-Specific) Diagnostic Bytes	18
3.4.3 Device Related Diagnostics	18
3.4.4 Identifier Related Diagnostics	20
4. GSD File	21
4.1 GSD File of the DN-CBM-DP Module	21
4.2 GSD File of the DN-DP Module	23
5. Configuration via SIMATIC Manager	25
5.1 Introduction	25
5.2 Course of Configuration	25
5.2.1 Set PROFIBUS Address	26
5.2.2 Parameter Telegram	27
5.2.3 Assigning the Slots of the DP Slave	28
5.2.4 Configuration of DeviceNet Modules	29
5.2.5 Save Settings to Hard Disc	29
5.3 Description of Input Window ' <i>DP Slave Properties</i> '	30

5.4 Configuration Examples	33
5.4.1 Configuration Example with txOffs and rxOffs	33
5.4.2 Configuration Example for the Local Slave	34
5.4.3 Configuration Example for a Remote Slave	35
5.4.4 Restrictions for the DN-gateway	35
6. Manual Configuration of a DP-Master	37
6.1 Structure of the Parameter Frame	37
6.2 Structure of the Configuration Frame	38
6.3 The Communication Window	42
7. DeviceNet Errata PROFIBUS-DP to DeviceNet Gateways	43
7.1 Master/Scanner: The UCMM loses Message Ids	43
7.2 Master/Scanner: CoS Allocation is incorrect	43
7.3 Slave/Device: Network Status is not always indicated correctly	44
7.4 Network Access State Machine after communication failure	44
7.5 Network Status on DeviceNet power loss	44

1. Overview

1.1 About this Manual

This manual describes the local firmware of the modules DN-CBM-DP and the DN-DP together. In this manual both modules are referred to as *DN-gateway*. Differences in the software are noted.

The local firmware controls the data exchange between PROFIBUS-DP (abbreviated to PROFIBUS below) and DeviceNet.

1.2 Introduction into Functionality of the Firmware

The *DN-gateway* simulates a slave device with a defined number of input and output bytes to the PROFIBUS. After the gateway has been configured DeviceNet modules can be operated like PROFIBUS slaves.

The PROFIBUS output bytes are transmitted to the CAN-bus. Received CAN data are treated as input data by the PROFIBUS.

The PROFIBUS station address is set directly at the *DN-gateway* by means of coding switches.

1.3 Configuration via PROFIBUS-DP

The *DN-gateway* is configured via the PROFIBUS. The Siemens SIMATIC Manager for S7, for example, can be used as a configuration tool. Here, the gateway is assigned with logical modules which are assigned with further parameters such as the PLC address, data direction, data length and MACID.

1.4 Range of Application

The DN-DP module can link any PROFIBUS DP master to a DeviceNet network. The *DN-gateway* operates as a PROFIBUS DP slave-I/O component. At the DeviceNet network the *DN-gateway* can operate as a scanner to control several DeviceNet slave devices or as a single slave device to be connected to one DeviceNet scanner.

The *DN-gateway* is always configured via a PROFIBUS DP master.

Configuration of the *DN-gateway* as a DeviceNet Scanner

As a DeviceNet scanner, several DeviceNet slave devices can be addressed. The maximum number is between 20 and 40. This number depends on the configured PROFIBUS DP slave I/O type of each DeviceNet slave device.

Up to a total of 312 user data bytes can be configured for the *DN-gateway* as a whole. For each direction up to 240 user data bytes can be set up.

Configuration of the *DN-gateway* as a Single DeviceNet Slave

The *DN-gateway* can be configured as exactly one DeviceNet slave.

With this configuration, it can be connected to exactly one DeviceNet scanner.

2. Functionality of the Local Firmware

The *DN-gateway* is able to work on the DeviceNet as scanner only, as slave only and also simultaneously as scanner and slave. The input and output data blocks of the DeviceNet modules are mapped to the PROFIBUS input and output data frames, respectively.

If the module is used as scanner and slave simultaneously, local scanner and slave use the same MACID to access the DeviceNet network. For further information on the configuration and data mapping, please refer to section 5.3.

The following figure represents the functionality of the firmware:

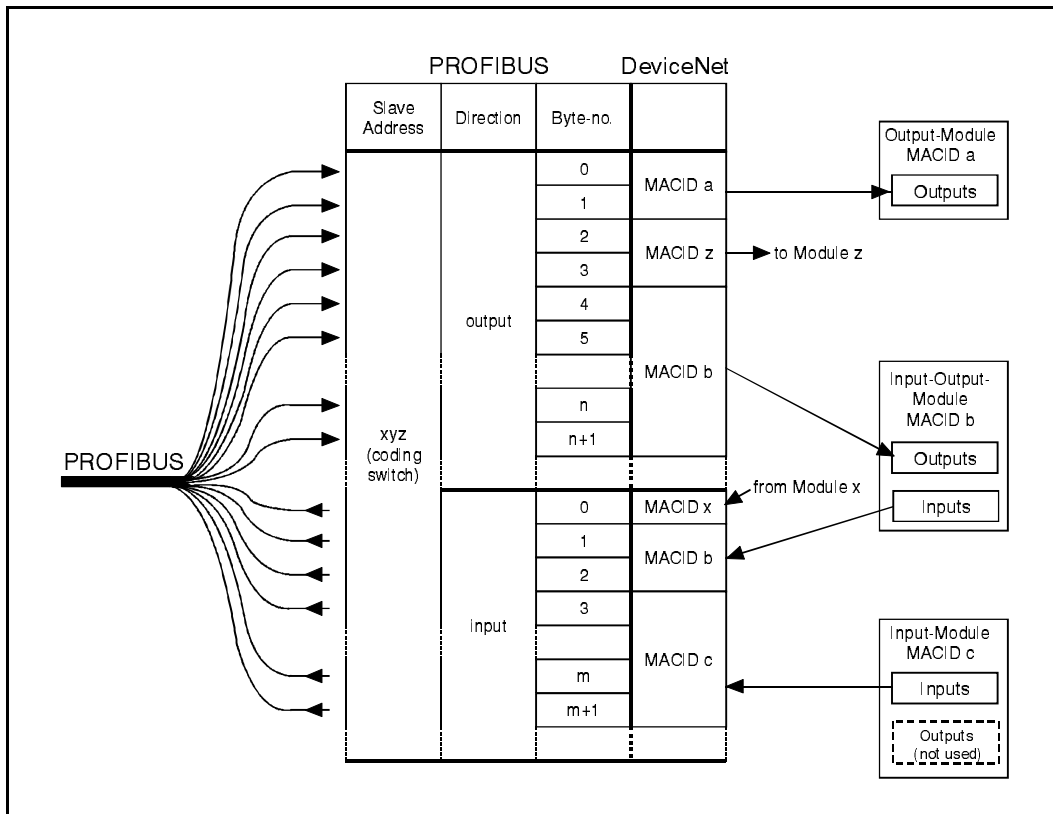


Fig. 2.1.1: Overview of functions of the *DN-gateway*

2.1 PROFIBUS-Slave Address

The *DN-gateway* simulates a slave module on the PROFIBUS side. The slave address is set by means of coding switches at the module. When powering on the module the hexadecimal PROFIBUS address is read from the hexadecimal switches. The settings have to be changed before switching on the module, because changes are ignored during operation.

The available address range is *hexadecimal* 0x01 to 0x7E or *decimal* 1 to 126. If an address smaller than 1 (0x01) is set, address 1 is valid. If an address larger than 126 (0x7E) is set, address 126 is valid.

If the address 255 (0xFF) is selected, the firmware enters the firmware upgrade mode on bootup. The upper coding switch (SW211, HIGH) is used to set the MSBs, while the LSBs are set by means

of the lower coding switch (SW210, LOW).

The PROFIBUS-slave address can *only* be set via coding switches. It *cannot* be programmed by means of a class 2 master via the command 'Set_Slave_Address'.

2.2 User Data

The *DN-gateway* simulates the DeviceNet modules as input and output bytes on the PROFIBUS side. The sum of configured input and output bytes must not exceed 312 bytes.

On the other hand the maximum output or input frame length is 240 bytes. Therefore you may only configure 240 input bytes together with 72 output bytes at maximum, and vice versa. Another restriction is that you **must** configure at least one output byte to make the PROFIBUS controller work.

2.3 Watchdog (Reaction Control)

The firmware can be run with activated or deactivated reaction control. It is recommendable, though, to run it with activated reaction control.

2.4 Diagnostics

The DP-slave diagnostics can be used. The module supports device related and identifier related diagnostics. The diagnostics will be described in more detail in section "Slave Diagnostics" on page 15.

2.5 Parameter Telegram (DeviceNet Bit Rate)

In addition to the seven standard bytes of the configuration, the *DN-gateway* supports three module-specific bytes. Here, the DP master can change the DeviceNet bit rate and the module's MACID. Setting the bit rate by means of the parameter telegram is described on page 28.

2.6 Global-Control Services (FREEZE, SYNC, UNSYNC)

The Global-Control services have not yet been implemented.

2.7 PROFIBUS-DP Profiles

The PROFIBUS-DP profiles are not supported yet.

3. Implementing and Diagnostics

3.1 Prerequisites for Implementation

This chapter describes the implementation of the *DN-gateway* at a PROFIBUS which is controlled by a Siemens SIMATIC-S7-300 or S7-400.

To implement the module as described below, you need the configuration program ‘SIMATIC-Manager’ with the tool ‘HW-configurator’.

3.2 Implementation

3.2.1 Strategy

Please make the following steps to implement the module:

1	Install and wire the <i>DN-gateway</i> (power supply, DeviceNet interface...; see hardware manual).
2	Set the PROFIBUS address of the module by means of the coding switch.
3	Connect the PROFIBUS connector to the PROFIBUS interface of the <i>DN-gateway</i> .
4	Switch on the power supply for the <i>DN-gateway</i> . Now the module has to run.

3.2.2 Start-Up

After switching on the power supply, the *DN-gateway* starts automatically. It does not have its own mains switch.

The module receives projection data from the DP master and evaluates the specifications in them. If the projection complies with the structure, the *DN-gateway* starts the data transfer.

3.2.3 Data Transfer

After the module is configured, the data transfer starts automatically: If the PLC master changes output data of a slot (i.e. sub module/identifier), the data is transmitted from the *DN-gateway* to the configured DeviceNet module. When the *DN-gateway* receives data, it provides these to the PLC master. The configuration is described in chapter 5 ‘Configuration via the SIMATIC-Manager’ starting at page 25.

3.3 Diagnostics via LED Display

The LED indication of the DN-CBM-DP and the DN-DP module are described in different chapters due to the differences in the hardware.

3.3.1 DN-CBM-DP Module

The indication of the LEDs of the DN-CBM-DP module (C.2846.02) is controlled by the firmware.

3.3.1.1 PROFIBUS LED (LED 3) of DN-CBM-DP

The status of LED3 is described in table 3.3.1

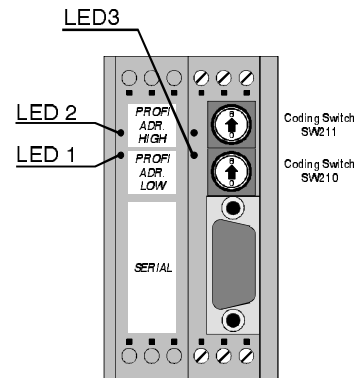


Fig. 3.3.1: Position of LEDs

LED	Function	LED Status	Meaning
LED 3 (red)	PROFIBUS- data exchange	off	no data exchange
		on	data exchange via PROFIBUS

Table 3.3.1: Status of LED3 of DN-CBM-DP

3.3.1.2 DeviceNet LEDs (LED1, LED2) of DN-CBM-DP

Since firmware revision 2.2.0 LED1 and LED2 of the DN-CBM-DP are used to emulate a DeviceNet compliant combined Module/Network Status LED.

Imagine that the LED1 is a green LED (while it is a red one in reality) and imagine that LED1 and LED2 are combined into a single LED. Under these assumptions these 2 LEDs behave like a DeviceNet Module/Network Status LED.

Display of DN-CBM-DP LED1 and LED2	Original DeviceNet Module / Network Status LED Display	DeviceNet Status	Meaning
LED1/ LED2/off slowly flashing alternating	green/red/off slowly flashing	Not Configured	DeviceNet network interface has not been configured
LED1: off LED2: off	off	Not On-line, Not Powered	- DN-CBM-DP has not completed the Dup_MAC_ID test yet - DN-CBM-DP may not be powered
LED1: on LED2: off	green;	Device Operational AND On-line, Connected	DN-CBM-DP is operating in normal condition and is on-line with connections in the established state
LED1: flashing LED2: off	flashing green;	Device Operational AND On-line, Not Connected	- DN-CBM-DP is operating in normal condition and is on-line with no connections in the established state - DN-CBM-DP has passed Dup_MAC_ID test, is on-line but has no established connections to other nodes, i. e. the internal slave of the DN-DP is not owned by a master/scanner.
LED1: off LED2: flashing	flashing red;	Minor Fault and/or Connection Time-Out	- Recoverable fault and/or one or more I/O-connections are in the Timed-Out state
LED1: off LED2: on	red;	Critical Fault or Critical Link Failure	DN-CBM-DP has an unrecoverable fault; may need replacing. Failed communication device. DN-CBM-DP has detected an error that has rendered it incapable of communicating on the network (Duplicate MAC_ID or Bus-off)
LED1 and LED2 are flashing alternating	flashing red/green;	Communication Faulted and Received an Identify Comm Fault Request-Long Protocol	A specific Communication Faulted device. DN-CBM-DP has detected a network access error and is in the Communication faulted state. DN-CBM-DP has subsequently received and accepted an Identify Communication Faulted Request- Long Protocol message (not yet implemented)

Table 3.3.2: Status LED1 and LED2 of DN-CBM-DP

3.3.2 DN-DP Module

The DN-DP module (C.2930.02) is equipped with four LEDs in the front panel.

The firmware controls the indication of the LEDs in accordance with DeviceNet Specification release 2.0.

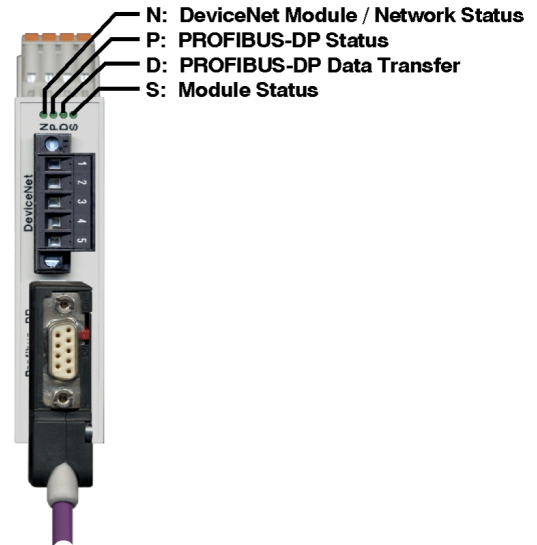


Fig. 2: Position of the LEDs in the front panel

LED	Function	LED Display	Status	Meaning
N (green/red)	DeviceNet Module / Network Status	slowly flashing green/red/off	Not Configured	DeviceNet network interface has not been configured
		off	Not On-line, Not Powered	- DN-DP has not completed the Dup_MAC_ID test yet - DN-DP may not be powered
		green	Device Operational AND On-line, Connected	DN-DP is operating in normal condition and is on-line with connections in the established state
		flashing green	Device Operational AND On-line, Not Connected	- DN-DP is operating in normal condition and is on-line with no connections in the established state - DN-DP has passed Dup_MAC_ID test, is on-line but has no established connections to other nodes, i. e. the internal slave of the DN-DP is not owned by a master/scanner.
		flashing red	Minor Fault and/or Connection Time-Out	Recoverable fault and/or one or more I/O-connections are in the Timed-Out state
		red	Critical Fault or Critical Link Failure	DN-DP has an unrecoverable fault; may need replacing. Failed communication device. DN-DP has detected an error that has rendered it incapable of communicating on the network (Duplicate MAC_ID or Bus-off)
		flashing red/green	Communication Faulted and Received an Identify Comm Fault Request- Long Protocol	A specific Communication Faulted device. DN-DP has detected a network access error and is in the Communication faulted state. DN-DP has subsequently received and accepted an Identify Communication Faulted Request- Long Protocol message (not yet implemented)

LED	Function	LED Display	Meaning	Error handling
P (green)	PROFIBUS- DP Status	off	no power supply	check the 24 V power supply
		1x short flash	looking for bit rate	the connection to the DP master has failed, check the PROFIBUS connection (fault in wiring in PROFIBUS cable, short circuit, terminating impedance in wrong position ?)
		2x short flashes	bit rate is monitored	check the PROFIBUS address specified
		3x short flashes	waiting for parameter telegram	parameter telegram is faulty, diagnostics via SIMATIC-Manager or system function SFC13 (DPNRM_DG) (see chap. 3.4)
		4x short flashes	waiting for configuration telegram	configuration telegram is faulty, diagnostics via SIMATIC-Manager or system function SFC13 (DPNRM_DG) (see chap. 3.4)
		on	PROFIBUS OK	-
D (green)	PROFIBUS- DP Data Transfer	off	no data exchange	-
		on	data exchange via PROFIBUS	-
S (green)	Module Status	on	+24 V power supply voltage connected	-
		flashes	module is in firmware update mode	-

Table 3.3.3: LED status of DN-DP module

3.4 Slave Diagnostics

In addition to the six diagnostic bytes predefined in standard DIN EN 19245, part 3, the *DN-gateway* supports some module-specific diagnostic bytes.

The slave diagnostics can be requested by the following function components:

Automation device family	Number	Name
SIMATIC with IM 308-C	FB 192	FB IM308C
SIMATIC S7/M7	SFC 13	SFC DPNRM_DG

Table 3.4.1: Function component for requesting the slave diagnostics

3.4.1 Diagnostic Bytes 0...5

The assignment of these diagnostic bytes has been predefined in standard DIN EN 19425, part 3. Below, the status messages will be described in consideration of the *DN-gateway*.

The following designations will be used for this:

Byte number	Status-byte designation
0	station status 1
1	station status 2
2	station status 3
3	master-PROFIBUS address
4	manufacturer-identification high byte
5	manufacturer-identification low byte

Table 3.4.2: Diagnostic bytes 0...5

3.4.1.1 Diagnostic Byte 0: Station Status 1

Station status 1 contains error messages of the DP slave. If a bit is '0', no error applies. A bit set to '1' signals an error.

Bit	Error message if bit value = '1'	Error handling
0	DP slave cannot be addressed by the master	<ul style="list-style-type: none"> - correct PROFIBUS address set at the <i>DN-gateway</i>? - bus connector correctly wired? - power supply available at <i>DN-gateway</i>? - power off/power on executed at <i>DN-gateway</i> in order to read in DP address?
1	DP slave is not yet ready for data exchange	<ul style="list-style-type: none"> - wait until the <i>DN-gateway</i> has completed start up of PROFIBUS
2	The configuration data transmitted from DP master to DP slave do not correspond to the DP slave structure.	<ul style="list-style-type: none"> - check whether the station type and the <i>DN-gateway</i> structure have been correctly entered via the configuration tool
3	The slave has got extended diagnostic data.	<ul style="list-style-type: none"> - request and evaluate extended diagnostic data
4	The requested function is not supported by the DP slave.	<ul style="list-style-type: none"> - check projecting
5	DP master cannot interpret the response of the DP slave.	<ul style="list-style-type: none"> - check bus structure
6	Wrong parameter.	<ul style="list-style-type: none"> - evaluate diagnostic bytes 9 and 10, see table 3.4.5 and 3.4.6
7	DP slave has already been set by another master.	<ul style="list-style-type: none"> - this bit is always '1', if you, e.g., just access the <i>DN-gateway</i> by means of a PG or another DP master. The PROFIBUS address of the setting master is in the diagnostic byte 'Master-PROFIBUS address'.

Table 3.4.3: Bits of station status 1

3.4.1.2 Diagnostic Byte 1: Station Status 2

Station status 2 contains status messages from the DP slave. If a bit is '1', the according message is active. A bit set to '0' signals an inactive message.

Bit	Error message if bit value = '1'
0	DP slave requests for parameter setting and reconfiguration when he sets this bit. The bit remains active until parameter setting is finished.
1	A diagnostics message applies. The DP slave cannot operate until the error has been removed (static diagnostics message).
2	This bit is always '1'.
3	The response monitoring for the <i>DN-gateway</i> is activated.
4	DP slave has received freeze command.
5	DP slave has received SYNC command.
6	This bit is always '0'.
7	DP slave is deactivated.

Table 3.4.4: Bits of station status 2

3.4.1.3 Diagnostic Byte 2: Station Status 3

Station status 3 is reserved and without significance for the *DN-gateway*.

3.4.1.4 Diagnostic Byte 3: Master-PROFIBUS Address

The PROFIBUS address of the master which was the last to set the DP slave and has got reading and writing access to the DP slave is stored in this byte.

3.4.1.5 Diagnostic Bytes 4 and 5: Manufacturer Identification

The manufacturer identification has been coded into two bytes.

For the **DN-CBM-DP** module the designation **0x0566** is returned.

For the **DN-DP** module the designation **0x0B4D** is returned.

3.4.2 Extended (Module-Specific) Diagnostic Bytes

The DN-CBM-DP module supports two types of extended diagnostics. First the device related diagnostics and second the identifier related diagnostics. For the exact meaning of these terms please refer to the PROFIBUS specification.

The *DN-gateway* supports diagnostic bytes 6 to 10 for module-specific diagnostic messages.

3.4.3 Device Related Diagnostics

The device related diagnostics shows information about the device (*DN-gateway*) status and can only occur, when the firmware is internally not in the data exchange mode.

The mapping from the diagnostic bytes content to clear text error strings for a PLC diagnostic tool is done by means of the GSD file. Because this allows only a coarse error display you may view the diagnostic bytes on a hex display with this tool. For further evaluation please look at the following tables.

Diagnostic byte offset	Meaning																			
5	Standard diag.	defined in the PROFIBUS spec. (see previous chapter)																		
6	header byte	device related diagram contains diagnostics block total length in bits 5...0 (len). Values for header byte: <table border="1" style="margin-left: 40px;"> <thead> <tr> <th>Bit no.</th> <th>7</th> <th>6</th> <th>5</th> <th>4</th> <th>3</th> <th>2</th> <th>1</th> <th>0</th> </tr> </thead> <tbody> <tr> <td>Content</td> <td colspan="2">Diagnostic type</td> <td colspan="6">block total length including header byte</td> </tr> </tbody> </table> Diagnostic type: 00 device related 01 identifier related (see chapter 3.4.4) 10 channel related (not used by gateway) 11 reserved (not used by gateway)	Bit no.	7	6	5	4	3	2	1	0	Content	Diagnostic type		block total length including header byte					
Bit no.	7	6	5	4	3	2	1	0												
Content	Diagnostic type		block total length including header byte																	
7	error code	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%;">0x01</td> <td style="width: 45%;">DPE_KILL_DNET: (len=2)</td> <td style="width: 40%;">still deallocating DeviceNet slaves from previous configuration</td> </tr> <tr> <td>0x02</td> <td>DPE_MEM_OVR: (len=3)</td> <td>internal memory overflow</td> </tr> <tr> <td>0x04</td> <td>DPE_PARA: (len=4)</td> <td>parameter error</td> </tr> <tr> <td>0x08</td> <td>DPE_CONF: (len=5)</td> <td>configuration error</td> </tr> <tr> <td>0x10</td> <td>DPE_DNET_DOWN: (len=3)</td> <td>“DeviceNet not operational”</td> </tr> </table>	0x01	DPE_KILL_DNET: (len=2)	still deallocating DeviceNet slaves from previous configuration	0x02	DPE_MEM_OVR: (len=3)	internal memory overflow	0x04	DPE_PARA: (len=4)	parameter error	0x08	DPE_CONF: (len=5)	configuration error	0x10	DPE_DNET_DOWN: (len=3)	“DeviceNet not operational”			
0x01	DPE_KILL_DNET: (len=2)	still deallocating DeviceNet slaves from previous configuration																		
0x02	DPE_MEM_OVR: (len=3)	internal memory overflow																		
0x04	DPE_PARA: (len=4)	parameter error																		
0x08	DPE_CONF: (len=5)	configuration error																		
0x10	DPE_DNET_DOWN: (len=3)	“DeviceNet not operational”																		
8...10	depends on error code, see table 3.4.6: device related diagnostics #2																			

Table 3.4.5: Device Related Diagnostics #1

Error Code (contained in diagnostic byte offset 7)	Meaning		
	byte 8	byte 9	byte 10
DPE_KILL_DNET (0x01)	not present	not present	not present
DPE_MEM_OVR (0x02)	internal error location	not present	not present
DPE_PARA (0x04)	1 not enough para bytes	received param comment length	not present
	2 too much para bytes	received param comment length	
	3 wrong ident no. high	received ident no high	
	4 wrong ident no. low	received ident no low	
	5 user byte 1 not zero	no. of wrong comment byte	
	6 wrong baud rate	no. of wrong comment byte	
DPE_CONF (0x08)	PROFIBUS identifier # (=slot number)	=0: global error >0: # of wrong byte in the identifiers comment field	error code; tbd.
DPE_DNET_DOWN (0x10)	contains CAN state and DN network state as described in table 3.4.7 and 3.4.8	not present	not present

Table 3.4.6: Device Related Diagnostics #2

The byte 8 shows additionally the DeviceNet network state, if the DeviceNet network is NOT operational.

Bit no.	7	6	5	4	3	2	1	0
Content	CAN state		DeviceNet Network state					

CAN state (bit 7, bit 6)	
	bit7 bit 6
0x00	0 0 CAN OK
0x40	0 1 Warn
0xC0	1 1 bus-off

DeviceNet Network state (bit 5 - 0)		
0x21	NEV_DUP_MAC_SEQ	The gateway performs the duplicate MACID check sequence.
0x22	NEV_OPERATIONAL	The DeviceNet network is in the operational state. This state is never seen as device related diagnostic message.
0x23	NEV_PWR_FAIL	The gateway has a DeviceNet network power fail detected (24V network power supply is missing).
0x24	NEV_BUS_OFF_PROBE	The gateway probes if it can transmit any DeviceNet messages (CAN frames) on the network after a bus-off condition occurred. The bus-off condition may be the result of a missing network power supply.
0x25	NEV_COMM_FAULT	The gateway is in the communication faulted state. This may occur after a bus-off condition or is a result of a duplicate MACID check failure.
0x26	NEV_MAC_RESET	(not yet implemented)
0x27	NEV_IDLE	(not yet implemented)

Table 3.4.7: Byte 8 if error code is 0x10

3.4.4 Identifier Related Diagnostics

The identifier related diagnostics can only occur, when the firmware is internally in the data exchange mode. When the gateway is configured one DeviceNet module is equivalent to one slot on the hardware configurator display. This way the DeviceNet module is configured via one so called 'identifier byte' and its comment bytes. The identifier byte blocks have numbers which are the same as the slot numbers.

In data exchange mode the gateway will send identifier related diagnostic bytes in the format described in the PROFIBUS specification, if any of the configured DeviceNet modules is not in the operational (connected) state. If all configured modules are operational no identifier related diagnostics will be present.

To clarify the behaviour of the identifier related diagnostics, assume a DeviceNet AC drive that may have a bit in its input data block that shows if it has a correct AC voltage supplied. If the AC drive is not connected to the DeviceNet network or may lack the 24 V DeviceNet power supply, you will get an identifier related diagnostics for this AC drive, i. e. the bit corresponding to its identifier block number (slot) will be set in the identifier diagnostic bit string. This shows that the DeviceNet connection is faulty. On the other hand when the AC power supply of this drive fails, you won't get any identifier related diagnostics, because from the DeviceNet point of view the module is operational. If you want to see this drive status, you have to configure this status byte as an input.

On startup you will at first get identifier related diagnostics for all configured DeviceNet modules after the Duplicate MACID Check succeeded, because the connections to the modules are not yet established. After some time all identifier related diagnostics should vanish.

One hint if you have configured the gateway to work as slave for a remote scanner. The bit of the identifier block, which you used to configure the gateway as slave, is set when the remote scanner does not maintain the connection to the local slave on the gateway.

4. GSD File

4.1 GSD File of the DN-CBM-DP Module

Below, the GSD file (Device Master Data) of the DN-CBM-DP module has been printed. The specification printed here are for orientation. Decisive is the data contained in the GSD file **DNDP0566.GSD**, included in the product package.

```

;=====
; (c) esd electronic system design GmbH Hannover
;
; PROFIBUS-DP Geratetestamdatei
; Version: 1.00
;
; Autor: Ulrich Hartmann
; Erstellungsdatum: V1.0 12.05.2000 uh born from cdps04a4.gsd
; Aenderungen:
;=====
; Art des Parameters
; (M) Mandatory (zwingend notwendig)
; (O) Optional (zusätzlich möglich)
; (D) Optional mit Default=0 falls nicht vorhanden
; (G) mindestens einer aus der Gruppe passend zur entsprechenden Baudrate
#PROFIBUS_DP
;--- Kapitel 2.3.2 Allgemeine DP-Schlüsselwörter ---
GSD_Revision = 1 ; (M ab GSD_Revision 1) (Unsigned8)
Vendor_Name = "esd" ; (M) Herstellername (Visible-String 32)
Model_Name = "DN-CBM-DP" ; (M) Herstellerbezeichnung des DP-Geraetes (Visible-String 32)
Revision = "V1.0" ; (M) Ausgabestand des DP-Geraetes (Visible-String 32)
Revision_Number = 1 ; (M ab GSD_Revision 1) (Unsigned8 (1 bis 63)) (1234)
Ident_Number = 1382 ; (M) Gerätetyp des DP-Gerätes (Unsigned16)
Protocol_Ident = 0 ; (M) Protokollkennung des DP-Geraetes 0: Profibus-DP (Unsigned8)
Station_Type = 0 ; (M) DP-Geraetetyp 0: DP-Slave (Unsigned8)
FMS_supp = 0 ; (D) kein FMS/DP-Mischgeraet (Boolean)
Hardware_Release = "V1.1" ; (M) Hardware Ausgabestand des DP-Geraetes (Visible-String 32)
Software_Release = "V1.0" ; (M) Software Ausgabestand des DP-Geraetes (Visible-String 32)
9.6_supp = 1 ; (G) 9,6 kBaud wird unterstuetzt
19.2_supp = 1 ; (G) 19,2 kBaud wird unterstuetzt
;31.25_supp = 1 ; fuer Gateway CAN-CBM-DP nicht moeglich (1234)
45.45_supp = 1 ; (G ab GSD_Revision 2) 45,45 kBaud wird unterstuetzt
93.75_supp = 1 ; (G) 93,75 kBaud wird unterstuetzt
187.5_supp = 1 ; (G) 187,5 kBaud wird unterstuetzt
500_supp = 1 ; (G) 500 kBaud wird unterstuetzt
1.5M_supp = 1 ; (G) 1,5 MBaud wird unterstuetzt
3M_supp = 1 ; (G ab GSD_Revision 1) 3 MBaud wird unterstuetzt
6M_supp = 1 ; (G ab GSD_Revision 1) 6 MBaud wird unterstuetzt
12M_supp = 1 ; (G ab GSD_Revision 1) 12 MBaud wird unterstuetzt
MaxTsd_r_9.6 = 60 ; (G)
MaxTsd_r_19.2 = 60 ; (G)
;MaxTsd_r_31.25 = 15 ; fuer Gateway DN-CBM-DP nicht moeglich (1234)
MaxTsd_r_45.45 = 60 ; (G ab GSD_Revision 2)
MaxTsd_r_93.75 = 60 ; (G)
MaxTsd_r_187.5 = 60 ; (G)
MaxTsd_r_500 = 100 ; (G)
MaxTsd_r_1.5M = 150 ; (G)
MaxTsd_r_3M = 250 ; (G ab GSD_Revision 1)
MaxTsd_r_6M = 450 ; (G ab GSD_Revision 1)
MaxTsd_r_12M = 800 ; (G ab GSD_Revision 1)
Redundancy = 0 ; (D) keine redundante Uebertragungstechnik
Repeater_Ctrl_Sig = 0 ; (D) RTS-Signalpegel (CNTR-P) Pin 4 des 9pol. SUB-D
; 0: nicht vorhanden 1: RS 485 2: TTL
24V_Pins = 0 ; (D) Bedeutung der 24V Pins des 9pol. SUB-D (Pin 7 24V; Pin 2 GND)
; 0: nicht angeschlossen 1: Input 2: Output
; Implementation_Type = "Visible-String" ; (1234)
Bitmap_Device = "DNDP00_N" ; (O ab GSD_Revision 1)
Bitmap_Diag = "DNDP00_D" ; (O ab GSD_Revision 1)
Bitmap_SF = "DNDP00_S" ; (O ab GSD_Revision 1)

```

```

;--- Kapitel 2.3.4      DP-Slave-bezogene Schluesselwoerter ---
Freeze_Mode_supp      = 0      ; (D) Der Freeze-Mode wird nicht unterstuetzt
Sync_Mode_supp        = 0      ; (D) Der Sync-Mode wird nicht unterstuetzt
Auto_Baud_supp        = 1      ; (D) Die Automatische Baudratenerkennung wird unterstuetzt
Set_Slave_Add_supp    = 0      ; (D) Die Slave-Adresse kann vom Master nicht gesetzt werden
;User_Prm_Data_Len    = 3      ; (D) Hoechstlaenge von User-Parameter-Daten
;User_Prm_Data=0x00,0x00,0x3F ; (O) User-Parameter-Daten
Min_Slave_Intervall   = 20     ; (M) Minimaler Abstand zwischen 2 DDLM_Data_Exchange-Aufrufen (xx * 100us)
Modular_Station       = 1      ; (D) 0: Kompaktstation 1: Modulare Station
Max_Module            = 64     ; (M falls modulare Station) Hoechstanzahl der Module einer Modularen Station
Max_Input_Len         = 240    ; (M falls modulare Station) Hoechstlaenge der Eingangsdaten einer Modularen
                          Station
Max_Output_Len        = 240    ; (M falls modulare Station) Hoechstlaenge der Ausgangsdaten einer Modularen
                          Station
Max_Data_Len          = 312    ; (O nur falls modulare Station) Groesste Summe der Ein- und Ausgangsdaten einer
                          Modularen Station in Bytes
Unit_Diag_Bit(0000) = "Deallocating DeviceNet modules" ;
Unit_Diag_Bit(0001) = "Memory overflow" ;
Unit_Diag_Bit(0002) = "Wrong parametrisation" ;
Unit_Diag_Bit(0003) = "Wrong configuration" ;
Max_Diag_Data_Len     = 16     ; max. 16 Byte Diagnosedaten
Modul_Offset          = 0      ; (D ab GSD_Revision 1) erste Steckplatznummer
Max_User_Prm_Data_Len= 3
PrmText=1
Text(0)=" 125 kbit/s"
Text(1)=" 250 kbit/s"
Text(2)=" 500 kbit/s"
EndPrmText
PrmText=2
Text(0)="No"
Text(1)="Yes"
EndPrmText
ExtUserPrmData=1 "DeviceNet-Bitrate"
Unsigned8 0 0-2
Prm_Text_Ref=1
EndExtUserPrmData
ExtUserPrmData=2 "Communication Window"
Bit(7) 0 0-1
Prm_Text_Ref=2
EndExtUserPrmData
ExtUserPrmData=3 "Own MACID"
Bitarea(0-5) 63 0-63
EndExtUserPrmData
Ext_User_Prm_Data_Const(0)=0x00,0x00,0x3F
Ext_User_Prm_Data_Ref(1)=1
Ext_User_Prm_Data_Ref(2)=2
Ext_User_Prm_Data_Ref(2)=3
Slave_Family = 9@DN@V01
Periphery      = "ET 200"
OrderNumber    = "C.2846.02"

```

4.2 GSD File of the DN-DP Module

Below, the GSD file (Device Master Data) of the DN-DP module has been printed. The specification printed here are for orientation. Decisive is the data contained in the GSD file **DNDP0B4D.GSD**, included in the product package.

```

=====
; (c) esd electronic system design GmbH Hannover
;
; PROFIBUS-DP Geratetestamdatei
; Version: 1.00
;
; Autor: Ulrich Hartmann
; Erstellungsdatum: V1.0 15.10.2007 uh born from dndp0566.gsd
; Aenderungen:
=====
; Art des Parameters
; (M) Mandatory (zwingend notwendig)
; (O) Optional (zusätzlich möglich)
; (D) Optional mit Default=0 falls nicht vorhanden
; (G) mindestens einer aus der Gruppe passend zur entsprechenden Baudrate
#PROFIBUS_DP
;--- Kapitel 2.3.2 Allgemeine DP-Schlüsselwörter ---
GSD_Revision = 1 ; (M ab GSD_Revision 1) (Unsigned8)
Vendor_Name = "esd" ; (M) Herstellername (Visible-String 32)
Model_Name = "DN-DP" ; (M) Herstellerbezeichnung des DP-Geraetes (Visible-String 32)
Revision = "V1.0" ; (M) Ausgabezustand des DP-Geraetes (Visible-String 32)
Revision_Number = 1 ; (M ab GSD_Revision 1) (Unsigned8 (1 bis 63)) (1234)
Ident_Number = 2893 ; (M) Gerätetyp des DP-Gerätes (Unsigned16)
Protocol_Ident = 0 ; (M) Protokollkennung des DP-Geraetes 0: Profibus-DP (Unsigned8)
Station_Type = 0 ; (M) DP-Geraetetyp 0: DP-Slave (Unsigned8)
FMS_supp = 0 ; (D) kein FMS/DP-Mischgeraet (Boolean)
Hardware_Release = "V1.1" ; (M) Hardware Ausgabezustand des DP-Geraetes (Visible-String 32)
Software_Release = "V2.3.0" ; (M) Software Ausgabezustand des DP-Geraetes (Visible-String 32)
9.6_supp = 1 ; (G) 9,6 kBaud wird unterstuetzt
19.2_supp = 1 ; (G) 19,2 kBaud wird unterstuetzt
;31.25_supp = 1 ; fuer Gateway CAN-CBM-DP nicht moeglich (1234)
;45.45_supp = 1 ; (G ab GSD_Revision 2) 45,45 kBaud wird unterstuetzt
93.75_supp = 1 ; (G) 93,75 kBaud wird unterstuetzt
187.5_supp = 1 ; (G) 187,5 kBaud wird unterstuetzt
500_supp = 1 ; (G) 500 kBaud wird unterstuetzt
1.5M_supp = 1 ; (G) 1,5 MBaud wird unterstuetzt
3M_supp = 1 ; (G ab GSD_Revision 1) 3 MBaud wird unterstuetzt
6M_supp = 1 ; (G ab GSD_Revision 1) 6 MBaud wird unterstuetzt
12M_supp = 1 ; (G ab GSD_Revision 1) 12 MBaud wird unterstuetzt
MaxTsd_r_9.6 = 60 ; (G)
MaxTsd_r_19.2 = 60 ; (G)
;MaxTsd_r_31.25 = 15 ; fuer Gateway DN-CBM-DP nicht moeglich (1234)
;MaxTsd_r_45.45 = 60 ; (G ab GSD_Revision 2)
MaxTsd_r_93.75 = 60 ; (G)
MaxTsd_r_187.5 = 60 ; (G)
MaxTsd_r_500 = 100 ; (G)
MaxTsd_r_1.5M = 150 ; (G)
MaxTsd_r_3M = 250 ; (G ab GSD_Revision 1)
MaxTsd_r_6M = 450 ; (G ab GSD_Revision 1)
MaxTsd_r_12M = 800 ; (G ab GSD_Revision 1)
Redundancy = 0 ; (D) keine redundante Uebertragungstechnik
Repeater_Ctrl_Sig = 0 ; (D) RTS-Signalpegel (CNTR-P) Pin 4 des 9pol. SUB-D
; 0: nicht vorhanden 1: RS 485 2: TTL
24V_Pins = 0 ; (D) Bedeutung der 24V Pins des 9pol. SUB-D (Pin 7 24V; Pin 2 GND)
; 0: nicht angeschlossen 1: Input 2: Output
; Implementation_Type = "Visible-String" ; (1234)
Bitmap_Device = "DNDPME_N" ; (O ab GSD_Revision 1)
Bitmap_Diag = "DNDPME_D" ; (O ab GSD_Revision 1)
Bitmap_SF = "DNDPME_S" ; (O ab GSD_Revision 1)
;--- Kapitel 2.3.4 DP-Slave-bezogene Schlüsselwörter ---
Freeze_Mode_supp = 0 ; (D) Der Freeze-Mode wird nicht unterstuetzt
Sync_Mode_supp = 0 ; (D) Der Sync-Mode wird nicht unterstuetzt
Auto_Baud_supp = 1 ; (D) Die Automatische Baudratenerkennung wird unterstuetzt
Set_Slave_Add_supp = 0 ; (D) Die Slave-Adresse kann vom Master nicht gesetzt werden
;User_Prm_Data_Len = 3 ; (D) Hoechstlaenge von User-Parameter-Daten
;User_Prm_Data=0x00,0x00,0x3F ; (O) User-Parameter-Daten
Min_Slave_Intervall = 6 ; (M) Minimaler Abstand zwischen 2 DDLM_Data_Exchange-Aufrufen (xx * 100us)
Modular_Station = 1 ; (D) 0: Kompaktstation 1: Modulare Station
Max_Module = 64 ; (M falls modulare Station) Hoechstanzahl der Module einer Modularen Station
Max_Input_Len = 240 ; (M falls modulare Station) Hoechstlaenge der Eingangsdaten einer Modularen Station

```

GSD File

```
Max_Output_Len      = 240 ; (M falls modulare Station) Hoechstlaenge der Ausgangsdaten einer Modularen Station
Max_Data_Len        = 312 ; (O nur falls modulare Station) Groesste Summe der Ein- und Ausgangsdaten einer
                    Modularen Station in Bytes
Unit_Diag_Bit(0000) = "Deallocating DeviceNet modules" ;
Unit_Diag_Bit(0001) = "Memory overflow" ;
Unit_Diag_Bit(0002) = "Wrong parametrisation" ;
Unit_Diag_Bit(0003) = "Wrong configuration" ;
Unit_Diag_Bit(0004) = "DeviceNet not operational" ;
Max_Diag_Data_Len   = 16      ; max. 16 Byte Diagnosedaten
Modul_Offset        = 0       ; (D ab GSD_Revision 1) erste Steckplatznummer
Module="Comm.Window" 0xB7
EndModule
Max_User_Prm_Data_Len= 3
PrmText=1
Text(0)=" 125 kbit/s"
Text(1)=" 250 kbit/s"
Text(2)=" 500 kbit/s"
EndPrmText
PrmText=2
Text(0)="No"
Text(1)="Yes"
EndPrmText
ExtUserPrmData=1 "DeviceNet-Bitrate"
Unsigned8 0 0-2
Prm_Text_Ref=1
EndExtUserPrmData
ExtUserPrmData=2 "Communication Window"
Bit(7) 0 0-1
Prm_Text_Ref=2
EndExtUserPrmData
ExtUserPrmData=3 "Own MACID"
Bitarea(0-5) 63 0-63
EndExtUserPrmData
Ext_User_Prm_Data_Const(0)=0x00,0x00,0x3F
Ext_User_Prm_Data_Ref(1)=1
Ext_User_Prm_Data_Ref(2)=2
Ext_User_Prm_Data_Ref(2)=3
Slave_Family = 9@DN@V01
OrderNumber      = "C.2930.02"
```


5. Configuration via SIMATIC Manager

5.1 Introduction

The *DN-gateway* is completely configured via the PROFIBUS and no configuration information remains in the gateway after a power cycle.



NOTICE

Without correct configuration via the SIMATIC manager the *DN-gateway* and the DeviceNet participants connected do not operate together and operation of the DeviceNet participants connected can be disturbed.

In particular the DeviceNet-Bitrate configured in the *DN-gateway* and the module-ID (MACID) must match the settings of the DeviceNet participants connected!

If problems should occur, further information can be obtained with the diagnostics as described in the chapters “4.3 Diagnostics via LED Display” and “4.4 Slave Diagnostics”.

5.2 Course of Configuration

Please follow the steps below to configure the *DN-gateway*:

1. Select the right *DN-gateway*

Select menu *Hardware Catalogue* and there *Additional Field Devices* and *Other*. There select e.g. DN-CBM-DP or DN-DP.

2. Set PROFIBUS Address

Set the PROFIBUS address as described in chapter 5.2.1 on page 26.

3. Parameter Telegramm

(set DeviceNet Bit Rate, Module-ID (MACID) and Communication Window)

Configure the configuration settings by means of the parameter telegram as described in chapter 5.2.2 on page 27.

4. Assignment of the Slots of the DP-slaves

Assign the slots as described in chapter 5.2.3 on page 28.

5. Configuration of the Slots (PLC-Address)

Configure the slots as described in chapter 5.2.4 on page 29.

6. Save settings on hard disk

Save the settings as described in chapter 5.2.5 on page 29.

5.2.1 Set PROFIBUS Address

A window opens in which you have to specify the PROFIBUS station address.



NOTICE

The *hexadecimal* address set at the coding switches has to be *converted* into a *decimal* value and entered here!

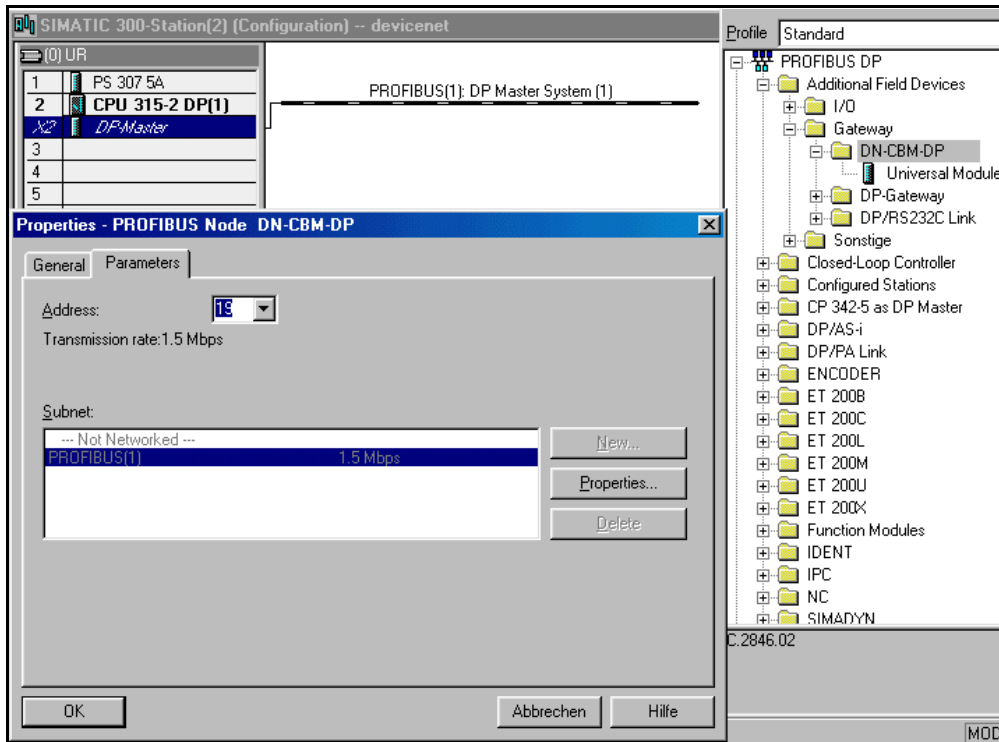


Fig. 5.2.1: Example: Setting the PROFIBUS address of the DN-CBM-DP

5.2.2 Parameter Telegram

In the configuration window the module 'DN-CBM-DP' (or 'DN-DP' respectively) is now automatically added. The DeviceNet bit rate and MACID defaults from the GSD file to 125 kBaud and MACID 63 (0x3F).

The Communication Window is disabled by default. You may change the DeviceNet baud rate, the Communication Window and the MACID in the DP Slave Properties window.

The module-specific bytes of the parameter telegram can be changed in the Properties window. To open the window double click the header of the DP-slave window (here: '(19) DN-CBM-DP').

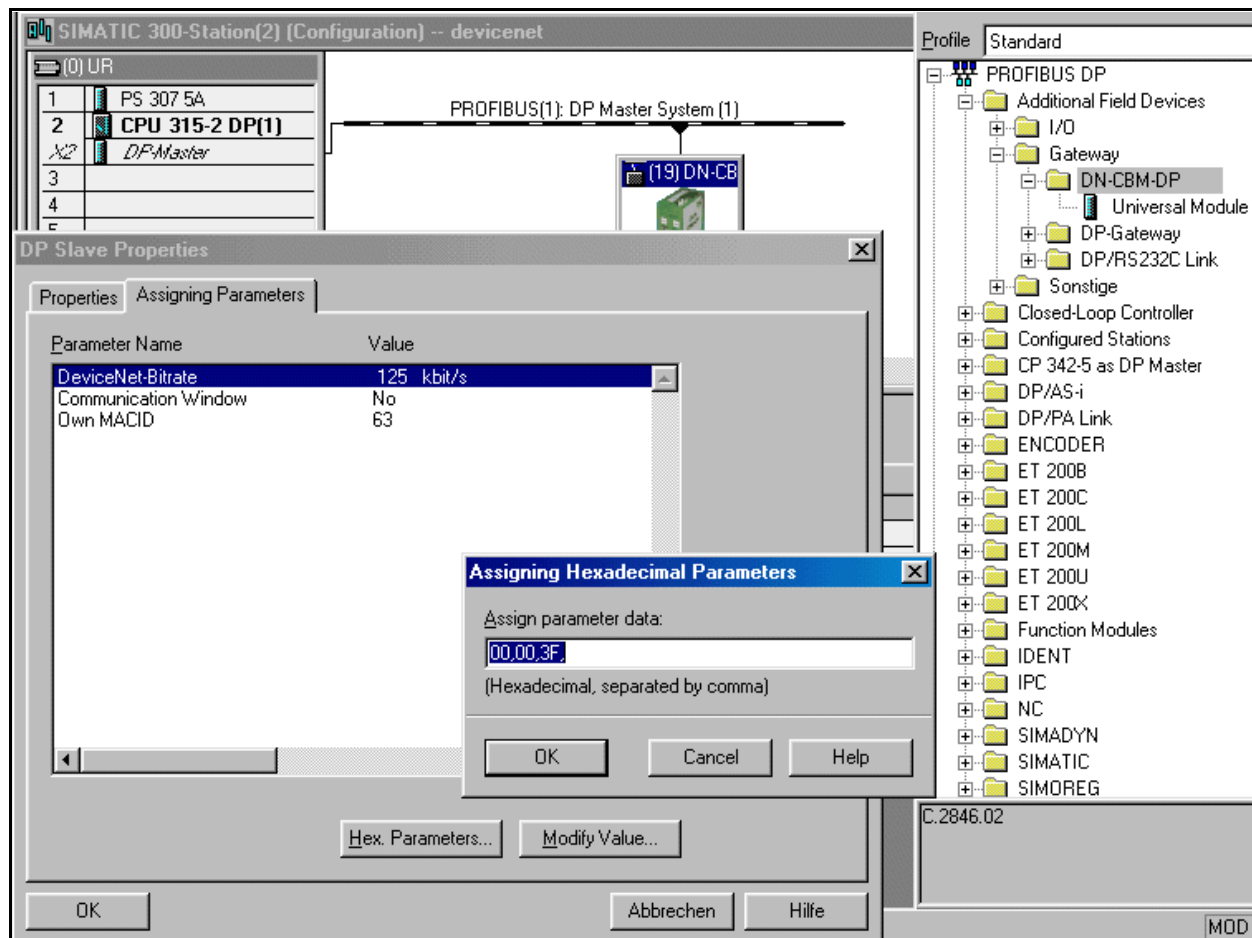


Fig. 5.2.2: Setting the DeviceNet bit rate in the DP-slave Properties window

To change the parameter data manually, click the button *Hex Parameters...* to open the dialogue box *Assigning Hexadecimal Parameters*.



NOTICE

The first byte of the parameter data has always to be set to '00'.
 In the second byte the bit rate is set. Please refer to table 5.2.1 for allowed values.
 In the third byte the MACID of the module (0...63 dec.) is selected by the hexadecimal value 0x00...0x3F. The most significant bits of this byte are flags and have to be zero for future compatibility. The DeviceNet MACID you specify here is used by the scanner and also by the local slave in the gateway.

2. hexadecimal parameter byte	Bit rate [kbit/s]
02	500
01	250
00	125

Table 5.2.1: Selection by hexadecimal parameters

5.2.3 Assigning the Slots of the DP Slave

The desired number of slots to be used by the DP slave for data exchange is set by double clicking the device 'Universal Module' for each DeviceNet module you want to communicate with. In the DP-slave window the assigned slots are represented by a '0'.

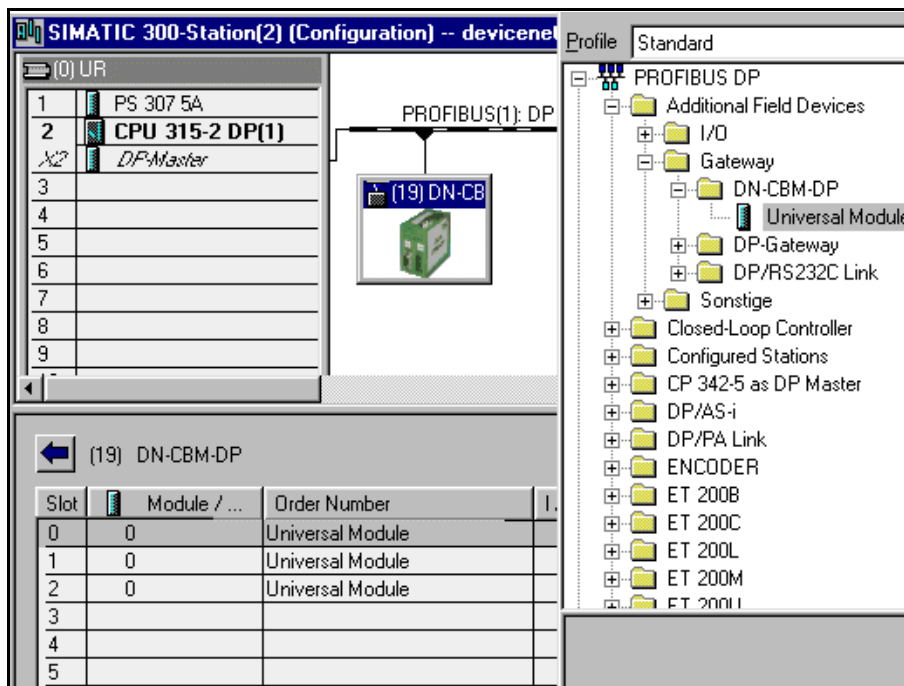


Fig. 5.2.3: Addition of new DeviceNet modules

5.2.4 Configuration of DeviceNet Modules

In order to configure the DeviceNet modules the slot entry has to be double clicked. A properties window opens in which the simulated PLC slots are configured. Below, two examples are shown:

PLC parameter:

Data direction: input/output
 Output address: 1 dec.
 Output length: 1
 Output unit: word
 Consistency: unit
 Input address: 1
 Input length: 7
 Input unit: byte
 Consistency: unit

PLC parameter:

Data direction: input/output
 Output address: 3 dec.
 Output length: 4
 Output unit: byte
 Consistency: unit
 Input address: 8
 Input length: 3
 Input unit: word
 Consistency: unit

DeviceNet parameter:

EPR: 0x3E8 = 1000 ms
 MACID: 5 hex
 Output length: 6
 Input length: 7
 Tx offset: 4

DeviceNet parameter:

EPR: 0x7D0 = 2000 ms
 MACID: 3 hex
 Output length: 4
 Input length: 6

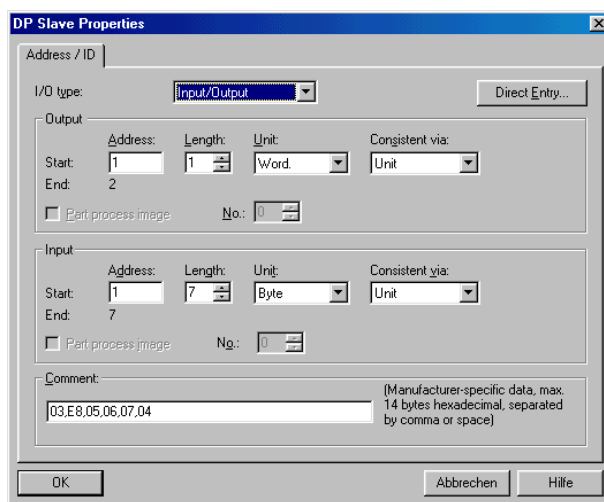


Fig. 5.2.4: Configuration example

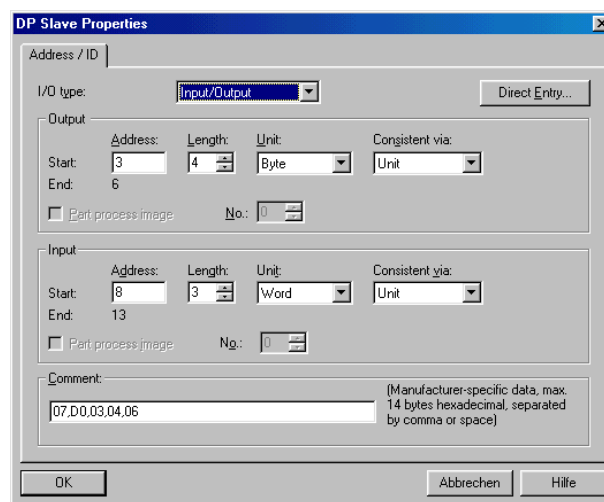


Fig. 5.2.5: Configuration example

The individual parameters of the properties window will be explained in detail in the following chapter.


5.2.5 Save Settings to Hard Disc

Now you have to save the settings via menu points *Station/Save* to hard disc. Afterwards the settings are transmitted to the PLC by means of menu points *Target System/Load in Unit*.

5.3 Description of Input Window ‘DP Slave Properties’

The first parameters configure the **PROFIBUS** side of a slot/module.

- In field **I/O-Type** ‘input’ or ‘output’ or ‘input/output’ has to be selected, depending on the data direction desired. Other properties are not permissible. The input/output direction is seen here from the PLC’s point of view.
- In field **Address** the PLC-I/O address is entered as a **decimal value**.
- By means of fields **Length** and **Unit** the number of data bytes is specified. You may also work with word wide data.
- The entry in field **Consistent via** shows whether the data is to be transmitted as individual unit (bytes, words, etc.) or as complete package during a PLC cycle. This function is only to be set to ‘whole length’ if required, because the transmission as ‘unit’ is faster. The upper limit for the I/O-length is 32 bytes if you want to transmit consistent over the total length.
The limit is determined by the S7-PLC, the *DN-gateway* gives no restrictions, which is able to transfer the whole PROFIBUS data frame in a consistent manner.



NOTICE

If the data is to be transmitted consistently for the entire length, you have to specify this here *and* you have to use SFC14 and SFC15 (refer to Step7-PLC Manual).

The following parameters configure how this slave behaves on the **DeviceNet** side:

- These Parameters are entered into the Comment field of this dialogue box. The data format for all bytes in the **Comment** field is hexadecimal. To understand the information in this section you should be familiar with the DeviceNet network.

The following table shows what you have to enter for the module.

Module	entry to Comment field (Optional bytes are in []):
output only module	EPR_H, EPR_L, MACID, ConsLen [,txOffs [,allocation [, option]]]
input only module	EPR_H, EPR_L, MACID, ProdLen [,rxOffs [,allocation [, option]]]
input/output module	EPR_H, EPR_L, MACID, ConsLen, ProdLen , [,txOffs [, rxOffs [,allocation [, option]]]]
to configure the local slave	0, 0, MACID==SCANID, ConsLen, ProdLen , [,txOffs [, rxOffs]]

Table 5.3.1: Comment field

Every name is a placeholder for one comment byte. **Optional bytes are printed in []**. All optional bytes may not be specified, if the following optional bytes don't need to be specified. For instance if you want to specify the <allocation> byte for an output module, you must also specify the <txOffs>.

The length unit for the fields `ConsLen`, `ProdLen`, `txOffs` and `rxOffs` is always byte independent of the unit that has been chosen for the PLC side (i.e. PROFIBUS side) of the configuration.

`EPR_H`,

`EPR_L`:

Expected Package Rate High and Low byte in ms. The expected package rate sets the timeout duration on the DeviceNet network for the connection to this slave. The scanner of the gateway will poll remote slaves all $\frac{EPR}{2}$ ms for a polled or bit-strobed connection. Maximum allowed value for polled/bit-strobed connections is 0x7FFF (32.767s).

Maximum value for COS connections is 0x3FFE (16.383s).



INFORMATION

If you're using multiple bit-strobed slaves on the network they can have only one common expected package rate by principle. The EPR of the bit-strobed slave with the least MACID is used as EPR for all bit-strobed slaves. Individual settings for the other bit-strobed slaves are ignored.

`MACID`:

Medium ACcess IDentifier

The `MACID` of the slave which should be accessed. If it is the same as the gateway's `MACID`, it means that this slot is intended to control the local slave in the gateway.

`ConsLen`:

Consumed data Length

The number of output bytes in the DeviceNet modules output data block. This number is inherent to the selected type of connection and the slave module.

It has to be derived from the manufacturer's documentation or the EDS file that belongs to the DeviceNet slave!

Output direction is seen from the gateways point of view for a remote slave.

`ProdLen`:

Produced data Length

The number of input bytes in the DeviceNet modules input data block. This number is inherent to the selected type of connection and the slave module.

It has to be derived from the manufacturer's documentation or the EDS file that belongs to the DeviceNet slave!

Input direction is seen from the gateways point of view for a remote slave.

`txOffs`:

transmit Offset

The PLC output bytes are inserted at <txOffs> in the output data block of the DeviceNet module. Default: 0.

`rxOffs`: receive Offset
 The PLC input bytes are extracted at `<rxOffs>` from the input data block of the DeviceNet module. Default: 0.

`allocation`: DeviceNet allocation byte
 This byte selects the kind of connection that is maintained between the local scanner and the remote slave. It defaults to 0x03 and this means a polled connection.

Bit no.	7	6	5	4	3	2	1	0
Content	Reserved	Acknowledge Suppression	Cyclic	Change of State	Reserved	Bit Strobed	Polled	Explicit Message

Table 5.3.2: Allocation choice byte contents

The following choices for the allocation byte are possible:

- cyclic and/or change of state (COS) (0x20/0x10)
- bit strobed (0x04)
- polled (0x02)
- explicit connection only (0x01)

Specify only one of these. Additional selection of explicit message connection (0x01) is possible and is implicitly done by the gateway itself.

If you want to use the explicit connection from your PLC program you have to configure and use the communication window.

`option`: options
 At the moment only for internal test purposes.

The single output bit for each bit-strobed slave is in the current implementation always sent as a zero bit. This should not do any harm, because most of the bit-strobed slaves do not evaluate this output bit. They interpret the received output CAN frame only as production trigger.

5.4 Configuration Examples

5.4.1 Configuration Example with txOffs and rxOffs

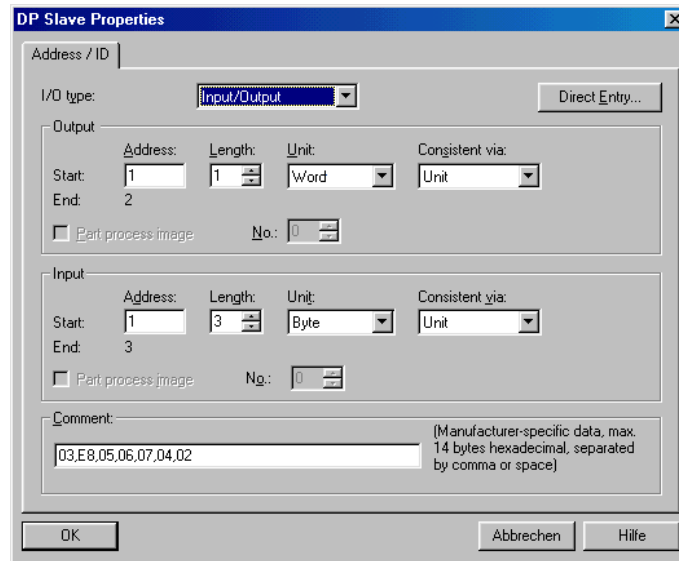


Fig. 5.5.1: DP Slave Properties of the example

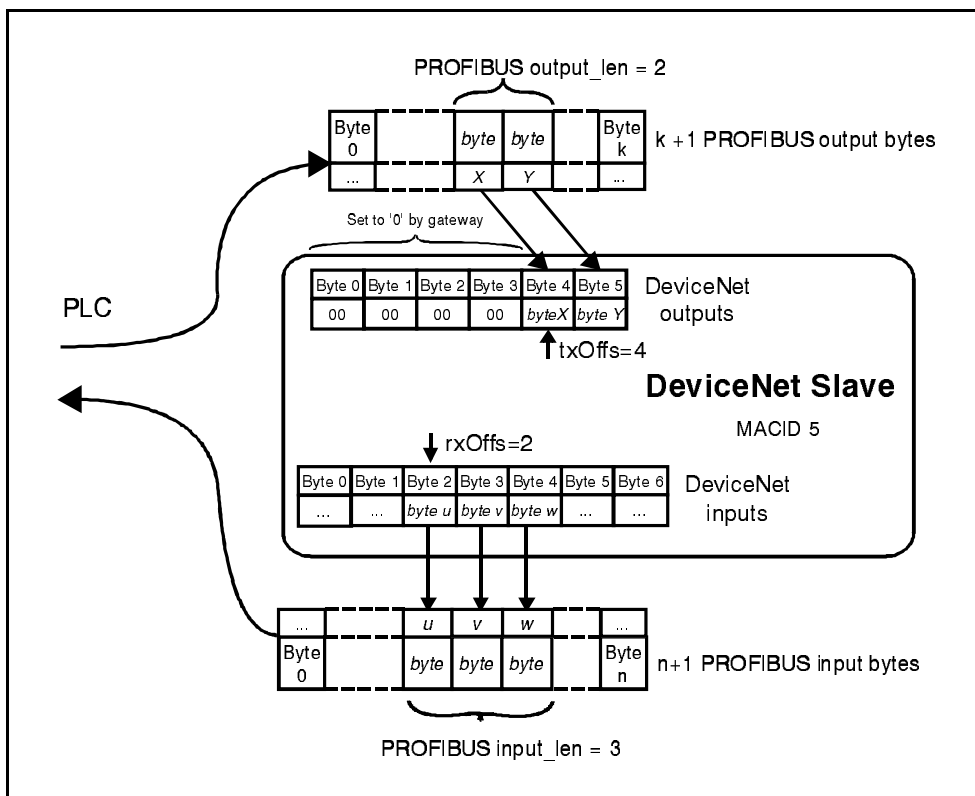


Fig. 5.5.2: Configuration example

The usage of `txOffs` and `rxOffs` is explained in this example. It makes sense if you only need to transfer some contiguous bytes to and from the DeviceNet module and not the whole data block. The usage of `txOffs` and/or `rxOffs` allows you to minimize the amount of data transferred over the

PROFIBUS. On the other hand keep in mind that the whole data block is transferred via CAN/DeviceNet to the DeviceNet module. In figure 5.5.2 you can see how the data are transferred, if you configure a slot like seen in Fig. 5.5.1.

The <output_len> output bytes from the PROFIBUS are inserted at `txOffs` in the output data block for the DeviceNet module. All bytes not written via the PROFIBUS default to zero.

For the input direction all data are read from the DeviceNet module, but only <input_len> bytes starting at `rxOffs` are transmitted over the PROFIBUS. All other bytes are discarded.

`ConsLen` as a synonym for consumed length and `ProdLen` as a synonym for produced length are the configuration values for the DeviceNet data connection. The meaning (direction) of consumed and produced length is always seen from the DeviceNet network's point of view. They always specify the whole amount of data transmitted over the DeviceNet. As these are DeviceNet terms the consumed length `ConsLen` corresponds to the output length as seen from the PLC's point of view for remote slaves.

5.4.2 Configuration Example for the Local Slave

The following example for a *Comment* field assumes that you have configured the DeviceNet gateway's MACID to 17 (dec.). Refer to Section 5.2, Step 3 for information how to determine the gateway's MACID.

In the *Comment* field for the local slave the fields `allocation` and `option` are not allowed, because the remote scanner decides the kind of connection to the local slave. Also the EPR values are "do not care" values because the remote scanner configures the EPR!

example <i>Comment</i> field for local slave	00, 00, 11, 0A, 02
---	--------------------

0x00 is used for the EPR fields as dummy placeholders.

This setup of the comment field tells the gateway to act as a slave on MACID 17 (0x11) for a remote scanner.

Its `ConsLen` is 0x0A, means it will accept ten bytes of input data from the remote scanner.

Its `ProdLen` is 0x02, means it will transmit two bytes of output data to the remote scanner.

The names for consumed length and produced length are corresponding to the DeviceNet's point of view, because these are DeviceNet terms. Therefore the meaning from the PLC's point of view is exactly reversed in comparison to the meaning for a configuration of a remote slave.

This may be confusing at first, but for judging the data direction from the terms consumed length (`ConsLen`) or produced length (`ProdLen`) you only need to look at the slaves from a DeviceNet scanner's point of view. On the other hand the meaning of `txOffs` and `rxOffs` is still the same for a local slave as it was for a remote slave, because "tx" and "rx" refer to the PLC's output and input point of view, respectively.

5.4.3 Configuration Example for a Remote Slave

You can configure a remote slave as a simple input device, using input only DP Slave properties.

example <i>Comment</i> field for bit-strobed slave , configured as an input only module	00, FA, 3F, 02, 00, 04
--	------------------------

Means for this slave EPR is 250 ms (0x00 FA), the MACID is 63 (0x3F), 2 Bytes input data and bit-strobed connection.

5.4.4 Restrictions for the *DN-gateway*

The *DN-gateway* can link any Profibus-DP master to a DeviceNet network. The *DN-gateway* operates as a Profibus-DP slave-I/O component. At the DeviceNet network the *DN-gateway* can operate as a scanner to control several DeviceNet slave devices or as a single slave device to be connected to one DeviceNet scanner. The *DN-gateway* is always configured via a Profibus-DP master. You can connect more than one *DN-gateway* to one Profibus Master. The Profibus specification states that it is possible to connect up to 124 Profibus-DP slave-I/O components.

All limitations mentioned have to be taken into consideration.

Configuration of the DN-gateway as a DeviceNet Scanner:

The maximum addressable number of DeviceNet slave devices depends on the configured Profibus-DP slave I/O type of each DeviceNet slave device. This limit is between 20 to 40 devices. The maximum number of Profibus DP configuration data bytes is 244 bytes. A configuration data block per external DeviceNet slave (slot) consists out of two or three bytes concerning the Profibus-DP configuration and up to 9 bytes concerning the *DN-gateway* configuration.

PROFIBUS-DP Slave I/O Type as Input/Output	
Profibus-DP	Manufacturer specific data
Octet 1	count of manufacturer specific data
Octet 2	length output bytes
Octet 3	length input bytes
Octet 4	EPR_H
Octet 5	EPR_L
Octet 6	MACID
Octet 7	ConsLen
Octet 8	ProdLen
Octet 9	txOffset
Octet 10	rxOffset
Octet 11	allocation
Octet 12	option

Example:

For the configuration of a DeviceNet slave device as an input without using the txOffset, allocation and option, the configuration frame would consist out off Octet 1, Octet 2, Octet 4, Octet 5, Octet 6 and Octet 8. In whole this are 6 bytes. If only such kind of devices are configured, the maximum configurable number of DeviceNet slave devices would be 244 bytes / 6 bytes = 40.

For the configuration of a DeviceNet slave device as an input and output with using the txOffset, rxOffset, allocation and option, the configuration frame would consist out off all Octets (1 to 12) In whole this are 12 bytes. If only such kind of devices are configured, the maximum configurable number of DeviceNet slave devices would be 244 bytes / 12 bytes = 20.

Please note that at least one output byte has to be configured.

In whole up to 312 payload data bytes can be configured for the whole gateway. The limit for each direction is 240 payload data bytes. Therefore the combinations of the max. number of input payload data bytes / max. number of output payload data bytes can be used:

Max. number of input payload data	Resulting max. number of output payload data	Max. number of payload data bytes
240	72	312
:	:	
156	156	
:	:	
72	240	

Configuration of the DN-gateway as one DeviceNet slave:

The *DN-gateway* can be configured as exactly one DeviceNet slave.

With this configuration, it can be connected to exactly one DeviceNet scanner. But with this configuration the *DN-gateway* can still access other DeviceNet slaves as a scanner!

Limitations due to the DN-gateway:

The min. Expected Package Rate is 16 ms (8 ms cycle)

Using the Communication Window:

The max. fragmentation length for Expl. Requests is 128 bytes.

Further information are available on request.

Limitations due to the DeviceNet specification:

The following baud rates can be chosen for the DeviceNet bus:

- 125 kbit/s
- 250 kbit/s
- 500 kbit/s

6. Manual Configuration of a DP-Master

The following description is based on the PROFIBUS-Specification-Normative-Part-8.

After power up a DP Master is sending 2 frames to each connected device in the network:

- parameter frame
- configuration frame

Accompanying to the explanations you will find a configuration example of a Module with the following parameters:

Module 1

- DeviceNet slave
- MACID = 63_d
- Produced data length = 7 byte
- Consumed data length = 7 byte
- Baudrate = 500 kBaud

6.1 Structure of the Parameter Frame

The structure of the user parameter data is described on page 27.

1. Byte	2. Byte	3. Byte
Always 00	Baudrate	own MACID

These 3 Bytes have to be implemented in the GSD file behind `User_Prm_Data`. Delete the semicolon (;) that defines this row as a comment.

Example for *Module 1*:

```

...
User_Prm_Data=0x00,0x02,0x10 ;    User-Parameter Data
...

```

With these settings the own MACID of the *DN-gateway* is adjusted to 16_d and the baud rate is set to 500 kBaud.

6.2 Structure of the Configuration Frame

The following description is based on the PROFIBUS-Specification-Normative-Part-8, chapter 9.3.5, page 734.

In contrast to common PROFIBUS slaves the data length of the *DN-gateway* is variable.

By using other DP Masters than Siemens SIMATIC, it is just valid to connect one *DN-gateway*. To declare a DeviceNet slave you must insert a module section in the GSD file. That is a section delimited by the keywords "Module=... End Module".

For the setup of the module section you need the following information about the connected DeviceNet devices:

- data length, input and /or output of each device
- MACID of each device
- Expected Package rate – EPR
- desired connection type (polled, COS)

One module configuration section has to be set up for each DeviceNet slave you want to communicate with.

On startup the DP-master assembles all module section's contents into the configuration frame that it transmits to the DP slave (*DN-gateway*)

The module section content is divided into *Octets* (PROFIBUS-Specification-Normative-Part-8, page 738, figure 16):

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<i>Octet 1</i>	count of manufacturer specific data						
<i>Octet 2</i>	length output bytes						
<i>Octet 3</i>	length input bytes						
<i>Octet 4</i>	manufacturer specific data						
<i>Octet 5</i>	manufacturer specific data						
etc.	manufacturer specific data						

Octet 1: Length of Manufacturer Specific Data

Because the *DN-gateway* uses always the special ID format to represent a connected DeviceNet device the identifier byte must have the following structure (see also PROFIBUS-Specification-Normative-Part-8, page 737):

	MSB Bit LSB							
	7	6	5	4	3	2	1	0
Content	00: free place 01: length byte for input follows 10: length byte for output follows 11: length byte for input/output follows (depending on the connected DeviceNet slave)		always 00		length of manufacturer specific data (Refer to page 30: Byte count of comment field.)			

Example Octet 1:

	MSB Bit LSB							
	7	6	5	4	3	2	1	0
Content	1	1	0	0	0	1	0	1

= **0xC5**

Module 1 is input/output (0xC0=) and
5 bytes manufacturer-specific data (0x05) will follow

Octet 2: Length Bytes of Output

Octet 2 gives the consistency, the structure (byte/word) and the number of the output bytes. Length bytes of the output as seen from the PROFIBUS master (see also PROFIBUS-Specification-Normative-Part-8, page 738)

	MSB							LSB														
Bit-No.:	7	6	5	4	3	2	1	0														
Content:	Consistency over 0: byte or word 1: complete length -> for <i>DN-gateway</i> do not care (write always to '1')	Length-format 0: byte-structure 1: word-structure	length bytes of inputs/outputs <table border="1" style="margin: 5px auto; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center;">Bit</th> <th style="text-align: center;">Meaning</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">5 4 3 2 1 0</td> <td></td> </tr> <tr> <td style="text-align: center;">0 0 0 0 0 0</td> <td>1 byte, resp. 1 word</td> </tr> <tr> <td style="text-align: center;">:</td> <td style="text-align: center;">:</td> </tr> <tr> <td style="text-align: center;">1 1 1 1 1 1</td> <td>63 bytes, resp. 63 words</td> </tr> </tbody> </table>						Bit	Meaning	5 4 3 2 1 0		0 0 0 0 0 0	1 byte, resp. 1 word	:	:	1 1 1 1 1 1	63 bytes, resp. 63 words				
Bit	Meaning																					
5 4 3 2 1 0																						
0 0 0 0 0 0	1 byte, resp. 1 word																					
:	:																					
1 1 1 1 1 1	63 bytes, resp. 63 words																					

Example Octet 2:

	MSB							LSB
Content	7	6	5	4	3	2	1	0
Content	1	0	0	0	0	1	1	0

= **0x86** consistency over hole length (0x80), the length format is byte-structure and 7 bytes output data are transferred (0x06 = 7 ... 1)

Octet 3: Length Bytes of Input

Length byte of the input seen from the PROFIBUS master. The bit assignment of *Octet 3* is similar to *Octet 2*.

Example Octet 3:

	MSB							LSB
Content	7	6	5	4	3	2	1	0
Content	1	0	0	0	0	1	1	0

= **0x86** consistency over hole length (0x80), the length format is byte-structure and 7 bytes input data are transferred (0x06 = 7...1)

Octett 4 and higher: Manufacturer Specific Data

Configuration of the DeviceNet side. Refer to page 29. The octets from octets 4 and higher are contained in the comment field of the SIMATIC manager.

Implement the bytes of the comment field similar as described for *Octet 4* and the following bytes.

Example Module 1:

Octet 4: 0x03

Octet 5: 0xE8

0x3E8 is the expected package rate. The decimal value is 1000 msec. With this setting the DeviceNet scanner is polling this slave each second.

Octet 6: 0x3F MACID = 63 dec

Octet 7: 0x07 consumed data length, 7 bytes

Octet 8: 0x07 produced data length, 7 bytes

In this example the length of the manufacturer specific data is *Octet 4* to *Octet 8* resulting in 5 Bytes.

The configuration frame segment for the Module 1 has the following structure:

0xC5	0x86	0x86	0x03	0xE8	0x3F	0x07	0x07
------	------	------	------	------	------	------	------

The configuration frame segment and has to be inserted into the GSD file as shown below:

```

...
Module="Module 1" 0xC5,0x86,0x86,0x03,0xE8,0x3F,0x07,0x07
EndModule
...

```

6.3 The Communication Window

Documentation provided on request.

7. DeviceNet Errata PROFIBUS-DP to DeviceNet Gateways

The DN-CBM-DP and the DN-DP have not been presented to an official ODVA test lab to apply for a conformance certificate. However the gateway's protocol implementation of the DeviceNet slave has been checked with the conformance test tool (CTT) that official ODVA test labs use while doing the conformance test for a DeviceNet appliance. The used release of the CTT was "DeviceNet Protocol Conformance Revision A22, November 2010".

During the tests some deviating behaviour was observed that is described in the following paragraphs:

7.1 Master/Scanner: The UCMM loses Message Ids

The scanner needs to allocate a message ID for the UCMM connection when it starts talking to an external slave. If the external slave doesn't answer and in that state the connection is closed then the UCMM Connection ID may be locked and the connection ID is lost for later connection attempts. There is only the limited amount of 16 connection IDs available.

The connections are closed when you reconfigure the PROFIBUS-DP master. This in turn solicits a new DP parametrization and configuration frame making a reconfiguration of the DeviceNet side necessary. It is only a minor problem if you only need to talk to a fixed list of (missing) slaves and don't reconfigure the PROFIBUS-DP master because as soon as a slave answers its connection ID is returned to the pool and reused (normal operation in the field).

This may become a problem if you're changing the hardware configuration of your PLC often during the development phase **and** have missing slaves on the DeviceNet bus. A power cycle for the gateway solves that issue.

This error does only affect the master/scanner side and is not tested in ODVA's conformance test.

7.2 Master/Scanner: CoS Allocation is incorrect

When using the CoS connection only, conflicts with the required behaviour of the slave occur. In particular a single CoS connection can not be chosen.

The allocation byte is determined by the configuration of the external DeviceNet slave via the "DP Slave Properties" window (see chapter 5.3).

The behaviour for the allocation of a CoS/Cyclic connection together with a Polled connection is described for the slave side in the DeviceNet specification Volume 3, Chapter 3.13 "Change of State/Cyclic Messages". In particular the behaviour, in the case that both connections (CoS/Cyclic & Polled) are not enabled using a single allocation request, is precisely specified.

The following deviations from the specification occur:

- If you use an allocation byte of 0x11 (CoS only & Explicit), the scanner will implicitly switch on the Polled connection if the consumed length of the external slave is unequal zero (0). According to the standard, the scanner should only use the CoS connection and leave the Polled connection unused. Therefore the gateway's scanner can not use the CoS connection as an input only connection.
- If you use an allocation byte of 0x13 (CoS & Polled & Explicit), the scanner will wrongly send an allocation request with a choice byte of 0x01 (Explicit) and then 0x02 (Polled Only).

**NOTICE**

To avoid these errors do not use the allocation bytes 0x11 (CoS only & Explicit) with input only modules or 0x13 (CoS & Polled & Explicit) when configuring an external slave.

7.3 Slave/Device: Network Status is not always indicated correctly

During the conformance test the network status is not always indicated correctly by the Status LED (especially if only UCMM connections do exist).

The DeviceNet object test of the CTT performs a Network Status LED test, because a combined Module/Network Status LED is provided. The Network Status LED test applies also to the combined LED, because the CTT does only test the interaction on the network. The states of the hardware, which are to be additionally indicated by the combined LED, do not occur during the test and consequently do not affect the display of the combined LED.

When the state machine is run though, the blinking codes of the combined DeviceNet Module/Network Status LED do not indicate the DeviceNet network states correctly.

For example, the slave should switch the LED to light permanently green already, if a connection is dynamically established via UCMM, but the LED remains blinking green.

7.4 Network Access State Machine after communication failure

This problem occurs in the communication failure state (for example after a CAN bus-off). It applies only to the DN-DP that can detect the 24 V power supply voltage of the DeviceNet interface. When the power is removed and then supplied again in that state, the Network Access State Machine should restart, carry out a Duplicate MACID Check and leave the communication failure state on Duplicate MACID Check success. But this doesn't happen.

**INFORMATION**

To leave the communication failure state, unplug the connector on the PROFIBUS side and afterward connect it again. This will restart the PROFIBUS communication and in turn the DeviceNet network state machine.

7.5 Network Status on DeviceNet power loss

The network status at loss of the 24V DeviceNet power supply voltage is not indicated correctly. The LED is not switched to blinking red if the power supply voltage on the DeviceNet bus is not supplied. The DN-CBM-DP can't even detect the state of the 24V DeviceNet power supply.