



**Deuschmann Automation**

Cam Controls | Fieldbus Gateways | Industrial Ethernet Products

**User manual  
Universal Fieldbus-Gateway  
UNIGATE®**



**UNIGATE RS232/485 - CANopen®  
UNIGATE RS232/422 - CANopen®  
UNIGATE SC232/485 - CANopen®  
UNIGATE SC232/422 - CANopen®**

V2700E

Deuschmann Automation GmbH & Co. KG Carl-Zeiss-Str. 8 D-65520 Bad Camberg  
Tel:+49-(0)6434-9433-0 Hotline: +49-(0)6434-9433-33 Fax: +49-(0)6434-9433-40  
Internet: <http://www.deuschmann.de>



<b>1</b>	<b>Information on CE marking of the module</b>	<b>8</b>
1.1.	EU Directive EMC	8
1.2.	Scope of application	8
1.3.	Note installation guidelines	8
1.4.	Installation of the unit	8
1.5.	Working on switch cabinets	8
<b>2</b>	<b>Information for the machine manufacturer</b>	<b>9</b>
2.1.	Introduction	9
2.2.	EU Machinery Directive	9
<b>3</b>	<b>Introduction</b>	<b>10</b>
3.1.	UNIGATE RS software flow-chart	11
3.2.	UNIGATE SC software flow chart	12
3.3.	UNIGATE block diagram	13
3.4.	UNIGATE application diagram	13
<b>4</b>	<b>Operation modes of the gateway</b>	<b>14</b>
4.1.	Configuration mode (config mode)	14
4.2.	The Debug cable for UNIGATE SC	15
4.3.	Test mode	15
4.4.	Data exchange mode	16
<b>5</b>	<b>RS-interface</b>	<b>17</b>
5.1.	Framing Check - only for UNIGATE RS	17
5.2.	RS-interface at the UNIGATE SC	17
<b>6</b>	<b>Mode of operation of the system</b>	<b>18</b>
6.1.	General explanation	18
6.2.	Interfaces	18
6.3.	Data exchange CANopen®	18
6.3.1	SDO-access	18
6.3.2	PDO-access	18
6.3.3	Possible data lengths	18
6.4.	Data exchange RS232/RS485	19
6.5.	The trigger byte -only at UNIGATE RS	19
6.6.	The length byte - only at UNIGATE RS	19
<b>7</b>	<b>Transparent mode - only at UNIGATE RS</b>	<b>20</b>
<b>8</b>	<b>Character-Delay-Time mode - only at UNIGATE RS</b>	<b>21</b>
8.1.	Receive direction	21
<b>9</b>	<b>Implemented protocols in UNIGATE RS</b>	<b>22</b>
9.1.	Protocol ARCNET	22
9.1.1	Data exchange	22
9.1.1.1	Broadcast	22
9.1.2	Data structure	22
9.1.2.1	Trigger byte	22
9.1.2.2	Length byte	22
9.1.2.3	ARCNET-ID	22
9.1.2.4	Data area	22

9.1.3	ARCNET configuration . . . . .	22
9.1.3.1	ARCNET-ID of the UNIGATE . . . . .	23
9.2.	Universal 232 protocol . . . . .	23
9.2.1	Data structure . . . . .	23
9.2.2	Fieldbus parameters . . . . .	23
9.2.3	RS232 parameter table . . . . .	23
9.2.3.1	Start character . . . . .	23
9.2.3.2	Length 232 . . . . .	23
9.2.3.3	ID . . . . .	24
9.2.3.4	Data area . . . . .	24
9.2.3.5	Checksum . . . . .	24
9.2.3.6	End character . . . . .	24
9.2.4	Communication sequence . . . . .	24
9.3.	The 3964 R protocol . . . . .	24
9.3.1	Data structure 3964R . . . . .	25
9.3.2	Protocol definitions . . . . .	25
9.3.3	Data communication . . . . .	25
9.3.3.1	Initiation of data communication by the low-priority user . . . . .	25
9.3.3.2	Conflicts . . . . .	25
9.3.3.3	Timeout times . . . . .	25
9.3.3.4	Retries . . . . .	26
9.3.3.5	Initiation of data communication by the high-priority user . . . . .	26
9.3.4	State of the 3964R communication . . . . .	26
9.4.	RK512 . . . . .	26
9.4.1	Data exchange . . . . .	26
9.5.	The fieldbus monitor protocol . . . . .	27
9.5.1	Function . . . . .	27
9.6.	Modbus-RTU . . . . .	27
9.6.1	Notes . . . . .	27
9.6.2	UNIGATE as Modbus-master . . . . .	28
9.6.2.1	Preparation . . . . .	28
9.6.2.2	Data structure . . . . .	28
9.6.2.3	Communication sequence . . . . .	29
9.6.3	UNIGATE as Modbus-slave . . . . .	29
9.6.3.1	Preparation . . . . .	29
9.6.3.2	Data structure . . . . .	29
9.6.3.3	Communication sequence . . . . .	29
9.6.3.4	Status report . . . . .	30
<b>10</b>	<b>Generating a script - only for UNIGATE SC . . . . .</b>	<b>31</b>
10.1.	What is a script? . . . . .	31
10.2.	Memory efficiency of the programs . . . . .	31
10.3.	What can you do with a script device? . . . . .	31
10.4.	Independence of buses . . . . .	31
10.5.	Further settings at the gateway . . . . .	31
10.6.	The use of the Protocol Developer . . . . .	32
10.7.	Accuracies of the baud rates at UNIGATE SC . . . . .	32

10.8. Script processing times . . . . .	33
<b>11 Hardware ports, switches and LEDs . . . . .</b>	<b>34</b>
11.1. Drawing of the unit . . . . .	34
11.1.1 Model UNIGATE RS / SC 232/485-CANopen . . . . .	34
11.1.2 Model UNIGATE RS / SC 232/422-CANopen . . . . .	35
11.2. Configuration of the UNIGATE RS . . . . .	35
11.2.1 CANopen . . . . .	35
11.2.2 RS232/RS485/RS422 . . . . .	35
11.3. Connectors . . . . .	36
11.3.1 Connector to the external device (RS-interface) . . . . .	36
11.3.2 CANopen® connector . . . . .	37
11.3.3 Debug plug . . . . .	37
11.3.4 Power supply . . . . .	37
11.3.5 Shield terminal lead . . . . .	37
11.4. LEDs . . . . .	37
11.4.1 LED "Bus Power" . . . . .	38
11.4.2 LED "Bus State" . . . . .	38
11.4.3 LED "Power" . . . . .	38
11.4.4 LED "State" . . . . .	38
11.4.5 LEDs "Error No. / Select ID" at UNIGATE RS . . . . .	38
11.5. Switches . . . . .	38
11.5.1 DIP-switch . . . . .	38
11.5.2 Slide switch Termination CANopen . . . . .	39
11.5.3 Rotary coding switches S4 + S5 (RS485 ID) . . . . .	39
11.5.4 Slide switch (RS485/RS232 interface) . . . . .	39
11.5.5 Slide switch (RS485/RS422 termination) . . . . .	39
<b>12 Error handling . . . . .</b>	<b>40</b>
12.1. Error handling at UNIGATE RS . . . . .	40
12.2. Error handling at UNIGATE SC . . . . .	41
<b>13 Housing variations . . . . .</b>	<b>42</b>
13.1. Main operational fields for housings . . . . .	42
13.2. Mounting instruction . . . . .	42
<b>14 Installation guidelines . . . . .</b>	<b>44</b>
14.1. Installation of the module . . . . .	44
14.1.1 Mounting . . . . .	44
14.1.2 Removal . . . . .	44
14.2. Wiring . . . . .	44
14.2.1 Connection systems . . . . .	44
14.2.2 CANopen® communication interface . . . . .	45
14.2.2.1 Bus line with copper cable . . . . .	45
14.2.2.2 Power supply . . . . .	45
14.2.2.3 Shield connection . . . . .	45
14.2.2.4 Equipotential bonding connection . . . . .	45
14.2.3 Line routing, shield and measures to combat interference voltage . . . . .	45

14.2.4	General information on line routing . . . . .	45
14.2.4.1	Shielding of lines . . . . .	46
<b>15</b>	<b>Technical data . . . . .</b>	<b>48</b>
15.1.	Device data . . . . .	48
15.1.1	Interface data . . . . .	50
15.2.	Housing data . . . . .	51
<b>16</b>	<b>Commissioning guide . . . . .</b>	<b>52</b>
16.1.	Note . . . . .	52
16.2.	Components . . . . .	52
16.3.	Installation . . . . .	52
16.4.	Dimensional drawing DIN-rail mounting . . . . .	52
16.5.	Commissioning . . . . .	52
16.6.	Setting the CANopen® address and Baud rate . . . . .	52
16.7.	CANopen® connection . . . . .	53
16.8.	Connection to the process device . . . . .	53
16.9.	Connecting the supply voltage . . . . .	53
16.10.	Shield connection . . . . .	53
<b>17</b>	<b>Servicing . . . . .</b>	<b>54</b>
17.1.	Returning a device . . . . .	54
17.2.	Downloading PC software and GSD files etc. . . . .	54
<b>18</b>	<b>Annex . . . . .</b>	<b>55</b>
18.1.	Explanations of the abbreviations . . . . .	55
18.2.	Hexadecimal table . . . . .	56
<b>19</b>	<b>Drilling templates. . . . .</b>	<b>57</b>
19.1.	Aluminium housing . . . . .	57
19.2.	High-grade steel housing . . . . .	58
19.3.	Polycarbonate housing . . . . .	59
<b>20</b>	<b>Declaration of conformity and certificates . . . . .</b>	<b>60</b>
20.1.	EG Certificate of Conformance . . . . .	60
20.2.	Certificate Germanischer Lloyd . . . . .	61

Disclaimer of liability

We have checked the contents of the document for conformity with the hardware and software described. Nevertheless, we are unable to preclude the possibility of deviations so that we are unable to assume warranty for full compliance. The information given in the publication is, however, reviewed regularly. Necessary amendments are incorporated in the following editions. We would be pleased to receive any improvement proposals which you may have.

Copyright

Copyright (C) Deutschmann Automation GmbH & Co. KG 1997 – 2011. All rights reserved. This document may not be passed on nor duplicated, nor may its contents be used or disclosed unless expressly permitted. Violations of this clause will necessarily lead to compensation in damages. All rights reserved, in particular rights of granting of patents or registration of utility-model patents.

Art.-No.: V2700E

# 1 Information on CE marking of the module

## 1.1 EU Directive EMC

The following applies to the module described in this User Manual:

Products which bear the CE mark comply with the requirements of EU Directive „Electromagnetic Compatibility“ and the harmonized European Standards (EN) listed therein.

The EU Declarations of Conformity are available at the following location for perusal by the responsible authorities in accordance with the EU Directive, Article 10:

Deutschmann Automation GmbH & Co. KG, Carl-Zeiss-Straße 8, 65520 Bad Camberg, Germany

## 1.2 Scope of application

The modules are designed for use in the industrial sector and comply with the following requirements

Scope of application	Requirement applicable to	
	Emitted interference	Interference immunity
Industry	EN 55011 Kl. A	EN 61000-6-2

## 1.3 Note installation guidelines

The module complies with the requirements if you

1. comply with the installation guidelines described in the User Manual when installing and operating the module.
2. also follow the rules below on installation of the equipment and on working on switch cabinets.

## 1.4 Installation of the unit

Modules must be installed in electrical equipment rooms/areas or in enclosed housings (e.g. switch boxes made of metal or plastic). Moreover, you must earth the unit and the switch box (metal box) or at least the top-hat rail (plastic box) onto which the module has been snapped.

## 1.5 Working on switch cabinets

In order to protect the modules against static electrical discharge, the personnel must discharge themselves electrostatically before opening switch cabinets or switch boxes.



## **2 Information for the machine manufacturer**

### **2.1 Introduction**

The UNIGATE CANopen® module does not constitute a machine as defined by the EU "Machinery" Directive. Consequently, the module does not have a Declaration of Conformity in relation to the EU Machinery Directive.

### **2.2 EU Machinery Directive**

The EU Machinery Directive stipulates the requirements applicable to a machine. The term "machine" is taken to mean a totality of connected parts or fixtures (see also EN 292-1, Paragraph 3.1)

The module is a part of the electrical equipment of the machine and must thus be included by the machine manufacturer in the Declaration of Conformity process.

### 3 Introduction

The UNIGATE RS232/RS485-CANopen® or the SC232/SC485-CANopen® module serves to adapt a serial port to the CANopen® according to CiA® DS 301. In this application, it functions as a gateway and operates as the CANopen® slave. It can be operated by any standard-compliant master.

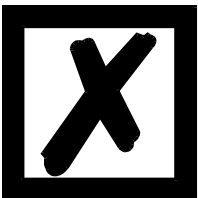
At the UNIGATE RS232/485 various transmission protocols are supported at the serial port:

- Modbus-RTU
- The 3964R protocol popular in Siemens equipment
- Monitoring on the basis of character delay time
- Start / End character
- Constant data lengths (Transparent mode)
- Customized protocols

These protocol variants will probably support most applications.

The module RS or SC essentially consists of the following hardware components:

- Electrically isolated interface to the CANopen
- CAN-controller SJA 1000
- Microprocessor T 89C51RD2
- RAM and EPROM
- Optionally electrically isolated RS-interface
- Serial port (RS232 and RS485) to the device connected externally

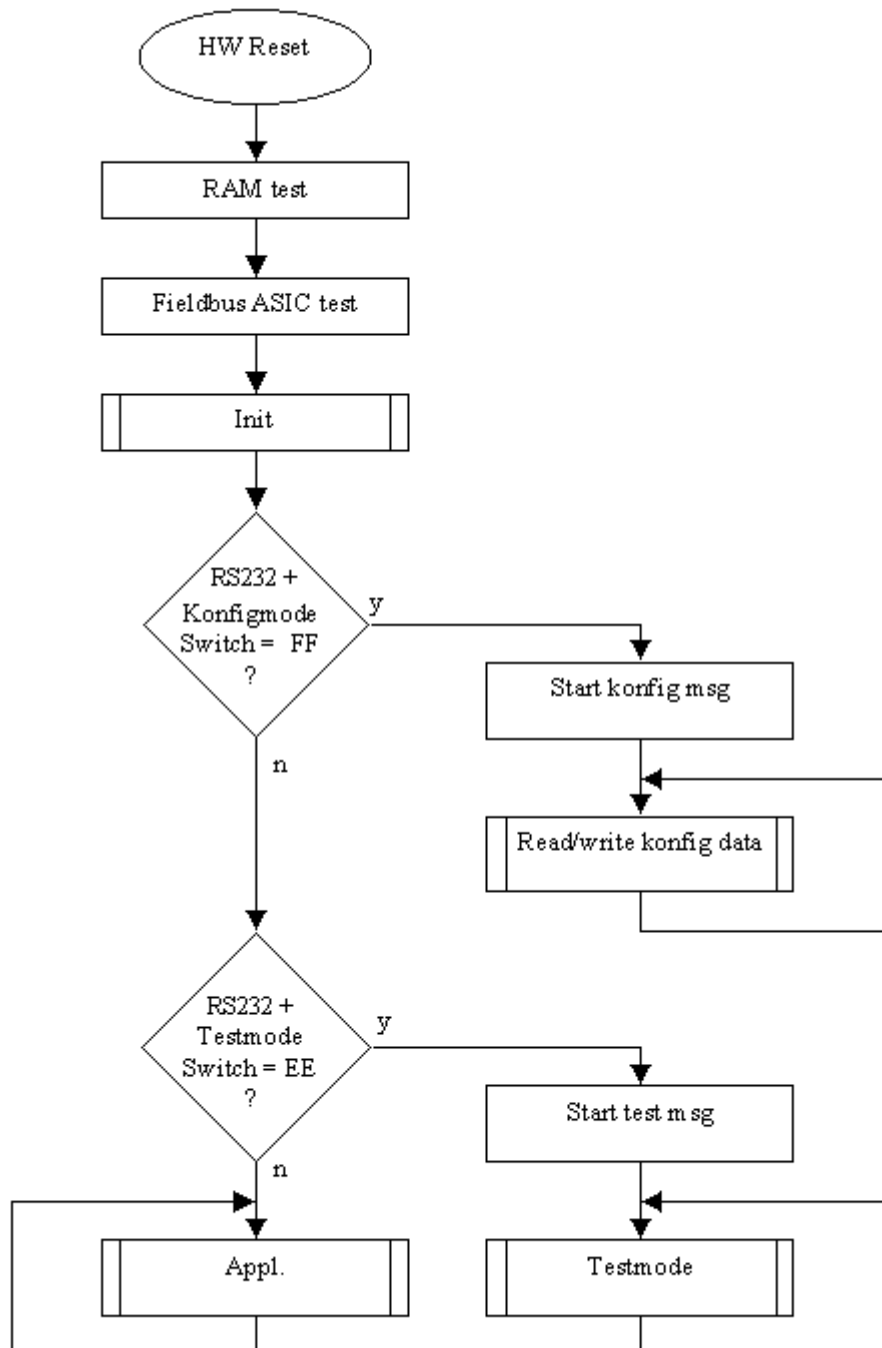


**Please note: This instruction manual is valid for UNIGATE RS (standard gateway with implemented protocols) and also for UNIGATE SC (gateway capable for scripts).**

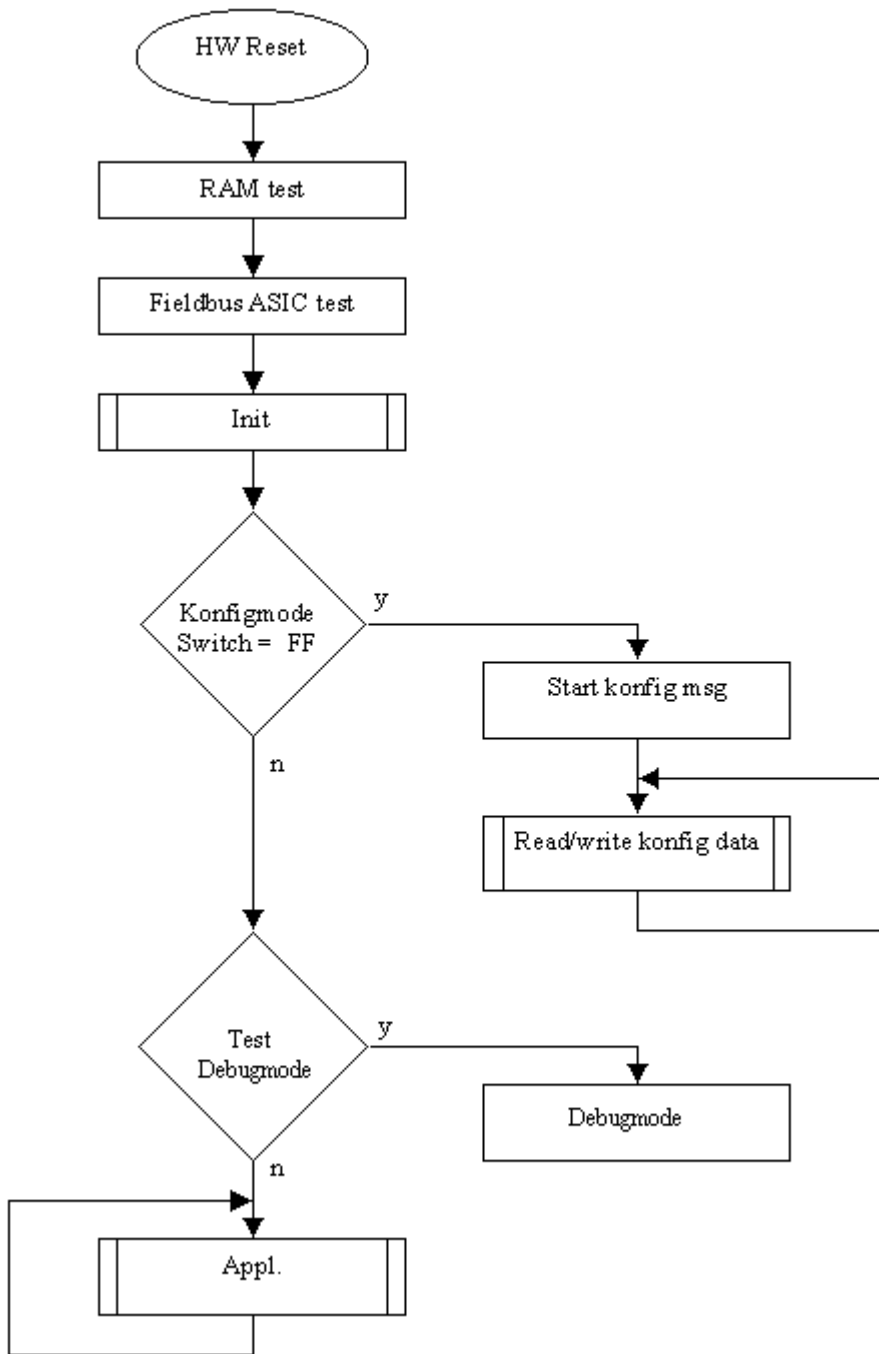
**The information given in this instruction manual is about the UNIGATE RS and the UNIGATE SC as well, unless differences are explicitly pointed out.**

### 3.1 UNIGATE RS software flow-chart

The following graph shows a typical application flow of a UNIGATE fieldbus-module.

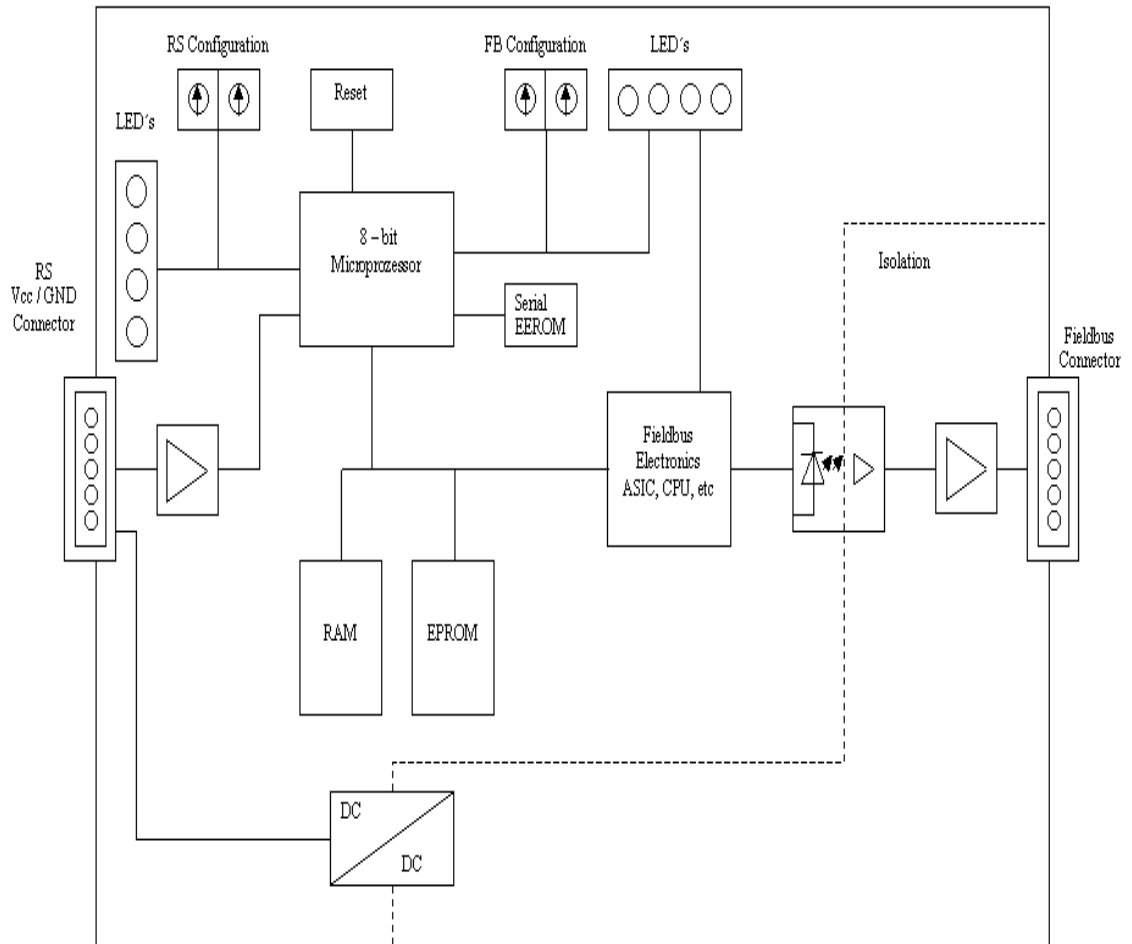


### 3.2 UNIGATE SC software flow chart



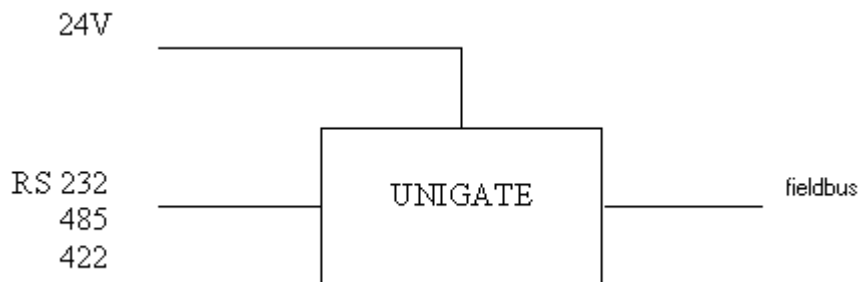
### 3.3 UNIGATE block diagram

The following graph shows a typical UNIGATE module-design.



### 3.4 UNIGATE application diagram

The following graph shows a typical connection scheme.



## 4 Operation modes of the gateway

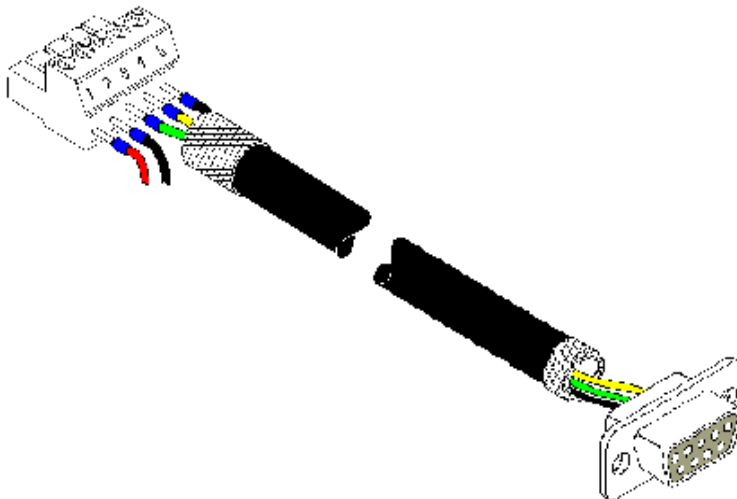
### 4.1 Configuration mode (config mode)

The configuration mode serves to configure the gateway. Adjustments at the gateway's configuration are only possible in this mode. The gateway will be starting in this mode in case both switches S4 as well as S5 are set on position "F" when switching on the gateway and also RS232 is to be selected as interface. Right after switching on the gateway in the configuration mode it will be sending its starting message, that looks analog with the following message: "RS-CO c(dA) switch=0x0000 Prot=0x00 SN=20790120".

In the configuration mode the gateway always operates with the settings 9600 Bauds, no Parity, 8 Databits and 1 Stopbit, the RS-State LED will always be flashing red/green, the "Error No/Select ID" LEDs are of no account for the user. All software revisions contain the configuration mode.

#### Connection cable from the gateway to the PC

The cable is supposed to look as follows:



#### Connection table 5pin screw-plug connector - PC

Screw-plug-connector	Name	D-Sub plug-connector	Name
Pin 3	Rx	Pin 3	Tx
Pin 4	Tx	Pin 2	Rx
Pin 5	GND	Pin 5	GND

## 4.2 The Debug cable for UNIGATE SC

The Debug cable consists of the following components: a 9-pole D-sub-socket with two exits, where respectively two cables are connected, one with a 3-pole Phoenix socket with mating connector and one with a 9-pole D-SUB plug.

**Table 1 (RS-product-side) 3-pole screw-plug connector**

Pin number	Color	Meaning
1	white	Rx-data
2	green	Tx-data
3	brown	Gnd

**D-sub (gateway RS-side), 1. RS-interface**

Pin number	Color	Meaning
2	white	Rx-data
3	green	Tx-data
5	brown	Gnd

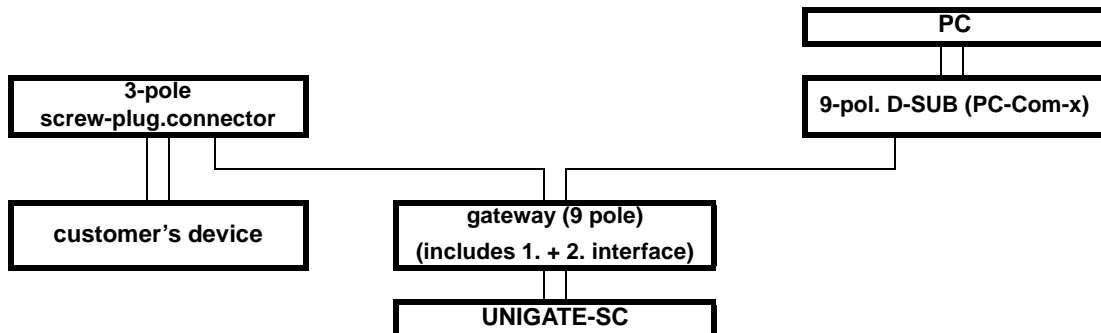
**Table 2 (PC-Com-x) 9-pole D-sub**

Pin number	Color	Meaning
2	white	Tx-diagnosis
3	green	Rx-diagnosis
5	brown	Gnd

**9-pole D-sub (gateway RS-side), 2. RS-interface**

Pin number	Color	Meaning
4	white	Tx-diagnosis
9	green	Rx-diagnosis
5	brown	Gnd

**View of the connector**



## 4.3 Test mode

### Setting of the test mode

The test mode is set by bringing the switches S4 and S5 in position "E". Beyond it the interface switch has to be set on "232". All other switches will not be taken into consideration for the setting of the test mode. Now the gateway has to be restarted with these settings (by a short disconnection from the power supply).

The test mode is contained from software revision V2.1 on. The test mode may be helpful to integrate the gateway in the relevant environment, for instance to "see" permanent changing data of the fieldbus in the SPS or also in order to test the parameters of the RS-interface.

**Mode of operation of the test mode**

After the restart in the test mode the gateway will be sending the values 0-15 in hexadecimal representation ("0".."F") in ASCII-coding with the current settings for the Baud rate, Parity, Start-, Data-, and Stopbits on the serial side every second. Simultaneously the same values are issued binary on the fieldbus-interface provided that this is possible on the fieldbus at the moment (In case the fieldbus has a data width of more than 1 byte, all characters of the fieldbus will be set to the current test character).

In this mode the State-LED on the RS-side will be flashing red/green, the "Error No/Select ID" LEDs will be displaying the value in a binary way, that is issued that moment. Additionally each character that is received at one of the interfaces will also be output at the same interface as a local echo. On the fieldbus-side only the first byte will be used for the local echo, that means on receiving as well as on transmitting only the first byte of the bus data is looked at, the other bus data do not change compared to the last data.

**4.4 Data exchange mode**

The gateway has to be in the data exchange mode, so that a data exchange between the RS-side of the gateway and the fieldbus is possible. As long as the gateway is not in the configuration mode or the test mode, the data exchange mode is active. In the data exchange mode the gateway will carry out the set protocol with those parameters preset through WINGATE.



## 5 RS-interface

### 5.1 Framing Check - only for UNIGATE RS

The length of the stop bit received by the gateway is checked through the function "Framing Check" from the software V 2.1 on. Here the stop bit generated by the gateway is always long enough, so that connected participants can evaluate the stop bit.

Please be aware that the function "Framing Check" becomes effective only in case of 8 data bit and the setting "No parity".

An error is detected and indicated by the Error LEDs in case the stop bit does not show the length 1 bit during the activated check.

The possible settings for this parameter are "enabled" and "disabled". The presetting for the "Stop Bit Framing Check" is "enabled".

### 5.2 RS-interface at the UNIGATE SC

On principle the hardware cannot be distinguished from the standard gateway. In addition to the regular hardware, a special hardware version is available, which however, is only required for the generation of a script. Due to technical reasons this advanced hardware is not available for all buses; a development can be made on another than the target hardware though.

Compared to the standard gateway, this development gateway is equipped with an extra-interface RS232, which however, is only available to the outside at the version with 9-pole D-SUB. This debug-interface itself is always operated with 9600 baud, no parity, 8 data bits and 1 stop bit. Apart from that there are no further differences, neither in the software nor in the hardware.

## 6 Mode of operation of the system

### 6.1 General explanation

Communication can be split into seven layers, Layer 1 to Layer 7, in accordance with the ISO/OSI Model.

The DEUTSCHMANN AUTOMATION gateways convert Layers 1 and 2 of the customized bus system (RS485 / RS232) to the corresponding fieldbus system. Layers 3 to 6 are blank, and Layer 7 is forwarded transparently on the standard gateways. However, customized adaptations are also possible here (e.g. adaptations to existing profiles of the fieldbus systems).

The gateway can be configured through the software WINGATE<sup>®</sup> that is also supplied (see also chapter 11.2).

### 6.2 Interfaces

The gateway features the RS232 and RS485 interfaces. Switchover is performed by means of a slide switch accessible for the customer. The CANopen<sup>®</sup> gateway thus allows access to all devices connected to the RS485 bus via one single CANopen<sup>®</sup> Node-ID resp. access to the device connected to the RS232 interface.

### 6.3 Data exchange CANopen<sup>®</sup>

The following three objects are existing in the CANopen-gateway for the data exchange on the CANopen-side:

- Adr. 2000H (Type DOMAIN):Data received by the gateway
- Adr. 2001H (Type DOMAIN):Data sent by the gateway
- Adr. 2002H (Type BYTE):Length of the data sent

The length of the receiving- and transmitting buffer (Obj. 2000 + 2001) is configured through WINGATE<sup>®</sup>.

#### 6.3.1 SDO-access

Generally the data can always be exchanged through SDO's (Obj. 2000 - 2002).

Likewise an access to all Mandatory-objects according to CiA<sup>®</sup> DS 301 is possible through SDO's.

#### 6.3.2 PDO-access

PDO's are supported according to the following table depending on the configured length and the PDO-length is set dynamically to the correct value:

Gateway receiving-data	Gateway transmitting-data	Receiv.-PDO1 (Adr = 512 + ID)	Transm.-PDO1 (Adr = 384 + ID)
Max. 8 Byte	Max. 8 Byte	Receiv. data	Transm. data
Max. 8 Byte	>8 Byte	Receiv. data	Length transmitting data
>8 Byte	Max. 8 Byte	-	Transm. data
>8 Byte	>8 Byte	-	Length transmitting data

#### 6.3.3 Possible data lengths

The table below shows the maximum transferable data in CANopen:

Input data	max. 255 bytes	Variable: maximum value in this case
Output data	max. 255 bytes	Variable: maximum value in this case
Emergency data	1 byte	See chapter Error handling

## 6.4 Data exchange RS232/RS485

Data exchange via the RS interface can be programmed as follows:

- Cyclic
- Only if input data changes
- Only if trigger byte changes

All data is transferred consistently by the gateway in both directions.

### Structure of the data:

Trigger byte	See "The trigger byte -only at UNIGATE RS", chapter 6.5, on page 19
Length byte	See "The length byte - only at UNIGATE RS", chapter 6.6, on page 19
User data	

Trigger byte and length byte are contained only if configured accordingly.

## 6.5 The trigger byte -only at UNIGATE RS

In case data is transferred cyclically via CANopen® (normally that is not the case), the gateway must detect when the user wishes to send new data via the serial port. For this reason, the user can set control of transmission via a trigger byte. In this mode, the gateway always sends (and only sends) when the trigger byte is changed.

If Trigger-Byte mode is activated, the gateway increments the trigger byte each time a telegram has been received.

The first byte in the CANopen® input/output data buffer is used as the trigger byte if this mode is activated.

## 6.6 The length byte - only at UNIGATE RS

The user can configure whether the transmit length is also to be stored as a byte in the input/output data area. In transmit direction, as many bytes as specified in this byte are sent. On reception of a telegram, the gateway enters the number of characters received.

## 7 Transparent mode - only at UNIGATE RS

The data is transferred transparently with no timeout monitoring functions whatsoever. The user presets the data on the master. This data is then forwarded to the external device with no timeout monitoring functions whatsoever. A response from the external device is also forwarded to the master with no changes. Any security mechanisms must be encrypted by the user in the net data.

## **8 Character-Delay-Time mode - only at UNIGATE RS**

The 'Character-delay-time mode' is not a protocol but a special variant of the "Universal 232" protocol' (see corresponding chapter).

### **8.1 Receive direction**

Data received is entered in the transmit buffer until the character delay time expires after reception of the last character. This time can be configured.

Transmit direction: The data is transmitted transparently.

## 9 Implemented protocols in UNIGATE RS

### 9.1 Protocol ARCNET

ARCNET stands for Attached Resource Computer NETwork.

It belongs to the group of Token-Bus-Networks. ARCNET is based on a multi master concept, that means every ARCNET participant is in the position to communicate with every other ARCNET participant without restrictions.

#### 9.1.1 Data exchange

This protocol allows a connection between ARCNET and CANopen.

With it those data, sent to the UNIGATE via CANopen, are passed on via ARCNET in accordance with the data structure described below. Here only those data packets are included that have been sent to the ARCNET-ID of the UNIGATE.

Data, the gateway receives via ARCNET, are passed on correspondingly via CANopen. With it the parameters configurable with WINGATE (Trigger- and lengthbyte) are included.

The length of the receive buffers and transmission buffers is also configured via WINGATE.

The mapping of the data to SDO's and PDO's is described in chapter 6.3 (Data exchange CANopen®).

##### 9.1.1.1 Broadcast

In case the destination address equals „0“, the data are sent to each ARCNET participant.

#### 9.1.2 Data structure

ARCNET	-	-	-	Data area
CANopen	Triggerbyte (opt.)	Lengthbyte (opt.)	ARCNET-ID	Data area

##### 9.1.2.1 Trigger byte

See „The trigger byte -only at UNIGATE RS“, chapter 6.5.

##### 9.1.2.2 Length byte

See „The length byte - only at UNIGATE RS“, chapter 6.6.

##### 9.1.2.3 ARCNET-ID

In case the data are transmitted from UNIGATE via CANopen, this byte is the ID of the ARCNET sender.

In the opposite direction (data get to the UNIGATE via CANopen), here the CANopen-sender registers the ARCNET-ID to which the gateway is supposed to send the data.

##### 9.1.2.4 Data area

In this field the useful data are copied transparently from ARCNET to CANopen® and the other way round.

#### 9.1.3 ARCNET configuration

- Baud rate: 2.5 Mbaud
- Short packet (is determined by the entry fieldbus length)

### 9.1.3.1 ARCNET-ID of the UNIGATE

The gateway itself also requires an ARCNET-ID, in order being in the position to exchange data in the net. This ID is set through the rotary switch on the ARCNET side. Here permissible values are 1 to 254 (01H..FEH). The ID 0 is not permitted, as it is the broadcast-address. 255 is only used for the configuration.



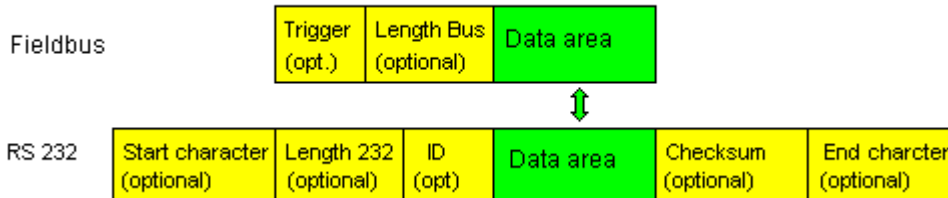
**Attention:**

*The rotary switches are only read in once when switching on the UNIGATE. Changes during the operation will be ignored!*

## 9.2 Universal 232 protocol

### 9.2.1 Data structure

Data structure, Universal 232 protocol



### 9.2.2 Fieldbus parameters

Trigger byte: See "The trigger byte -only at UNIGATE RS", chapter 6.5, on page 19

Length byte: See "The length byte - only at UNIGATE RS", chapter 6.6, on page 19

### 9.2.3 RS232 parameter table

#### 9.2.3.1 Start character

If this character is defined, the gateway evaluates only the data at the RS232 interface following this start character. Each transmission from the gateway via the RS232 interface is initiated with the start character in this case.

#### 9.2.3.2 Length 232

If this byte is activated, the gateway, at the receive end, awaits as many bytes of useful data (without checksum) as specified in this byte by the RS232 transmitter. At the transmission end, the gateway then sets this byte to the number of useful data items transmitted by it (without checksum). If byte "Length232" is not defined, the gateway, on reception at the RS232 interface, waits for the end criterion if this is defined. If no end criterion is defined either, as many characters as can be transferred in the fieldbus transmit buffer are read in via the RS232 interface.

As a special case for this parameter also a length byte with additional Timeout monitoring can be set in WINGATE. In that case the received characters will be discarded at a Timeout.



**Attention:**

*If "Timeout" is selected as end character, then this byte has no significance.*

### 9.2.3.3 ID

This byte is present only for reasons of compatibility. It is never activated.

### 9.2.3.4 Data area

The useful data is transferred in this field.

### 9.2.3.5 Checksum

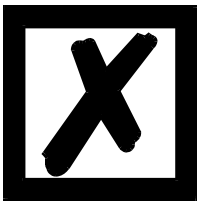
At the universal 232 protocol the following checksums can be selected:

XOR, bitwise sum, XOR with negated result and bitwise sum with negated result.

The checksum is always generated on the basis of bytes "Length232", "ID" and "Data area" if present. The checksum is generated by the gateway at the transmit end independently. On reception from the RS232 interface, the gateway checks the checksum and then transfers the useful data (without checksum) to the fieldbus buffer if no checksum errors have been detected. Otherwise, a local error message issued.

### 9.2.3.6 End character

If this character is defined, the gateway receives data from the RS232 interface up to this character. The "Timeout" criterion can be defined as a special case. In this case, the gateway continues to receive characters until a defined pause occurs. In the special case "Timeout" the "Length 232-byte" has no significance. At the transmit end, the gateway inserts the end character, if defined, as the last character of a transmission.



**Attention:**

***The configuration of an end character together with a check sum should be avoided, as a check sum that exactly corresponds with the end character might be the result. In such a case a misinterpretation and following a checksum-error is the result.***

## 9.2.4 Communication sequence

The useful data (data area) arriving via the fieldbus is copied in accordance with chapter 9.2.1 transparently into the RS232 data field and transferred via the RS interface, whereby the protocol is supplemented in accordance with the configuration (start character, end character...). NO acknowledgement is issued!

If the "Trigger byte" (see chapter 6.5) is active, data is sent only on a change of this byte. If the "Length byte" (see chapter 6.6) is active, only as many of the following bytes as specified there are transferred.

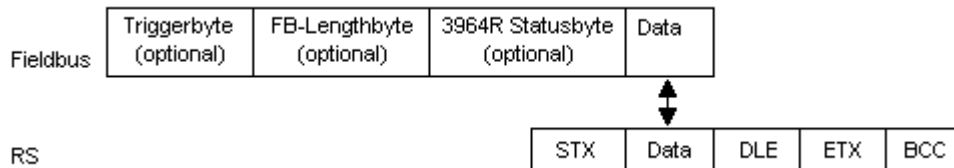
Receive data at the RS interface is evaluated in accordance with the configured protocol, and the data field (data area (see chapter 9.2.1)) is sent to the fieldbus-master. If more characters have been received than the fieldbus block length, the trailing bytes are truncated and an Rx Overrun is indicated. If less have been received, padding with 0 occurs. If the "Length byte" is active, the number of received useful data items is entered there. If the, "Trigger byte" is active, this is incremented by one after each complete reception operation at the RS interface.

## 9.3 The 3964 R protocol

The 3964 protocol is used to transfer data between two serial devices. One partner must be a high-priority partner and the other must be a low-priority partner in order to resolve initialisation conflicts.



### 9.3.1 Data structure 3964R



### 9.3.2 Protocol definitions

The telegram format is as follows:



- The received net data is forwarded (transparently) in both directions unchanged.
- **Attention:** The DLE-doubling is excluded from it; that means one DLE (10H) on the bus-side is sent on the RS-side twice. A double DLE on the RS-side is only sent once to the bus-master.
- Data blocking is not scheduled.
- The net data length is restricted to 236 bytes per telegram.
- Communication always runs between high-priority and low-priority communication partners.



**Attention:**

*The gateway is always using "Even Parity", as stipulated in the specification.*

### 9.3.3 Data communication

#### 9.3.3.1 Initiation of data communication by the low-priority user

If the low-priority user also receives an STX in response to a transmitted STX, it interrupts its transmit request, reverts to Receive mode and acknowledges the received STX with DLE.

A DLE in the data string is duplicated and included in the checksum. The BCC is computed from XORing all characters.

#### 9.3.3.2 Conflicts

#### 9.3.3.3 Timeout times

The timeout times are preset by the definition of the 3964R protocol and cannot be overwritten !!!  
tq = acknowledgement timeout time (2 s).

The acknowledgement timeout time is started after transmission of control character STX. If no positive acknowledgement arrives within the acknowledgement timeout time, the job is repeated (max. 2 x). If it has not been possible to complete the job positively after two repetitions, the high-priority device nevertheless attempts to establish contact with the low-priority partner by transmitting STX (cycle corresponds to tq).

tz = character timeout time ( 200 ms)

If the 3964 R driver receives data, it monitors arrival of the individual characters within period tz. If no character is received within the timeout time, the protocol terminates transfer. No acknowledgement is sent to the coupling partner.

### 9.3.3.4 Retries

In the event of negative acknowledgement or timeout, a telegram transmitted by the high-priority user is repeated twice. After this, the gateway signals communication as disturbed but still attempts to re-establish the connection.

### 9.3.3.5 Initiation of data communication by the high-priority user

In the case of a negative acknowledgement or timeout, a telegram transmitted by the external device is repeated twice before a fault is signalled.

### 9.3.4 State of the 3964R communication

The current state of the 3964R communication will be mapped in the fieldbus, if the parameter 3964R is activated in WINGATE. The states can be taken from the below table. This parameter takes effect from software revision 3.41 on and can be adjusted by means of the WINGATE software.

This information is only presented in the bus data from the gateway to the Master, however, it applies to the communication via 3964R in both directions.

Name	Value	Description
3964R_NO_ACTION	0	If this value is presented, no 3964R communication is active.
3964R_WAIT_AFTER_STX	1	After the transmission of the STX character the sender waits until the receiver has sent its confirmation.
3964R_WAIT_QUITTUNG	2	The receipt character is waited for.
3964R_WAIT_DATA	5	The gateway waits as receiver for the useful data.
3964R_WAIT_ZVZ	9	An execution time is waited for.

## 9.4 RK512

The gateway is not able to support the entire RK512-protocol (any data length through sequence telegram), as the handshake to the fieldbus-master is getting more complex, as though the gateway only uses 3964R, and the higher protocol is directly handled by the fieldbus-master (4 byte receipt and response data if necessary).

However, UNIGATE supports a simple kind of the RK512-protocol, whereas the data exchange is restricted to 1 word (fixed). In return the end user does not have to care about the problems of the quasi-full-duplex-operation, since he receives a fixed response to any command.

And apart from that the complete data exchange exclusively takes place through PDO's, which can be handled by any PLC very easily and quickly.

### 9.4.1 Data exchange

In detail the data exchange takes place as follows:

Commands from the RS1-side: PDO1 (Tx) [6 byte]

PDO1 (Tx) byte	1	2	3	4	5	6	(At FETCH Byte 5 + 6 equal 0)
	Cmd	Typ	DB	DW	DatH	DatL	
RK512-byte	3	4	5	6	11	12	Dat only at Cmd=SEND

Response to the above mentioned command from fieldbus-master in PDO1 (Rx) [2 byte]

PDO1 (Rx) byte	1	2	(At FETCH)
	Dat	Dat	
	Receipt	Receipt	(At SEND) = last two byte of the RK512-receipt

Commands from fieldbus-master = PDO2 (Rx) [6 byte]

PDO2 (Rx) byte	1	2	3	4	5	6	(Any byte 5 + 6 at FETCH)
	Cmd	Typ	DB	DW	DatH	DatL	
RK512-byte	3	4	5	6	11	12	(Dat only at Cmd=SEND)

Response to the above mentioned command from UNIGATE in PDO2 (Tx) [2 byte]

PDO2 (Tx) byte	1	2	
	Dat	Dat	(At FETCH)
	Receipt	Receipt	(At SEND) = last two byte of the RK512-receipt

Examples:

PLC will describe DB3, DW7 with 1234H:

PDO2 (Rx) = 41H 44H 03H 07H 12H 34H=> Receipt from UNIGATE: PDO2 (Tx) = 00H 00H

RS-device gets the data word from DB2 DW9 from PLC (there it says 4711H)

PDO1 (Tx) = 45H 44H 02H 09H 00H 00H=> Response from PLC: 47H 11H

The 4 PDO's are depicted on the objects as follows:

Object 2000H = PDO2 (Rx) (6 byte)

Object 2001H = PDO1 (Tx) (6 byte)

Object 2002H = PDO2 (Tx) (2 byte)

Object 2003H = PDO1 (Rx) (2 byte)



**Attention:**

*The gateway is always using "Even Parity", as stipulated in the specification.*

## 9.5 The fieldbus monitor protocol

This protocol has the function that all process data of the fieldbus are made available via the RS232 interface.

### 9.5.1 Function

For other CANopen® participants the gateway seems to be a normal CANopen-slave with 1 Rx and 1 Tx PDO each 1 byte as well as the objects 2000H to 2002H. The PDO's and objects are not used though.

Instead the gateway filters all PDO's out of the CAN data transfer whose COB-ID is in sector 181H..57FH.

The data of the PDO's is transferred via the RS232 interface in the following form:

COBID(low), COBID(high), PDO-Data(1)...PDO-Data(n) ; n=PDO data length

The transfer format on the RS interface is configured via WINGATE and corresponds to the UNIVERSAL-232 protocol (see also chapter 9.2). Thus it is possible to configure for example start and end signs, check sums a.s.o.

All the other CAN data packages - COB-ID out of 181H..57FH - are processed according to the CANopen® specification DS301.

Data that are received by the gateway via the RS interface are ignored. Thus there is only one data stream from the CANopen® to the RS interface.

## 9.6 Modbus-RTU

### 9.6.1 Notes

→ For reasons of simplicity, "Modbus-RTU" is referred to as "Modbus" in the text below.

- "Modbus-ASCII" is currently not supported.
- The terms "input" and "output" are always viewed from the gateway's point of view, i. e. fieldbus input data is the data sent by the fieldbus-master to the gateway.

## 9.6.2 UNIGATE as Modbus-master

### 9.6.2.1 Preparation

Before data exchange is commenced, the "Baud rate" and "Parity" parameters and, if applicable, the "Trigger byte" and the "Length byte" must be set. The number of "Start bits", "Stop bits" and "Data bits" are permanently preset.

In addition, a "Response time" which corresponds to the maximum time up to which the Modbus slave responds after a request must be set.

Since the Modbus operates with a variable data format - dependent on the required function and data length - but since the fieldbus requires a fixed data length, this must be preset by means of the configuration with WINGATE®. This length should be selected by the user such that the longest Modbus request resp. response can be processed. If a Modbus response is longer than the preset fieldbus length, the gateway signals an "Rx buffer overflow".

The user can choose whether the fieldbus requests are forwarded to the Modbus cyclically, in the event of a change, or on request. If he chooses cyclic mode, the next request follows directly after a response of the Modbus slave.

In "Change" mode, detection of a change is based on the fact that the fieldbus data are transferred only in case of an SDO/PDO-request via the Modbus.

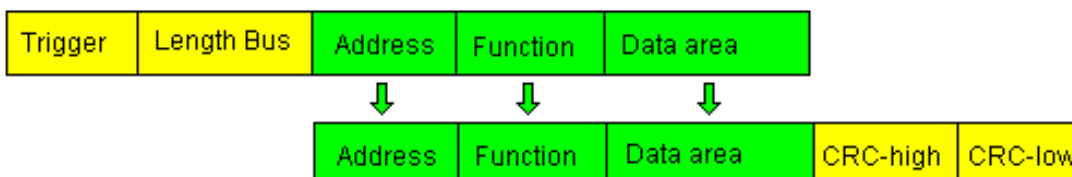
The third mode (Modbus request on request) necessitates the first byte in the fieldbus containing a trigger byte (see chapter 6.5). This byte is not transferred to the Modbus and serves only to start a Modbus transmission. For this purpose, the gateway constantly monitors this trigger byte and sends data to the Modbus only when this byte has changed. In the reverse direction (to the fieldbus), the gateway transfers the number of received Modbus data records in this byte, i.e. this byte is incremented by the gateway after each data record.

If the "Length byte" is activated (see chapter 6.6), the gateway transfers only the number of bytes specified there. The number of received Modbus data items is saved in the direction of the fieldbus-master. The length always refers to bytes "Address" to "Data n" (inclusive in each case), always without CRC checksum.

As the gateway determines the data length independently by means of the Modbus function, the length byte is not required.

### 9.6.2.2 Data structure

(optional) (optional)



### 9.6.2.3 Communication sequence

The gateway always acts as the slave with respect to the fieldbus and always acts as the master at the Modbus end. Thus, data exchange must always be started by the fieldbus-master. The gateway fetches this data which must be structured in accordance with Chapter "Data structure", from the fieldbus-master, determines the valid length of the Modbus data if the length byte is not activated, adds the CRC checksum and sends this data record as a request on the Modbus.

The response of the selected slave is then sent to the fieldbus-master by the gateway - without CRC checksum. If no response occurs within the stipulated "Response time", the gateway signals a "TIMEOUT ERROR".

## 9.6.3 UNIGATE as Modbus-slave

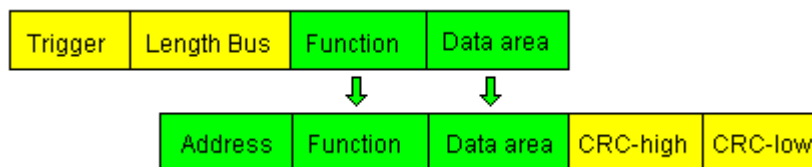
### 9.6.3.1 Preparation

Before data exchange is commenced, the parameters "Trigger byte" and "Length byte", "Baud rate", "Parity", "Start bits", "Stop bits" and "Data bits" must be set.

In addition, a "Response time" which corresponds to the maximum time up to which the fieldbus-master responds after a request must be set and at the rotary switch on the RS-side the Modbus-ID has to be set, under which the gateway is addressed in the Modbus.

Since the Modbus operates with a variable data format - dependent on the required function and data length - but since the fieldbus requires a fixed data length, this must be preset by UNIGATE. This length should be selected by the user such that the longest Modbus request resp. response can be processed. If a fieldbus-telegram is longer than the preset fieldbus length, the gateway signals an "Rx buffer overflow".

### 9.6.3.2 Data structure



### 9.6.3.3 Communication sequence

The gateway always acts as the slave with respect to the fieldbus and also acts as slave at the Modbus end. A data exchange is always initiated by the Modbus-master via the RS-interface. If the Modbus-address (1st Byte) which is sent out by the Modbus-master is identical with the address set on the gateway, the gateway sends the received data (without Modbus-address and CRC-check sum) to the fieldbus-master (look picture above). With it the gateway optionally completes as an introduction a Trigger byte and a Length byte.

The fieldbus-master detects when it has to analyze a record via the Trigger byte which is incremented by the gateway at every inquiry. The number of the following Modbus-data is to be found in the length byte.

Now the fieldbus-master has to analyze the Modbus-inquiry and it has to send back the answer in the same format (optionally with the leading Trigger byte and Length byte) via the fieldbus to the gateway.

The gateway then takes this answer and completes the Modbus-address and the CRC and sends the data to the Modbus-master via the RS-interface. With it the data exchange is completed and the gateway waits for a new inquiry from the Modbus-master.

#### **9.6.3.4 Status report**

If no response occurs within the stipulated "Response time" from the fieldbus-master, the gateway a "Timeout-Error".

The last Modbus-ID that was transferred to the RS-interface is indicated by the yellow LEDs.

## 10 Generating a script - only for UNIGATE SC

### 10.1 What is a script?

A script is a sequence of commands, that are executed in that exact order. Because of the fact that also mechanisms are given that control the program flow in the script it is also possible to assemble more complex processes from these simple commands.

The script is memory-oriented. It means that all variables always refer to one memory area. While developing a script you do not have to take care of the memory management though. The Protocol Developer takes on this responsibility for you.

### 10.2 Memory efficiency of the programs

A script command can carry out e. g. a complex checksum like a CRC-16 calculation via data. For the coding of this command only 9 byte are required as memory space (for the command itself). This is only possible when these complex commands are contained in a library.

A further advantage of this library is, that the underlying functions have been in practical use for a couple of years and therefore can be described as 'void of errors'. As these commands are also present in the native code for the controller, at this point also the runtime performance of the script is favorable.

### 10.3 What can you do with a script device?

Our script devices are in the position to process a lot of commands. In this case a command is always a small firmly outlined task. All commands can be put into classes or groups. A group of commands deals with the communication in general. This group's commands enable the gateway to send and receive data on the serial side as well as on the bus-side.

### 10.4 Independence of buses

Basically the scripts do not depend on the bus, they are supposed to operate on. It means that a script which was developed on a Profibus gateway can also be operated on an Interbus without changes, since the functioning of these buses is very similar. In order to also process this script on an Ethernet gateway, perhaps further adjustments have to be made in the script, so that the script can be executed reasonably.

There are no fixed rules how which scripts have to operate properly. When writing a script you should take into account on which target hardware the script is to be executed, so the necessary settings for the respective buses can be made.

### 10.5 Further settings at the gateway

Most devices require no further adjustments, except for those made in the script itself. However, there are also exceptions to it. These settings are made by means of the software WINGATE. If you know our UNIGATE-series, you are already familiar with the proceeding with it. An example is the adjustment of the IP-address and the net-mask of an Ethernet-gateway. These values have to be known as fixed values and are not available for the runtime. Another reason for the configuration of the values in WINGATE is the following: After an update of the script these values remain untouched, i. e. the settings that were made once are still available after a change of the script.

Only this way it is also possible that the same script operates on different Ethernet-gateways, that feature different IP-addresses.

## 10.6 The use of the Protocol Developer

The Protocol Developer is a tool for an easy generation of a script for our script gateways. Its operation is exactly aimed at this use. After starting the program the script that was loaded the last time is loaded again, provided that it is not the first start.

Typical for Windows script commands can be added by means of the mouse or the keyboard. As far as defined and required for the corresponding command, the dialog to the corresponding command is displayed, and after entering the values the right text is automatically added to the script. The insertion of new commands by the Protocol Developer is carried out in a way that existing commands will not be overwritten. Generally a new command is inserted in front of the one where the cursor is positioned. Of course the commands can also be written by means of the keyboard or already written commands can also be modified.

## 10.7 Accuracies of the baud rates at UNIGATE SC

The baud rate of the serial interface is derived from the processor's crystal frequency. Meanwhile all Script-gateways, except for the MPI-Gateways (20 MHz), are working with a crystal frequency of 40 MHz.

You can enter any desired integer baud rate into the script. After that the firmware adjusts the baud rate, that can be derived the most precisely from the crystal frequency.

The baud rate the gateway is actually working with (BaudIst) can be determined as follows:

$$\begin{aligned} \text{BaudIst} &= (\text{F32} / \text{K}) \\ \text{F32} &= \text{Crystal frequency [Hz]} / 32 \\ \text{K} &= \text{Round}(\text{F32} / \text{BaudSoll}); \\ &\quad \text{Round}() \text{ is a commercial roundoff} \end{aligned}$$

Example:

The actual baud rate is to be calculated, when 9600 baud are pre-set, where the gateway is operated with 40 MHz:

$$\begin{aligned} \text{F32} &= 40000000 / 32 = 1250000 \\ \text{K} &= \text{Round}(1250000 / 9600) = \text{Round}(130.208) = 130 \\ \text{BaudIst} &= 1250000 / 130 = 9615.38 \end{aligned}$$

I. e.: The baud rate actually adjusted by the gateway is 9615.38 baud

The resulting error in per cent can be calculated as follows:

$$\text{Error}[\%] = (\text{abs}(\text{BaudIst} - \text{BaudSoll}) / \text{BaudSoll}) * 100$$

In our example the following error results:

$$\text{Error} = (\text{abs}(9615.38 - 9600) / 9600) * 100 = 0.16\%$$

In practise errors below 2% can be tolerated!

In the following please find a listing of baud rates at a 40 MHz-crystal frequency with the corresponding errors:



4800 baud: 0.16%  
9600 baud: 0.16%  
19200 baud: 0.16%  
38400 baud: 1.35%  
57600 baud: 1.35%  
62500 baud: 0%  
115200 baud: 1.35%  
312500 baud: 0%  
625000 baud: 0%

## 10.8 Script processing times

The Script is translated by the Protocol Developer and the consequently generated code is loaded into the Gateway. Now the processor in the Gateway interprets this code. In this case, there are commands that can be processed very fast (e. g. "Set Parameter"). There are also commands, however, that take longer (e. g. copying 1000 bytes). Consequently, for one thing the processing time differs due to the kind of Script command. But the processing time of the Script commands is considerably more determined by the processor time that is available for this process. Since the processor has to carry out several tasks simultaneously (multitasking system) only a part of the processor's capacity is available for the Script processing. The following tasks - in the order of priority - are executed on the processor:

- Sending and receiving data at the Debug-interface (provided that the Protocol Developer has been started on the PC)
- Sending and receiving data at the RS-interface
- Sending and receiving data at the Fieldbus-interface
- Tasks controlled via internal clock (1 ms) (e. g. flashing of an LED)
- Processing of the Script

From experience approximately 0.5 ms can be calculated for each Script line. This value confirmed itself again and again in many projects as a standard value. He is always quite right if the processor has enough time available for the Script processing.

By means of the tasks mentioned above, the following recommendation can be formulated in order to receive a rather fast Script processing:

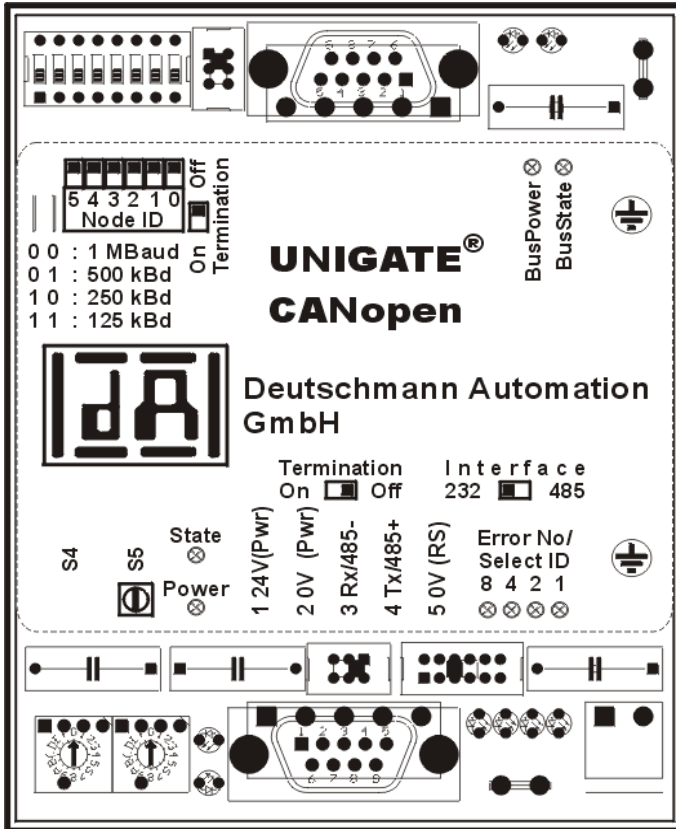
- Deactivate the Debug-interface (it is the normal case in the serial use)
- Keep the data length at the RS-interface as small as possible. The baud rate is not the problem here, but the amount of characters which are transferred per second.
- Do not unnecessarily extend the data length at the Fieldbus side. Especially at acyclical bus data, if possible do only send them when changes were made. The data length at buses that are configured to a fixed length (e. g. Profibus) should not be longer than absolutely necessary.

If the processing time should be too large in spite of these measures, there is the possibility to generate a customized Script command, that executes several tasks in one Script command. Please contact our support department for this purpose.

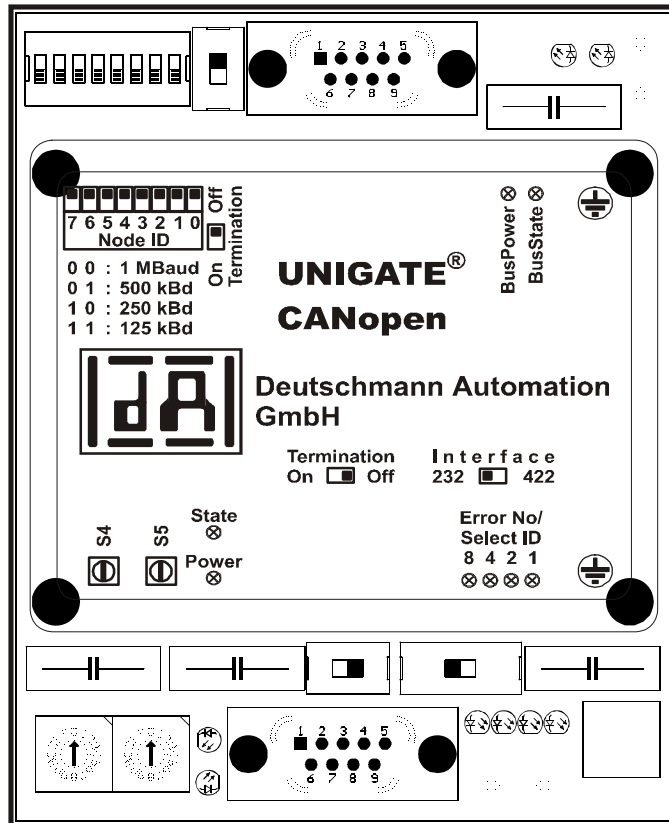
# 11 Hardware ports, switches and LEDs

## 11.1 Drawing of the unit

### 11.1.1 Model UNIGATE RS / SC 232/485-CANopen



### 11.1.2 Model UNIGATE RS / SC 232/422-CANopen



## 11.2 Configuration of the UNIGATE RS

The gateway is delivered by the manufacturer with the following configuration

- Transparent data transmission (i.e. Layer 7 is transferred unchanged)

The configuration of the gateway can be changed by the customer. That is what the software WINGATE<sup>®</sup> that comes along with the gateway is meant for.

For the configuration the gateway has to be brought to the configuration mode. For that reason the switches S4 and S5 have to be set to the position "F" each and the interface-switch has to be set to "232". Consequently a connection to the PC has to be established and the gateway has to be started once more. The program WINGATE<sup>®</sup> automatically selects the interface parameters correctly. For the operation of WINGATE<sup>®</sup> please take a look at the WINGATE<sup>®</sup> Online help.

### 11.2.1 CANopen

- Baud rate: According to the setting DIP-switch
- Sync: Supported
- Node guarding: Supported (**Default value: Guardtime=500 ms Lifetime=3**)

### 11.2.2 RS232/RS485/RS422

- Start bit: 1
- Data bits: 8
- Stop bit: 1
- Parity: None
- Baud rate: 9600 Baud

The supplied WINGATE® software can be used to change this configuration. Likewise, it can be used to select customized protocols. A description is given below.

## 11.3 Connectors

### 11.3.1 Connector to the external device (RS-interface)

The connection cable to the external device must be plugged in at the connector accessible on the underside of the device.

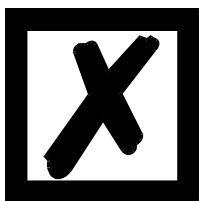
Pin assignment (5-pin screw-type plug connector; not available for RS422)

Pin No.	Name	Function
1	10.8..30 V/DC power supply	Power supply
2	0 V power supply	Power supply
3	RX / RS485- (RS485 B)	Receive signal
4	TX / RS485+ (RS485 A)	Transmit signal
5	GND	Reference for PIN 3 + 4

Pin assignment (9-pin Sub-D, plug)

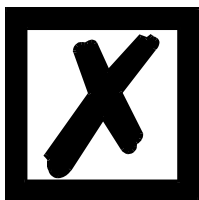
(Standard for the RS232/422-model, optional for the RS232/485-model)

Pin No.	Name	Function
1	Shield	
2	Rx	Receive signal
	RS485- RS422- (Tx)	Transmit-receive signal Transmit signal
3	Tx	Transmit signal
	RS485+ RS422+ (Tx)	Transmit-receive signal Receive signal
4		
5	GND	Ground connection, reference for PIN 2+3+6+7
6	RS422- (Rx)	Receive signal
7	RS422+ (Rx)	Receive signal
8		
9		



**Attention:**

*The assignment of the 9-pin Sub-D plug may differ from the above pin assignment and also a customized configuration is possible.*



**Attention:**

*In case the RS-interface is NOT potentially divided, "GND" and "supply 0V" are connected internally!*

**Attention:**

***In case RS422 is selected, then the termination switch has to be set to ON, in order to avoid communication errors!***

Pin assignment (2-pin screw-type/plug connector; only in conjunction with 9-pin D-Sub.)

Pin No.	Name	Function
1	10.8..30 V / DC	10.8..30 V supply voltage
2	0 V / DC	0 V supply voltage

### 11.3.2 CANopen® connector

The connector (labelled: CANopen) for connection to CANopen® is located on the upper side of the device.

Pin No.	Name	Function
1		
2	CAN-L	Dominant low
3	CAN-GND	CAN Ground
4		
5		
6		
7	CAN-H	Dominant high
8		
9		

### 11.3.3 Debug plug

Please compare to chapter 4.2.

### 11.3.4 Power supply

The device must be powered with 10.8...30 VDC.

Please note that the devices of the series UNIGATE can not be operated with AC voltage.

On the version with 5-pin screw-type/plug connector, the power supply is routed via this connector; on the version with 9-pin D-Sub connector, the power supply is routed via the separate 2-pin screw-type/plug connector.

### 11.3.5 Shield terminal lead

The shield signal for the electronic circuitry is connected to the top-hat rail via the connector provided. The shield signal for the CANopen® cable shield is not electrically connected to the shield signal of the electronic circuitry for reasons relating to interference immunity.

## 11.4 LEDs

The gateway features 8 LEDs with the following significance:

LED Bus Power	green	CANopen® supply voltage
LED Bus State	red/green	Interface status, CANopen
LED Power	green	RS485/RS232 supply voltage
LED State	red/green	Interface status RS485/RS232
LED Error No / Select ID	yellow	Binary display of the connection/error number

In the configuration mode these displays are not valid and only meant for internal use.

The gateway UNIGATE SC has also 8 LEDs available with the following significance:

LED Bus Power	green	CANopen® supply voltage
LED Bus State	red/green	Interface state, CANopen
LED Power	green	Supply voltage RS485/RS232
LED State	red/green	User-defined
LED Error No. / Select ID	yellow	User-defined

#### 11.4.1 LED “Bus Power“

This LED is connected directly to the electrically isolated supply voltage of the CANopen® end.

#### 11.4.2 LED “Bus State“

Lights green	CAN-Status = OPERATIONAL
Blinks green	CAN-Status = PREOPERATIONAL or PREPARED
Blinks red	Guarding error
Lights red	General CANopen® error

#### 11.4.3 LED “Power“

This LED is connected directly to the (optionally also electrically isolated) supply voltage of the RS485/RS232 end.

#### 11.4.4 LED “State“

Lights green (only RS)	Data exchange active via RS485/RS232
Flashes green (only RS)	RS485/RS232 ok but no permanent data exchange
Flashes green/red (only RS)	No data exchange since switching on
Lights red	General gateway error (see LEDs Error No.)
Flashes red (only RS)	UNIGATE is in the configuration mode

#### 11.4.5 LEDs “Error No. / Select ID“ at UNIGATE RS

If these 4 LEDs flash and LED “State“ simultaneously lights red, the error number is displayed in binary notation (conversion table, see Annex) in accordance with the table in Chapter "Error handling". Otherwise, the address with which communication is currently running via the RS485 interface is displayed, also in binary notation.

### 11.5 Switches

The gateway features 7 switches with the following functions:

DIP-switch	Node-ID and Baud rate
Slide switch “Termination“	Switchable CANopen® terminating resistor
Rotary coding switch S4	RS485 ID (High Byte)
Rotary coding switch S5	RS485 ID (Low Byte)
Slide switch “Interface“	Selector switch for RS485 or RS232 interface
Slide switch “Termination“	Switchable RS485 terminating resistor

#### 11.5.1 DIP-switch

This DIP-switch is used to set the Node-ID and Baud rate.

Only the IDs 0-63 can be adjusted with the hardware DIP-switch. In order to set an address in the area 64-127, an offset of 64 to the hardware setting is required. This setting can be carried out with WINGATE® at CANopen® gateways from software revision 3.1 on. The setting "Node ID Offset 64" can have the values "enabled" and "disabled". In case the value is "enabled", then the value 64 is added to the hardware setting.

### **11.5.2 Slide switch Termination CANopen**

If the gateway is operated as the first or last physical device in the CANopen, there must be a bus termination at this gateway. In order to do this, either a bus terminating resistor must be activated in the connector or the resistor (220  $\Omega$ ) integrated in the gateway must be activated. In order to do this, slide the slide switch to position ON. In all other cases, the slide switch must remain in position OFF. Please refer to the general fieldbus literature for further information on the subject of bus termination.

### **11.5.3 Rotary coding switches S4 + S5 (RS485 ID)**

These two switches are used to set the RS485 ID of the gateway in hexadecimal notation provided an ID is required for the bus. Please refer to the Annex for a conversion table from decimal to hexadecimal. This value is read in only once when the gateway is activated.

### **11.5.4 Slide switch (RS485/RS232 interface)**

This slide switch is used to select whether an RS485 interface or an RS232 interface is connected at the connector to the external device.

### **11.5.5 Slide switch (RS485/RS422 termination)**

If the gateway is operated as the first or last physical device in the RS485 bus, there must be a bus termination at this gateway. In order to do this, either a bus terminating resistor in the connector or the resistor (150  $\Omega$ ) integrated in the gateway must be activated. In order to do this, slide the slide switch to position ON. In all other cases, the slide switch must remain in position OFF. Please refer to the general RS485 literature for further information on the subject of bus terminations.

If the integrated resistor is used, please allow for the fact that this also activates a pull-down resistor (390  $\Omega$ ) to ground and a pull-up resistor (390  $\Omega$ ) to VCC.

At the RS422-interface the transmission line is terminated. The receive line is always firmly terminated.

## 12 Error handling

### 12.1 Error handling at UNIGATE RS

If the gateway detects an error, the error is signalled by the "State" LED lighting red and, simultaneously, the error number being indicated by means of LEDs "Error No." as shown in the table below. In addition, this error number is transmitted as an Emergency-telegram via CANopen. The code 61xx (hex), that indicates an internal Firmware error according to DS301 is used for it. The current error code is entered for "xx" in accordance with the below table. A detail error, that is used for internal purposes may still be included in byte 3 and 4 of the emergency message.

A distinction can be made between two error categories:

Serious errors (1-5): In this case, the gateway must be switched off and switched back on again. If the error occurs again, the gateway must be exchanged and returned for repair.

Warnings (6-15): These warnings are displayed for one minute simply for information purposes and are then automatically reset. If such warnings occur frequently, please inform After-Sales Service.

Error 10 is additionally activated in case of the heartbeat-error.

In the configuration mode these displays are not valid and only meant for internal use.

LED8	LED4	LED2	LED1	Error no. resp. ID	Error description
0	0	0	0	0	Reserved
0	0	0	1	1	Initialisation error of the RS485/RS232 interface
0	0	1	0	2	EEROM error
0	0	1	1	3	Stack error
0	1	0	0	4	Hardware fault of the fieldbus ASIC
0	1	0	1	5	Configuration error of the gateways (unknown protocol)
0	1	1	0	6	Reserved
0	1	1	1	7	RS485/RS232 transmit buffer overflow
1	0	0	0	8	RS485/RS232 receive buffer overflow
1	0	0	1	9	Timeout on reception RS485/RS232 interface
1	0	1	0	10	Transmit error of the RS485/RS232 interface
1	0	1	1	11	Parity- or frame-check-error
1	1	0	0	12	Addressing error of the RS485/RS232 interface
1	1	0	1	13	Internal error
1	1	1	0	14	General error of the RS485/RS232 interface
1	1	1	1	15	Internal error

Table 1: Error handling at UNIGATE RS



## 12.2 Error handling at UNIGATE SC

LED8	LED4	LED2	LED1	Error no. resp. ID	Error description
0	0	0	0	0	Reserved
0	0	0	1	1	Hardware fault
0	0	1	0	2	EEROM error
0	0	1	1	3	Internal memory error
0	1	0	0	4	Fieldbus hardware fault or wrong fieldbus ID
0	1	0	1	5	Script error
0	1	1	0	6	Reserved
0	1	1	1	7	RS-transmit buffer overflow
1	0	0	0	8	RS-receive buffer overflow
1	0	0	1	9	RS-time out
1	0	1	0	10	General fieldbus error
1	0	1	1	11	Parity- or frame-check-error
1	1	0	0	12	Reserved
1	1	0	1	13	Fieldbus configuration error
1	1	1	0	14	Fieldbus buffer overflow
1	1	1	1	15	Reserved

Table 2: Error handling at UNIGATE SC

## 13 Housing variations

Optional Deutschmann Automation GmbH & Co. KG supplies gateways with housing-variations that fulfill the protection type IP 65/66.

Housings are designed to protect installations against dangers from the outside. These dangers can be of diverse nature and they are very important for the selection criteria of a specific housing variation. It should be decisive for the decision which dangers are to be considered at the place of action and which conditions the environment stipulate.

Now Deutschmann Automation offers two different housing-variations:

- Aluminium housings
- High-grade steel housings
- (Polycarbonate housings, only supplied until January 2006)

Aluminium housings are robust terminal housings for distributions and connections of machines and plants, high-grade steel housings are particularly used in aggressive environments and in the hygiene field (Housing data, see chapter 15.2).

### 13.1 Main operational fields for housings

#### Main operational fields of aluminium housings with protection type IP 66

- Machines, control- and plant manufacturing
- In aggressive and rough environmental conditions e. g. sea water field / on ships
- Electrical- and building installation engineering
- Measuring-, switching- and controlling technique
- EMC Directives - relevant fields
- Heating-, air-conditioning- and ventilation technique and
- Medical field

#### Main operational fields of high-grade steel housings with protection type IP 66

- In aggressive and rough environmental conditions e.g. sea water field
- In fields with highest purity and cleanness (e.g. food field)
- EMC Directives applications

### 13.2 Mounting instruction

#### Mounting instruction of the aluminium housing

The assembly with the dimensions (160x160x90 BxHxT) is developed for wall mounting.

#### Mounting

1. Using the drill template (see chapter 19, Drilling templates) and mark the four holes (the measures can also be taken from the reverse side of the housing).
2. Unscrew the cover and fix the housing with four screws (max M6) in the marked holes.

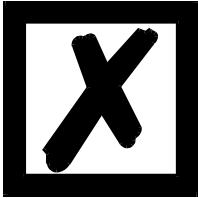


3. *The housing have to be fixed on a grounded mounting place.*
4. *Under the housing must be 5 cm place for the insertion of the cable.*

**Attention: All leads have to be shielded.**

The fieldbus leadings have to go through the PG - screwing - without connecting the shield with the housing.

The shield of the fieldbus cable have to be put exclusive on the D-sub-plug



***Except for the DeviceNet gateway:  
Here the shield is put on the PG-screwing.***

## 14 Installation guidelines

### 14.1 Installation of the module

The module of size (90 x 127 x 55 mm W x H x D) has been developed for switch cabinet use (IP 20) and can thus be mounted only on a standard mounting channel (deep top-hat rail to EN 50022).

#### 14.1.1 Mounting

- Engage the module from the top in the top-hat rail and swivel it down so that the module engages in position.
- Other modules may be rowed up to the left and right of the module.
- There must be at least 5 cm clearance for heat dissipation above and below the module.
- The standard mounting channel must be connected to the equipotential bonding strip of the switch cabinet. The connection wire must feature a cross-section of at least 10 mm<sup>2</sup>.
- An earthing terminal must be positioned next to the module so as to allow the shield connection on the device to be implemented as short as possible with a flexible wire (1.5 mm<sup>2</sup>).

#### 14.1.2 Removal

- First disconnect the power supply and signal lines.
- Then push the module up and swivel it out of the top-hat rail.

Vertical installation

The standard mounting channel may also be mounted vertically so that the module is mounted turned through 90°.

### 14.2 Wiring

#### 14.2.1 Connection systems

The following connection systems must resp. may be used when wiring the module:

- Standard screw-type/plug connection (power supply + RS)
  - Push-lock terminals (connection terminals for earthing)
  - 9-pin SUB-D plug connectors (CANopen® and RS232 connection)
- a) In the case of standard screw-type terminals, one lead can be clamped per connection point. It is best to then use a screwdriver with a blade width of 3.5 mm to firmly tighten the screw.

Permitted cross-sections of the line:

- Flexible line with wire-end ferrule: 1 x 0.25 ... 1.5 mm<sup>2</sup>
- Solid conductor: 1 x 0.25 ... 1.5 mm<sup>2</sup>
- Tightening torque: 0.5 ... 0.8 Nm

b) The plug-in connection terminal strip is a combination of standard screw-type terminal and plug connector. The plug connection section is coded and can thus not be plugged on the wrong way round.

c) The 9-pin SUB-D plug connectors are secured with two screws with "4-40-UNC" thread. It is best to use a screwdriver with a blade width of 3.5 mm to screw the screw tight.  
Tightening torque: 0.2... 0.4 Nm

## 14.2.2 CANopen® communication interface

### 14.2.2.1 Bus line with copper cable

This interface is located on the module in the form of a 9-pin SUB-D plug (pin) on the upper surface of the housing.

- Plug the CANopen® connector onto the SUB-D socket labelled "CANopen".
- Firmly screw the securing screws of the plug connector tight using a screwdriver.
- If the module is located at the start or end of the CANopen® line, you must connect the bus terminating resistor integrated in the gateway. In order to do this, slide the slide switch to the position labelled ...on...
- If the module is not located at the start or at the end, you must set the slide switch to position "off".

### 14.2.2.2 Power supply

The device must be powered with 10,8...30 V DC.

- Connect the supply voltage to the 5-pin or optional 2-pin plug-in screw terminal in accordance with the labelling on the front panel of the device.

### 14.2.2.3 Shield connection

The module features two contact points for equipotential bonding and the shield of the RS end. The shield of the CANopen® cable is connected to the equipotential bonding system via an RC snubber circuit. This means that there are two electrically isolated shields in the device. This guarantees higher interference immunity of the module since the "cable shield current" which may be up to a few Amperes owing to potential differences between two bus users is not discharged via the device.

If the device is subject to high mechanical or chemical stress, it is advisable to use a tin-plated top-hat rail in order to ensure greater contact stability of the shield connection!

### 14.2.2.4 Equipotential bonding connection

- Fit an earthing terminal to the top-hat rail directly next to the module. The earthing terminal automatically establishes an electrical connection to the top-hat rail.
- Connect the shield connection terminal to the earthing terminal using a flexible wire with a diameter of 1.5 mm<sup>2</sup> which should be as short as possible.
- Connect the top-hat rail to the equipotential bonding rail with as low an impedance as possible. Use a flexible earthing wire with a cross-section of at least 10 mm<sup>2</sup> for this.

## 14.2.3 Line routing, shield and measures to combat interference voltage

This Chapter deals with line routing in the case of bus, signal and power supply lines, with the aim of ensuring an EMC-compliant design of your system.

### 14.2.4 General information on line routing

- Inside and outside of cabinets

In order to achieve EMC-compliant routing of the lines, it is advisable to split the lines into the following line groups and to lay these groups separately.

- ⇒ Group A:
- shielded bus and data lines (e.g. for CANopen, RS232C and printers etc.)
  - shielded analogue lines
  - unshielded lines for DC voltages  $\geq 60$  V
  - unshielded lines for AC voltage  $\geq 25$  V
  - coaxial lines for monitors
- ⇒ Group B:
- unshielded lines for DC voltages  $\geq 60$  V and  $\geq 400$  V
  - unshielded lines for AC voltage  $\geq 24$  V and  $\geq 400$  V
- ⇒ Group C:
- unshielded lines for DC voltages  $> 400$  V

The table below allows you to read off the conditions for laying the line groups on the basis of the combination of the individual groups.

	Group A	Group B	Group C
Group A	1	2	3
Group B	2	1	3
Group C	3	3	1

Table: Line laying instructions as a function of the combination of line groups

- 1) Lines may be laid in common bunches or cable ducts.
- 2) Lines must be laid in separate bunches or cable ducts (without minimum clearance).
- 3) Lines must be laid in separate bunches or cable ducts inside cabinets but on separate cable racks with at least 10 cm clearance outside of cabinets but inside buildings .

#### 14.2.4.1 Shielding of lines

Shielding is intended to weaken (attenuate) magnetic, electrical or electromagnetic interference fields.

Interference currents on cable shields are discharged to earth via the shielding bus which is connected conductively to the chassis or housing. A low-impedance connection to the PE wire is particularly important in order to prevent these interference currents themselves becoming an interference source.

Wherever possible, use only lines with braided shield. The coverage density of the shield should exceed 80 %. Avoid lines with foil shield since the foil can be damaged very easily as the result of tensile and compressive stress on attachment. The consequence is a reduction in the shielding effect.

In general, you should always connect the shields of cables at both ends. The only way of achieving good interference suppression in the higher frequency band is by connecting the shields at both ends.

The shield may also be connected at one end only in exceptional cases. However, this then achieves only an attenuation of the lower frequencies. Connecting the shield at one end may be more favorable if

- it is not possible to lay an equipotential bonding line
- analogue signals (a few mV resp. mA) are to be transmitted
- foil shields (static shields) are used.

In the case of data lines for serial couplings, always use metallic or metallized plugs and connectors. Attach the shield of the data line to the plug or connector housing. Do not connect the shield to one Pin of the multipole connector!

If there are potential differences between the earthing points, a compensating current may flow via the shield connected at both ends. In this case, you should lay an additional equipotential bonding line.

Please note the following points when shielding:

- Use metal cable clips to secure the shield braiding. The clips must surround the shield over a large area and must have good contact.
- Downstream of the entry point of the line into the cabinet, connect the shield to a shielding bus. Continue the shield as far as the module, but do not connect it again at this point!

Shielded data lines and unshielded power supply lines (< 60 V DC) are routed to and connected to the module. All cable shields must be earthed at both ends in order for the module to comply with all required EMC limits.

- Downstream of the entry point into the switch cabinet, the CANopen® cable shield must be connected to the equipotential bonding strip.
- You must earth the shielded RS232C connection cable via the shield terminal on the module using a flexible wire with a cross-section of at least 1.5 mm<sup>2</sup> which is as short as possible.
- You must also earth the other end of the cable shield of the RS232C connection cable. (see also chapter 14.2.2.3)

## 15 Technical data

### 15.1 Device data

The technical data of the module is given in the table below.

No.	Parameter	Data	GL-certified design	Explanations
1	Location	Switch cabinet	On ships	Top-hat rail mounting
2	Enclosure	IP24 / IP66	IP 66	Protection against foreign bodies and water to IEC 529 (DIN 40050)
4	Service life	10 years	10 years	
5	Housing size	90 x 127 x 55 mm	120 x 122 x 80 mm	W x H x D
6	Installation position	Any	Any	
7	Weight	0.3 kg	1.15 kg	
8	Operating temperature	-20°C ..+55°C humidity (non-condensing)	0°C ..+55°C	
9	Storage/transport temperature	- 40 °C ... + 70 °C	-40°C..70°C	
10	Atmospheric pressure during operation during transport	795 hPa ... 1080 hPa 660 hPa ... 1080 hPa	-	
11	Installation altitude	2000 m 4000 m	Not applicable	Unrestricted Restricted - Ambient temperature ≤ 40 °C
12	Relative humidity	Max. 80 %	Max. 100 %	No condensation, no corrosive atmosphere
14	External power supply	10.8...30 V DC	24 V ± 20%	Standard power supply unit to DIN 19240
15	Current consumption at 24 V DC	Typ. 120 mA, max 150 mA	Max. 150 mA	At 10.8V: typ. 350 mA
16	Reverse voltage protection	Yes	Yes	But does not function!
17	Short-circuit protection	Yes	Yes	
18	Overload protection	Poly-switch	Poly-switch	Thermal fuse
19	Undervoltage detection (USP)	≤ 9 V DC	-	
20	Emergency power supply	≥ 5 ms	-	Device fully operable

Table: Technical data of the module



The table below lists all tests, standards and regulations on the basis of which the module has been tested.

No.	Parameter	Data	GL-certified design	Explanations
1	Vibration test	5 Hz ≤ f ≤ 26 Hz, amplitude = 0.75 mm 26 Hz ≤ f ≤ 500 Hz, acceleration = 20 m/s <sup>2</sup> → Frequency sweep : 1 octave/min. → 10 frequency sweeps each in x, y, z	2 Hz ≤ f ≤ 25 Hz : ±1.6 mm 25 Hz ≤ f ≤ 100 Hz: 4 g	IEC 60068-2-6 (Fe)
2	Shock test	Shock waveform = semi-sinusoidal Acceleration = 15g (150 m/s <sup>2</sup> ) Shock duration = 11 ms → 3 shocks in +/- direction in x, y, z	-	(IEC 68-2-27-Ea)
3	ESD	8 kV discharge in air 4 kV contact discharge	8 kV discharge in air 4 kV contact discharge	EN 50082-2
4	Electromagnetic fields	10 V/m	80 MHz..1GHz, 10V/m	EN 50082-2
5	BURST	2 kV / 5 kHz supply voltage 1 kV / 5 kHz data lines	2 kV / 5 kHz supply voltage 1 kV / 5 kHz data lines	EN 50082-2
6	Emitted interference	Limit value class A	Wire-bound: 10 KHz ..150 KHz: 120 dBμV .. 69 dBμV 150 KHz.. 1.5 MHz: 79 dBμV .. 63dBμV 1.5 MHz..30 MHz: 63 dBμV  Radiation: 150 KHz..30 MHz: 80 dBμV..50dBμV 30 MHz.. 100 MHz: 60 dBμV.. 54dBμV 100 MHz.. 1GHz: 54 dBμV 156 MHz.. 165 MHz: 24 dBμV	EN 55011
7	Approvals	CE mark	GL	→ Certificate of Conformity

Table: Tests, standards and regulations

### 15.1.1 Interface data

The table below lists the technical data of the interfaces and ports on the device. The data has been taken from the corresponding Standards.

No.	Interface designation Physical interface	CANopen RS485	RS232-C RS232-C	RS485/RS422 RS485/RS422
1	Standard	CiA® DS 102	DIN 66020	EIA Standard
2	Transmission mode	Symmetrical asynchronous serial half-duplex  → Difference signal	Asymmetrical asynchronous serial full duplex  → Level	Symmetrical asynchronous serial half-duplex full duplex at RS422  → Difference signal
3	Transmission method	Master / slave	Master / slave	Master / slave
4	Number of users : - Transmitters - Receivers	32 32	1 1	32 32
5	Cable length: - Maximum  - Baud rate-dependent	1300 m 50 kBd → 1300 m 100 kBd → 640 m 200 kBd → 310 m 500 kBd → 112 m 1 MBd → 40 m	15 m  no	1200 m  <93.75 kBd → 1200 m 312, kBd → 500 m 625 kBd → 250 m
6	Bus topology	Line	Point-to-point	Line
7	Data rate: - Maximum  - Standard values	1Mbit/s  125 kB 250 kB 500 kB 1 MB	120 kBit/s 2.4 k/B 4.8 k/B 9.6 kBit/s 19.2 kBit/s 38.4 kBit/s 57.6 kBit/s	625 kBaud 2.4 kBit/s 4.8 kBit/s 9.6 kBit/s 19.2 kBit/s 57.6 kB 312.5 kB 625 kB
8	Transmitter: - Load - Maximum voltage - Signal, unloaded - Signal, loaded	54 Ω - 7 V ... 12 V ± 5 V ± 1.5 V	3 ... 7 kΩ ± 25 V ± 15 V ± 5 V	54 Ω - 7 V ... 12 V ± 5 V ± 1.5 V
9	Receiver: - Input resistance - Max. input signal - Sensitivity	12 Ω - 7 V ... 12 V ± 0.2 V	3 ... 7 Ω ± 15 V ± 3 V	12 Ω - 7 V ... 12 V ± 0.2 V
10	Transmit range (SPACE): - Voltage level - Logic level	- 0.5 ... + 0.05 V 0	+ 3 ... + 15 V 0	- 0.2 ... + 0.2 V 0
11	Transmit pause (MARK): - Voltage level - Logic level	+ 1.5 ... +3 V 1	- 3 ... -15 V 1	+ 1.5 ... +5 V 1

Table: Technical data of the interfaces and ports on the module

## 15.2 Housing data

The table below lists the technical data of the different housing variations.

	Aluminium housing	High-grade steel housing	Polycarbonat housing (supplied until January 2006)
<b>Material</b>	AlSi12, DIN 1725	High-grade-steel 1.25 mm, 1.4404/316L	Polycarbonat, thermoplasti- cally splashed
<b>Protection type</b>	IP 66, EN 60529	IP 66, EN 60529	IP 66, EN 60529
<b>Impact strength</b>	> 7 Nm, EN 50014		> 30 kv/mm
<b>Temperature resistance</b>	- 40° to + 90°C	-60° C to + 130° C	-40° to + 100°
<b>Seals</b>	Polyurethan seal, compact porosity, foamed	Polyurethan seal, compact porosity, foamed	Spring groove system
<b>Surface coating</b>	Powder mold coating, color RAL 7001 silvery gray	Polished grain 240	Tinted RAL 7035
<b>Outside dimen- sion</b>	160 x 160 x 90 cm	150 x 150 x 78 cm	120 x 122 x 85 cm
<b>Weight (with gateway)</b>	2.05 kg	1.60 kg	0.45 kg

## 16 Commissioning guide

### 16.1 Note

Only trained personnel following the safety regulations may commission the UNIGATE.

### 16.2 Components

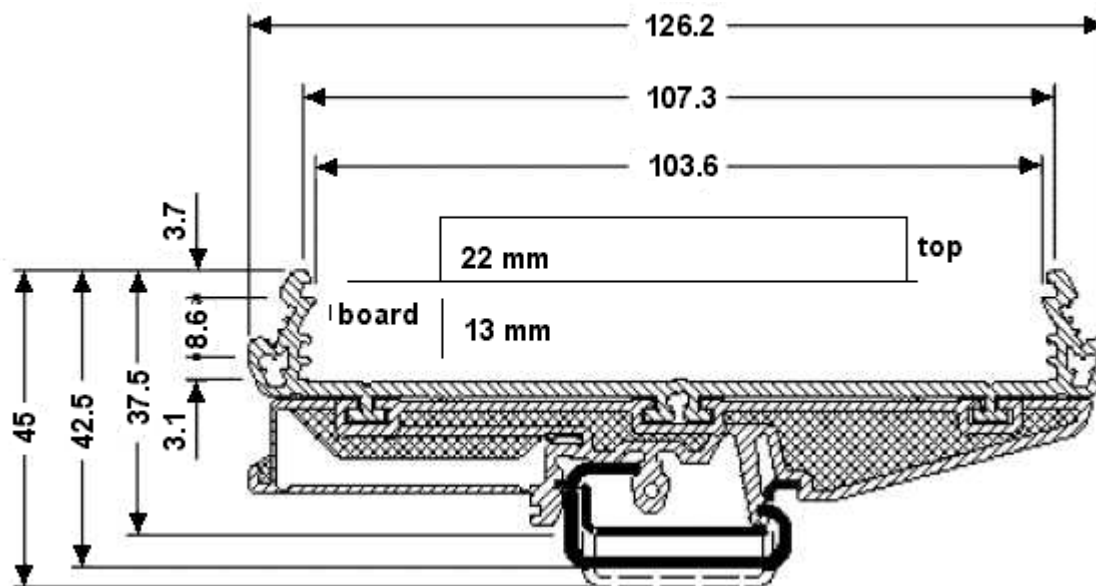
You will require the following components to commission the UNIGATE:

- UNIGATE
- Connection cable from gateway to the process
- Connector for CANopen® connection to the gateway
- CANopen® cable (this cable is generally already installed on site!)
- 10.8..30 V DC power supply (DIN 19240)

### 16.3 Installation

The UNIGATE module features enclosure IP 20 and is thus suitable for switch cabinet use. The device is designed for snapping onto a 35 mm top-hat rail.

### 16.4 Dimensional drawing DIN-rail mounting

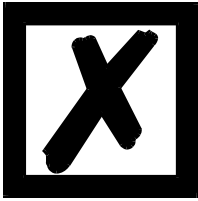


### 16.5 Commissioning

It is essential that you perform the following steps during commissioning in order to ensure that the module operates correctly:

### 16.6 Setting the CANopen® address and Baud rate

Set the CANopen® address and the Baud rate at the fieldbus end of the module on the DIP-switch.

**Attention:**

***The CANopen® address set must correspond to the planned address!  
All users in the CANopen® have to use the same Baud rate!  
These values are read in only on power-up of the gateway!***

**16.7 CANopen® connection**

Connect the device to the CANopen® at the interface labelled "CANopen".

**16.8 Connection to the process device**

Please also read the User Manual provided for the process device when commissioning the process device.

**16.9 Connecting the supply voltage**

Please connect 10.8...30 DC voltage to the terminals provided for this.

**16.10 Shield connection**

Connect the PE wire at the terminal provided for this. Earth the top-hat rail onto which the module has been snapped.

## 17 Servicing

Should questions which are not covered in this Manual crop up, please contact us directly.

Please note down the following information before calling:

- Device designation
- Serial number (S/N)
- Article number
- Error number and error description

You can reach us during hotline hours which are as follows

Monday to Thursday from 8 am to midday and from 1 pm to 4 pm, Friday from 8 am to midday.

Deutschmann Automation GmbH & Co. KG  
Carl-Zeiss-Straße 8  
D-65520 Bad-Camberg  
Germany

Central office and sales department +49-(0)6434-9433-0  
Technical hotline +49-(0)6434-9433-33

Fax sales department +49-(0)6434-9433-40  
Fax technical hotline +49-(0)6434-9433-44

E-mail technical hotline [hotline@deutschmann.de](mailto:hotline@deutschmann.de)

### 17.1 Returning a device

If you return a device, we require as comprehensive a fault/error description as possible. We require the following information in particular:

- What error number was displayed?
- What is the supply voltage ( $\pm 0.5$  V) with gateway connected?
- What were you last doing or what last happened on the device (programming, error on power-up, ...)?

The more precise information a fault/error description you provide, the more exactly we will be able to pinpoint the possible causes.

### 17.2 Downloading PC software and GSD files etc.

You can download the current version of WINGATE<sup>®</sup> free of charge from our Internet server.

<http://www.deutschmann.de>

## 18 Annex

### 18.1 Explanations of the abbreviations

#### General

CL	=	Product group CL (Compact Line)
CX	=	Product group CX
GT	=	Galvanic separation RS-side
GY	=	Housing color gray
RS	=	Product group RS
SC	=	Product group SC (Script)
232/485	=	Interface RS232 and RS485 switchable
232/422	=	Interface RS232 and RS422 switchable
DB	=	Additional RS232 DEBUG-interface
D9	=	Connection of the RS through 9-pin D-SUB instead of 5-pin screw-plug connector
PL	=	Board only without DIN-rail module and without housing cover
PD	=	Board only without DIN-rail module and with housing cover
AG	=	Gateway installed in a die-cast aluminum housing
EG	=	Gateway installed in a stainless steel housing
IC	=	Product group IC (IC-design DIL32)
16	=	Script memory expanded to 16KB
5V	=	Operating voltage 5V
3,.3V	=	Operating voltage 3.3V

#### Fieldbus

ASI	=	AS-Interface (AS-i)
CO	=	CANopen
C4	=	CANopen V4
C4X	=	CANopen V4-version X (see comparison table UNIGATE® IC for the respective product)
DN	=	DeviceNet
EC	=	EtherCAT
EI	=	Ethernet/IP
FE	=	Ethernet 10/100 MBit
FEX	=	Ethernet 10/100 MBit-version X (see comparison table UNIGATE® IC for the respective product)
IB	=	Interbus
IBL	=	Interbus
LN62	=	LONWorks62
LN512	=	LONWorks512
MPI	=	Siemens MPI®
PL	=	Powerlink
PN	=	Profinet-IO
PBDP	=	ProfibusDP
PBDPL	=	ProfibusDP-version L (see comparison table UNIGATE® IC for the respective product)
PBDPX	=	ProfibusDP-version X (see comparison table UNIGATE® IC for the respective product)
PBDPV0	=	ProfibusDPV0
PBDPV1	=	ProfibusDPV1
RS	=	Serial RS232/485/422

## 18.2 Hexadecimal table

Hex	Decimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

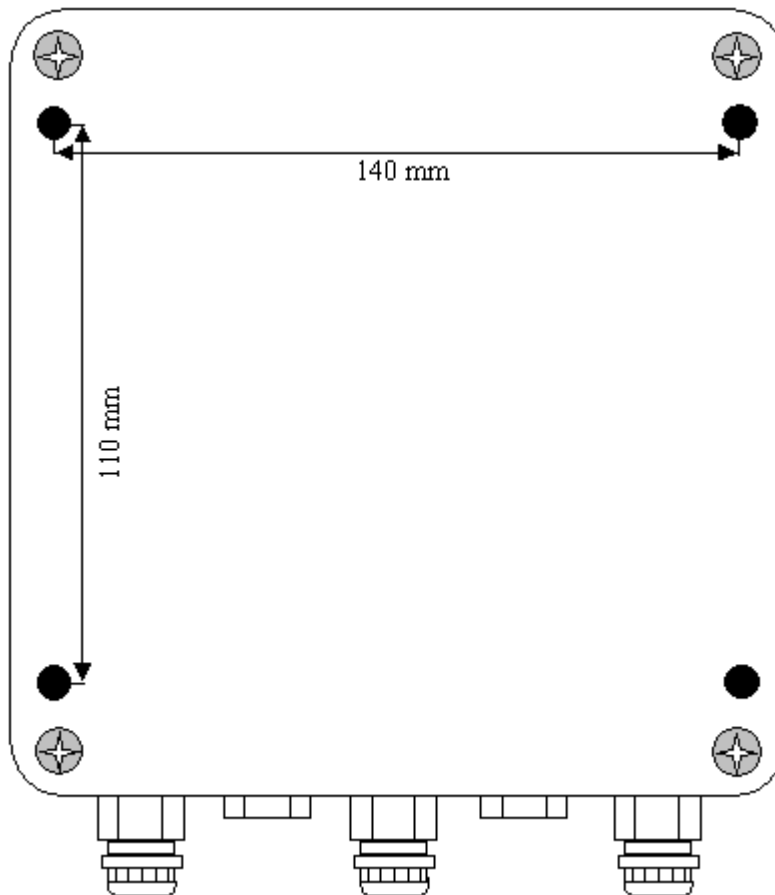


## 19 Drilling templates

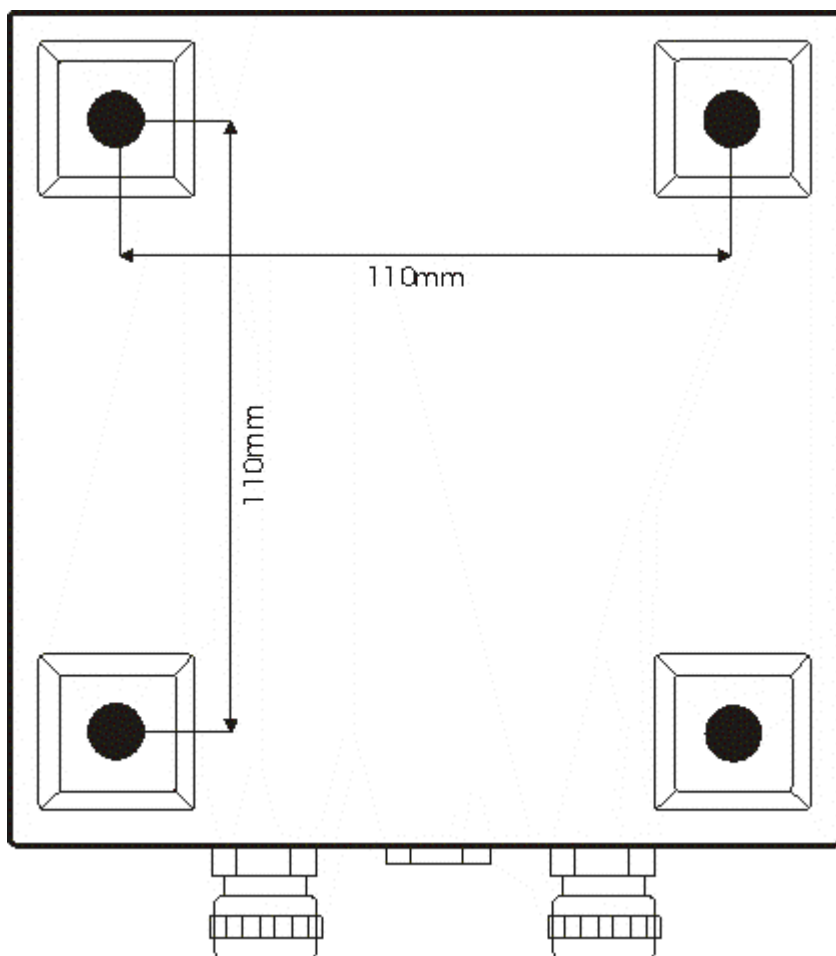


The following drawings are not true to scale!

### 19.1 Aluminium housing

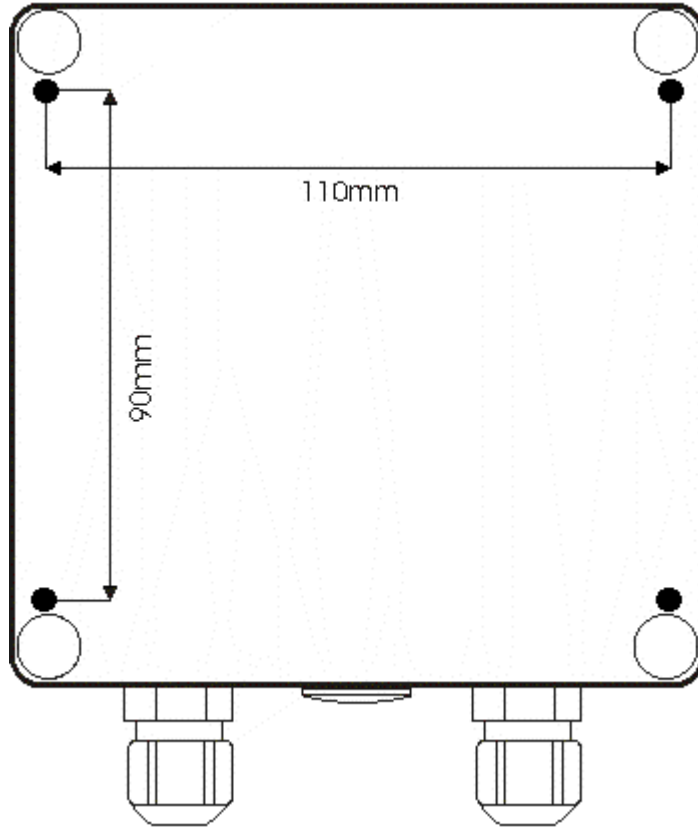


### 19.2 High-grade steel housing



### 19.3 Polycarbonate housing

! Supplied until January 2006 !



## 20 Declaration of conformity and certificates

### 20.1 EG Certificate of Conformance

#### EG Conformity Certification As defined by EG-EMV guideline (92/31/EWG)

Herewith we,

company **Deutschmann-Automation GmbH**  
**Max-Planck-Straße 21**  
**D-65520 Bad Camberg**  
**Tel: + 49 (0)6434 / 9433-0**  
**Fax: + 49 (0)6434 / 9433-40**

declare that the product mentioned afterwards, was developed, produced and issued in accordance with the above mentioned EG guideline.

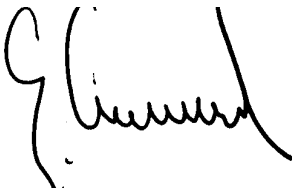
Product: Gateway

Model name: *UNIGATE® RS-CANopen*

Used norms:

<b>EN61000-6-2</b>	<b>Interference immunity</b>
EN61000-4-2	ESD
EN61000-4-3	Insolation E-field
EN61000-4-4	Burst
EN61000-4-6	Induction
<b>EN55011-A</b>	<b>Emission</b>
EN55011-A	Interference voltage
EN55011-A	Interference radiation

Bad Camberg, 05.05.2003




Gunther Lawaczeck



Michael M. Reiter

20.2 Certificate Germanischer Lloyd





Germanischer Lloyd

## Type Approval Certificate


This is to certify that the undemoted product(s) has/have been tested in accordance with the relevant requirements of the GL Type Approval System.


Certificate No.	42 407 - 01 HH
Company	Deutschmann Automation GmbH Max-Planck-Straße 21 D-65520 Bad Camberg
Product Description	Universal fieldbus-gateways
Type	Unigate RS232/485-AG; -CO(V3028); -DN(V3029); -IB(V3030); -PB(V3031)
Environmental Category	D
Technical Data / Range of Application	Power supply: 24 VDC Componentry for interfacing RS232/RS485/RS422 link with fieldbus.  Supported serial communication protocols: (optional potential free) - MODBUS-RTU, 3964R, control of character delay time, start/stop character, constant data length (transparent mode), customer specific protocols.  Supported fieldbus communication: (potential free) - CANopen V. 2.0 with CAN-controller SJA 1000 - CO(V3028) - DeviceNet V. 2.0 with CAN-controller SJA 1000 - DN(V3029) - INTERBUS V. 2.0 with SuPI 3 -IB(V3030) - Profibus-DP V.2.0 with ASIC SPC3 -PB(V3031)  Degree of protection: IP24 top hat rail version IP66 casing version
Test Standard	Regulations for the Performance of Type Tests 1997 Regulations for the Use of Computer and Computer Systems 1994
Documents	Test report : EMC Test Haus Dr. Schreiber 17/189/ 190/ 191/ 257/ 258; TÜV Rheinland E2111072 E 02, 968/U 123/2001; Design manual hardware and Software; Software Questionnaire according to requirement class 3 - dated 2001-09-17. System documents according to submitted files.
Remarks	None
Valid until	2006-12-26

Page 1 of 1 Type Approval Symbol  

File No. LB.06

Hamburg, 2001-12-27

  
 J. Wittburg

  
 T. Huthoff

This certificate is issued on the basis of "Regulations for the Performance of Type Tests, Part 0, Procedure".