

**Deuschmann**  
*your ticket to all buses*

## **Instruction Manual Universal Fieldbus-Gateway UNIGATE<sup>®</sup> IC - Powerlink**



Manual Art.-No.: V3705E



<b>1</b>	<b>General introduction</b>	<b>9</b>
<b>2</b>	<b>The UNIGATE® IC</b>	<b>10</b>
2.1	Technical introduction	10
2.2	Availability	10
2.3	Firmware	10
2.4	The serial standard interface	10
2.5	The synchronous interface	10
2.6	The Debug-interface	10
2.7	UNIGATE® IC hardware survey	11
<b>3</b>	<b>Hardware design.</b>	<b>12</b>
3.1	Ports	12
3.2	Pinout	12
3.2.1	-Boot enable	13
3.2.2	Load out (SPI-Master: SS0-)	13
3.2.3	Data out (SPI-Master: SS1-)	13
3.2.4	Data In (SPI: MISO)	13
3.2.5	Load In (SPI: MOSI)	13
3.2.6	Clock	13
3.2.7	-Reset In	14
3.2.8	LED-PL	14
3.2.9	-Config Mode	14
3.2.10	DbgTX, DbgRx	14
3.2.11	TE	14
3.2.12	TX, RX	14
3.3	Software	14
3.4	Basic line of proceeding	15
3.5	Connection examples	15
3.6	Layout examples	18
3.7	Handling (mounting the UNIGATE® IC on the carrier board)	21
3.8	Requirement for Ethernet-Interface	22
<b>4</b>	<b>The serial interface</b>	<b>23</b>
4.1	Overview	23
4.2	Initialization of the serial interface	23
4.3	Use of the serial interface	23
4.4	Further operation modes	23
<b>5</b>	<b>Synchronous serial interface</b>	<b>24</b>
5.1	Shift register operation	24
5.1.1	Example-Script	24
5.2	SPI mode	25
5.2.1	Example-Script	25

---

<b>6</b>	<b>The Debug-interface</b>	<b>26</b>
6.1	Overview of the Debug-interface	26
6.2	Starting in the Debug-mode	26
6.3	Communication parameter for the Debug-interface	26
6.4	Possibilities with the Debug-interface	26
6.5	Commands of the Debug-interface	26
<b>7</b>	<b>Script and configuration</b>	<b>27</b>
7.1	Overview	27
7.2	The configuration mode	27
7.3	Update the script	27
7.4	Configuration of the UNIGATE® IC	27
7.4.1	Ethernet	27
7.4.2	RS232/RS485/RS422	28
<b>8</b>	<b>Generating a script</b>	<b>29</b>
8.1	What is a script?	29
8.2	Memory efficiency of the programs	29
8.3	What can you do with a script device?	29
8.4	Independence of buses	29
8.5	Further settings at the gateway	29
8.6	The use of the Protocol Developer	30
<b>9</b>	<b>FTP-Server</b>	<b>31</b>
9.1	Script-update via FTP	31
9.2	System configuration update via FTP	31
<b>10</b>	<b>WEB-Server</b>	<b>32</b>
<b>11</b>	<b>Powerlink</b>	<b>34</b>
11.1	Data exchange mode	34
11.2	Powerlink address-assignment	34
11.3	Project planning	34
11.4	Possible data lengths	34
11.5	Literature	34
<b>12</b>	<b>Error handling at UNIGATE® IC</b>	<b>35</b>
<b>13</b>	<b>Firmware-update</b>	<b>36</b>
13.1	Overview	36
13.2	Execution of the firmware-update	36
13.3	Adjusting the firmware-update-mode	36
13.3.1	Adjustment by hardware	36
13.3.2	Adjustment by software	36
13.4	Note on safety	36

13.5	Operation mode of the IC . . . . .	36
<b>14</b>	<b>Technical data . . . . .</b>	<b>37</b>
14.1	Mechanics of the UNIGATE® IC . . . . .	37
14.1.1	General dimensions of UNIGATE® IC . . . . .	37
14.1.2	Dimensions UNIGATE® IC (Powerlink only) . . . . .	37
14.2	Technical data UNIGATE® IC - Powerlink . . . . .	38
<b>15</b>	<b>Accessory . . . . .</b>	<b>39</b>
15.1	Adapter RS232 . . . . .	39
15.2	Adapter RS485 . . . . .	39
15.3	FirmwareDownloadTool (FDT) . . . . .	39
15.4	Protocol Developer . . . . .	39
15.5	Developerkit UNIGATE® IC-AB IC . . . . .	39
15.5.1	Developerboard UNIGATE® IC-AB . . . . .	39
15.5.2	Quick start . . . . .	40
<b>16</b>	<b>Appendix . . . . .</b>	<b>41</b>
16.1	Explanations of the abbreviations . . . . .	41
16.2	Basis board . . . . .	42
16.2.1	Overview basis board for PN, EI + PL 3.3V . . . . .	42
16.2.2	Connectors of the basis board. . . . .	46
16.2.2.1	Connector to the external device (RS-interface). . . . .	46
16.2.2.2	Powerlink connector . . . . .	47
16.2.2.3	Power supply of the basis board. . . . .	47
16.2.2.4	Shield terminal lead . . . . .	47
16.2.2.5	Rotary coding switches . . . . .	47
16.2.2.6	Slide switch (RS485/RS232 interface). . . . .	47
16.2.2.7	Slide switch (RS485 termination) . . . . .	47
<b>17</b>	<b>Servicing . . . . .</b>	<b>49</b>
17.1	Downloading PC software, XDD-file, Script examples etc. . . . .	49



#### 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 – 2015. 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.





## 1 General introduction

In the past the integration of a fieldbus connection required an enormous effort from the progress engineers. On account of the large variety of communication systems it is not enough to compile the right combination of communication hardware; due to their standards and fundamentals different busses also require the corresponding skills of the engineers.

This does not apply in case of the UNIGATE® IC by Deutschmann Automation any more. All digital functions, software, stack and driver as well as optocoupler are integrated on a UNIGATE® IC in correspondence with the standard. In addition to the reduction of the required size, also different fieldbuses can easily be integrated.

**Through the flexible firmware of UNIGATE® IC no software-changes are required on the side of the customer!**

Since 1997 Deutschmann Automation has experience in the field of fieldbus gateways; this enormous experience results in the UNIGATE® IC as a consistent sequel of this successful product line.

### **Terminology**

In the entire document and in all parts of the software that is to be used, the terms Input and Output are used. Input and Output are ambiguous, always depending on the viewpoint. We see the fieldbus as central interface and as integral component of your device; therefore in all places it is always referred to data from the viewpoint of the Slave, that is Input data, as data from the Master to the Slave - regardless of the used bus.

### **Representation of numbers**

Numbers in decimal format are always represented without prefix and without suffix as well. Hexadecimal numbers are always marked with the prefix 0x.

## 2 The UNIGATE® IC

### 2.1 Technical introduction

The UNIGATE® IC by Deutschmann Automation contains all components that are required for the communication in a fieldbus in one single module. Therefore a developer does not have to take care for that detail any more, only a hardware redesign is necessary in order to integrate the UNIGATE® IC and the required plug connectors.

### 2.2 Availability

The module is available as Powerlink. Further fieldbusses are either planned or being worked on. They will only differ in the connections of the busses. The meaning of the general pins 1 - 9 as well as 24 and 26 - 32 will remain unchanged also for further fieldbus implementations.

You can find an up-to-date list for all UNIGATE® ICs under:  
<http://www.deutschmann.de>.

### 2.3 Firmware

UNIGATE® IC is programmed via scripts. On principle any script, that has been developed for a UNIGATE® IC, can also be operated on the UNIGATE® IC.

### 2.4 The serial standard interface

Intelligent devices, that already feature a micro controller or a microprocessor, are generally supplied with a serial asynchronous interface with a TTL-level. It is directly connected with the TTL-interface of the UNIGATE® IC. For more information on this serial interface see chapter 4 on page 23.

### 2.5 The synchronous interface

In addition to the standard interface there is also the possibility of the synchronous input and output. That way for instance digital IOs can be connected through shift register components or also analog IOs can be connected through a DA-converter with serial in-/output. For synchronous IOs 256 signals at the most can be used (256 bit). Wiring examples can be found in chapter 3.5 and for software examples see chapter 5. This interface can also be used to connect modules or devices with SPI-interface. It is also possible to build, for instance digital or analogous I/O-modules, with the customer's device not being equipped with an own controller. The fieldbus IC is also operable autonomously without that controller.

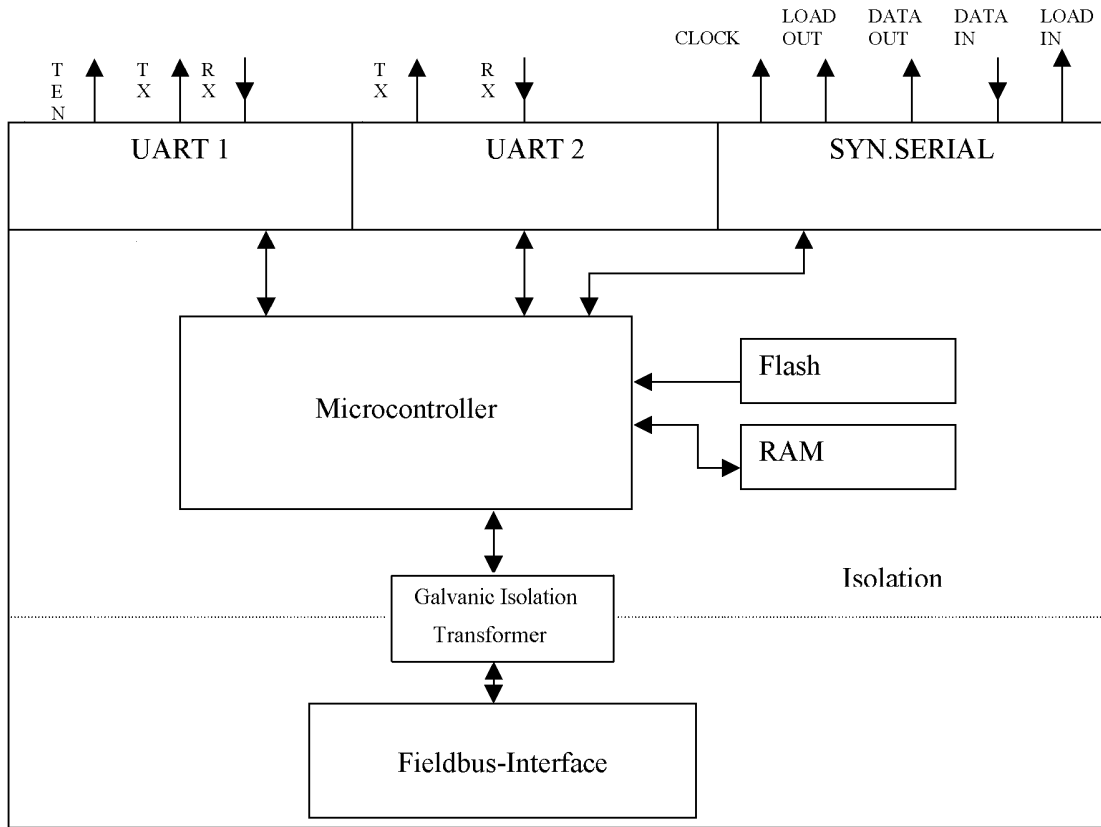
### 2.6 The Debug-interface

The UNIGATE® IC features a Debug-interface, which allows to process a script step by step and also to monitor or manipulate data. This is indispensable for the development of a script. Usually a script is developed with the software Protocol Developer. For more details take a look at the instruction manual Protocol Developer.

**All interfaces can independently be used at the same time.**

## 2.7 UNIGATE® IC hardware survey

The hardware of the UNIGATE® IC consists of some few standard components. The picture below shows the functional structure of the IC.



### 3 Hardware design

This chapter gives basic advise, that is required in order to load UNIGATE® IC into your own hardware designs. In the following all ports of UNIGATE® IC are described in detail.

#### 3.1 Ports

UNIGATE® IC features 32 pins in its layout as a DIL 32 component. Pin 10 -12 and 21 - 23 as well are not wired due to the electrical isolation. The exact mechanical dimensions can be taken from chapter 14 on page 37.



In the layout boreholes for ALL 32 pins have to be planned.

#### 3.2 Pinout

Pin	Technical specifications	Name	Description	Remark
1*	3.3V ± 5% < 500mA	Vcc	+3.3V voltage supply	
2	IN <sub>Logic</sub>	-BE	boot enable	
3	OUT <sub>Driver</sub>	Load out (SS0-)	strobe signal for synchronous, serial interface	
4	OUT <sub>Driver</sub>	Data out (SS1-)	output data for synchronous, serial interface	
5	IN <sub>Logic</sub>	Data in (MISO)	input data of the synchronous, serial interface	internally pulled up with 10 kΩ
6	OUT <sub>Logic</sub>	Load in (MOSI)	input data of the synchronous, serial interface; alternatively strobe signal of the output data	
7	OUT <sub>Driver</sub>	Clock (SCK)	clock pulse signal for synchronous, serial interface	
8	IN <sub>Reset</sub>	-Reset in	reset-input of the IC	internally pulled up with 100 kΩ
9*	connected to pin 1	Vcc	+3.3V voltage supply	
10-12	nc	nc	no pin	
13	according to norm	RJ45_6	Ethernet-signal RD-	galvanically isolated insulation voltage 1000 Vrms
14	according to norm	RJ45_3	Ethernet-signal RD+	galvanically isolated insulation voltage 1000 Vrms
15	according to norm	RJ45_2	Ethernet-signal TD-	galvanically isolated insulation voltage 1000 Vrms
16	according to norm	RJ45_1	Ethernet-signal TD+	galvanically isolated insulation voltage 1000 Vrms
17	according to norm	LED-LINKACT	Ethernet linkpulses found / network data traffic	
18	according to norm	nc	not connected	
19	according to norm	nc	not connected	
20	according to norm	Ground (PE)	earth	
21-23	nc	nc	no pin	
24*	connected to pin 32	GND	ground supply voltage of the IC	
25	*)	LED-PL	bus status LED	

26	IN <sub>Logic</sub>	-Config Mode	signal to start the configuration mode	internally pulled up with 10 kΩ
27	OUT <sub>Logic</sub>	DbgTX	serial Debug TX	
28	IN <sub>Logic</sub>	DbgRX	serial Debug RX	internally pulled up with 10 kΩ
29	IN <sub>Logic</sub>	RX	serial data RX	internally pulled up with 10 kΩ
30	OUT <sub>Logic</sub>	TX	serial data TX	
31	OUT <sub>Logic</sub>	TE	transmit enable	
32*	GND	GND	ground supply voltage of the IC	

\* The supply voltage is 3.3V +/- 5 %, max. 500 mA DC.

The Ethernet signals are galvanically isolated. The insulation voltage is 1000 Vrms.

\*) Anode of the green LED

<b>3.3V ± 5%, &lt; 500mA</b>	V <sub>IL</sub>	V <sub>IH</sub>
IN <sub>Reset</sub>	< 0.25V / 1mA	>1.5V / 10μA
IN <sub>Logic</sub>	< 0.5V / 0.4mA	>1.5V / 10μA
	V <sub>OL</sub>	V <sub>OH</sub>
OUT <sub>Logic</sub>	< 0.6V / 1mA	>2.3V / 0,1mA
OUT <sub>Driver</sub>	< 0.5V / 4mA	>2.5V / 4mA

### 3.2.1 -Boot enable

The IC is started in the firmware update mode with the level GND during the power up process. See also chapter 13 on page 36.

### 3.2.2 Load out (SPI-Master: SS0-)

Strobe signal for the synchronous serial interface. With the positive edge at this output data is taken from the connected shift registers to the physical outputs.

In SPI mode this Pin serves as a low-active Slave-Select-Signal.

### 3.2.3 Data out (SPI-Master: SS1-)

On this line data is output on the synchronous serial interface. The most significant bit of the data is output first.

In SPI mode this Pin serves as a low-active Slave-Select-Signal.

### 3.2.4 Data In (SPI: MISO)

Data is read in on the synchronous serial interface via this signal. The most significant bit of the data is expected first.

In SPI mode this Pin serves as data transfer from Slave to Master.

### 3.2.5 Load In (SPI: MOSI)

This pin is the strobe signal for the input data of the synchronous serial interface.

In SPI mode this Pin serves as data transfer Master to Slave.

### 3.2.6 Clock

This signal is the clock line for the synchronous serial interface. That signal is equally valid for data input and data output.

### 3.2.7 -Reset In

- A reset generator (Max 809) is on board; with it in the normal case the reset input is not required. In this case the reset input has to be connected with VCC, in order to avoid interferences (see chapter 3.6).
- If the the customer's application has to initiate a reset of the UNIGATE® IC, then the reset input can also be connected with a reset output of the customer's application instead of connecting it with VCC. Here all specifications of the reset signal, mentioned in chapter 3.2 have to be kept. The reset-impulse is supposed to last at least 10 ms.

### 3.2.8 LED-PL

A green LED can be connected to this line (see chapter 3.6). It is glowing in the state of "Ethernet ok" and in the state "Waiting for link connection" it is flashing.

### 3.2.9 -Config Mode

If the pin has the level GND, then the IC starts in the configuration mode.

### 3.2.10 DbgTX, DbgRx

They are transmission line and receive line as well of the IC's Debug-interface. For the function description of the Debug-interface see chapter 6 on page 26.

### 3.2.11 TE

The transmit enable signal allows the connection of RS485 drivers to the IC's serial interface. The signal is set to High whenever the IC sends via the line TX.

### 3.2.12 TX, RX

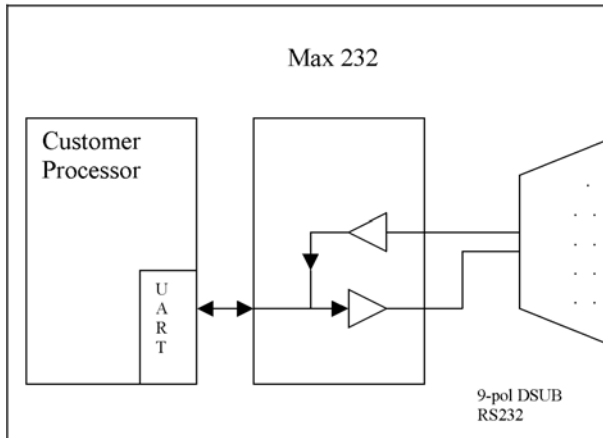
Transmission and receive line of the serial interface. This interface is programmable in accordance with the description in chapter 4 on page 23.

## 3.3 Software

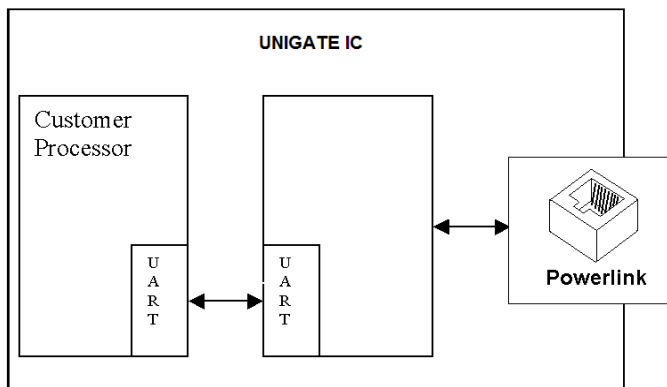
The software executes script-commands, which in turn control the IC's hardware and they process their complete protocol by software. The script itself can be generated by the company Deutschmann Automation or with the software Protocol Developer by yourself. For a detailed description of the script.commands of the Protocol Developer see the instruction manual Protocol Developer and the online documentation concerning script-commands.

### 3.4 Basic line of proceeding

In theory it is enough to replace the RS232-driver that is included in your application by the UNIGATE<sup>®</sup> IC.



Your device, which on the whole is supposed to be assembled as shown above, will now be modified in a way that the Powerlink is available at the 9-pol. socket. However, a hardware redesign is necessary in order to keep the assignment in standard form.

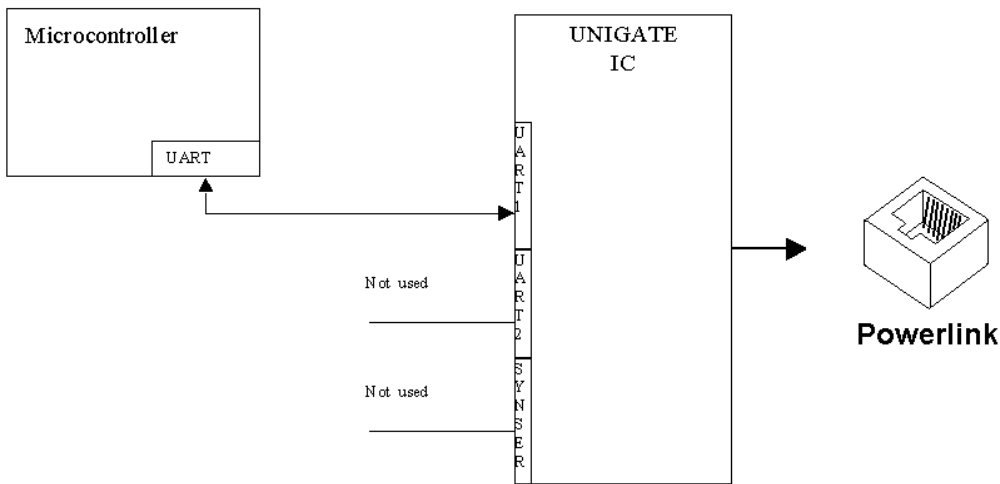


After the RS232-driver has been replaced by the UNIGATE<sup>®</sup> IC, Powerlink is available at an RJ45-socket.

### 3.5 Connection examples

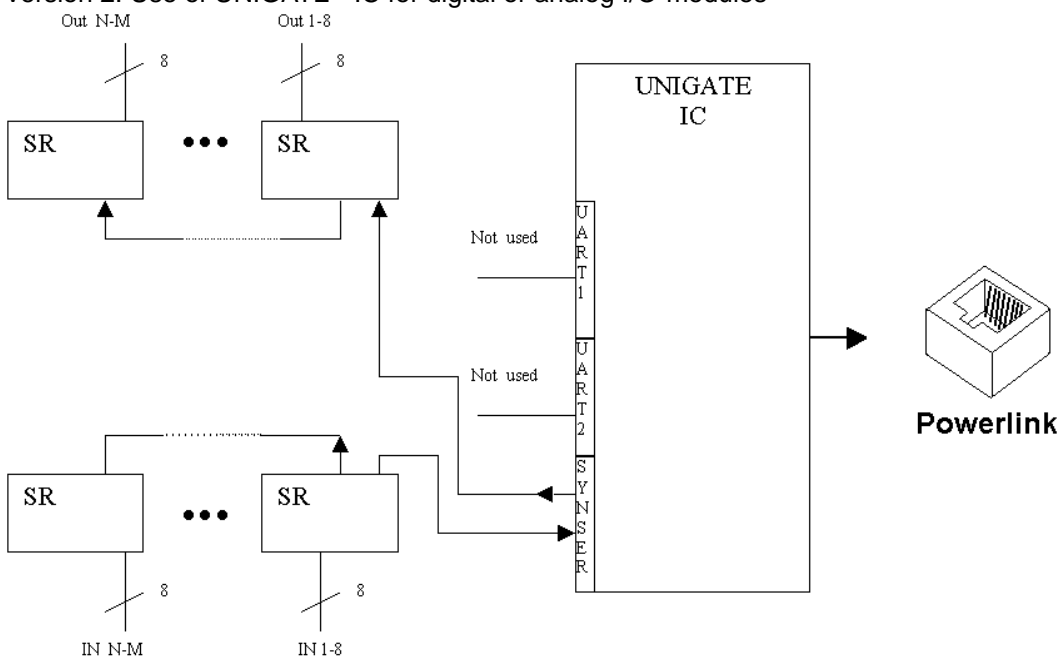
Here you will find some advise that offers help for a re-design. In the following several versions are listed, that should make it easier for you to decide.

Version 1: Use as a pure link module for the bus



The UNIGATE<sup>®</sup> IC independently processes the communication with the customer's device via the TTL-interface.

Version 2: Use of UNIGATE<sup>®</sup> IC for digital or analog I/O-modules

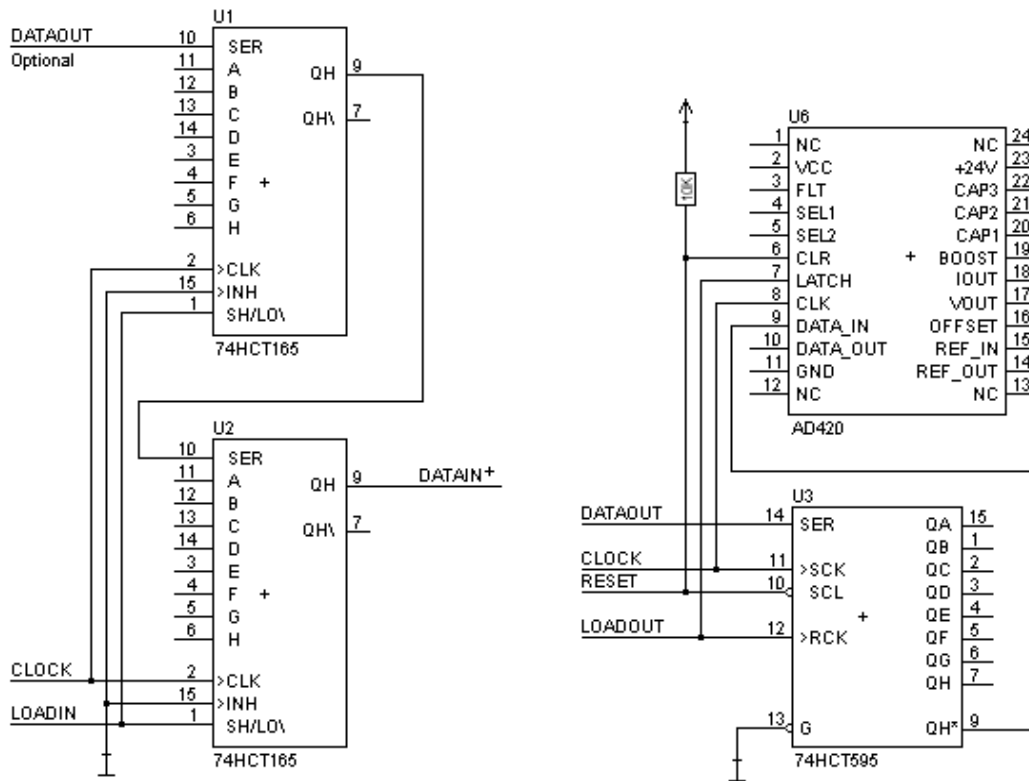


Here only the synchronous serial interface is used, the asynchronous serial interface is basically of no account. If you want to program the script in your completed application, then the use of a connector for the asynchronous interface is recommended. With it you can carry out the ISP-programming.

**For this operating mode no additional controller is required on your application!**



The following circuit diagram is an example for how shift register components can be connected to the IC.

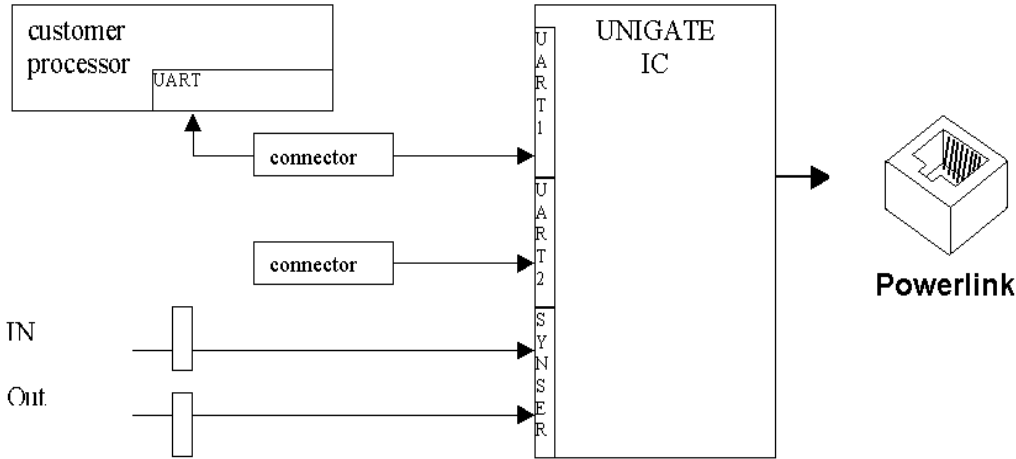


Version 3: Example for digital I/Os

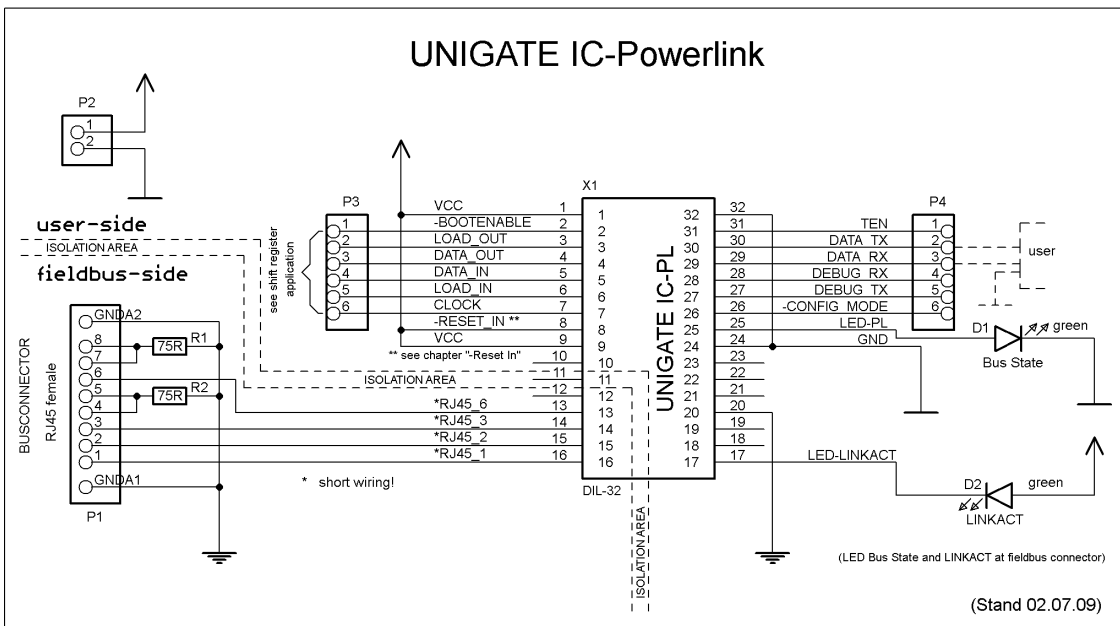
The serial synchronous and the asynchronous interface as well can be operated by UNIGATE<sup>®</sup> IC at the same time. Here the possibility results that an existing application can be extended by additional digital or analog I/Os.

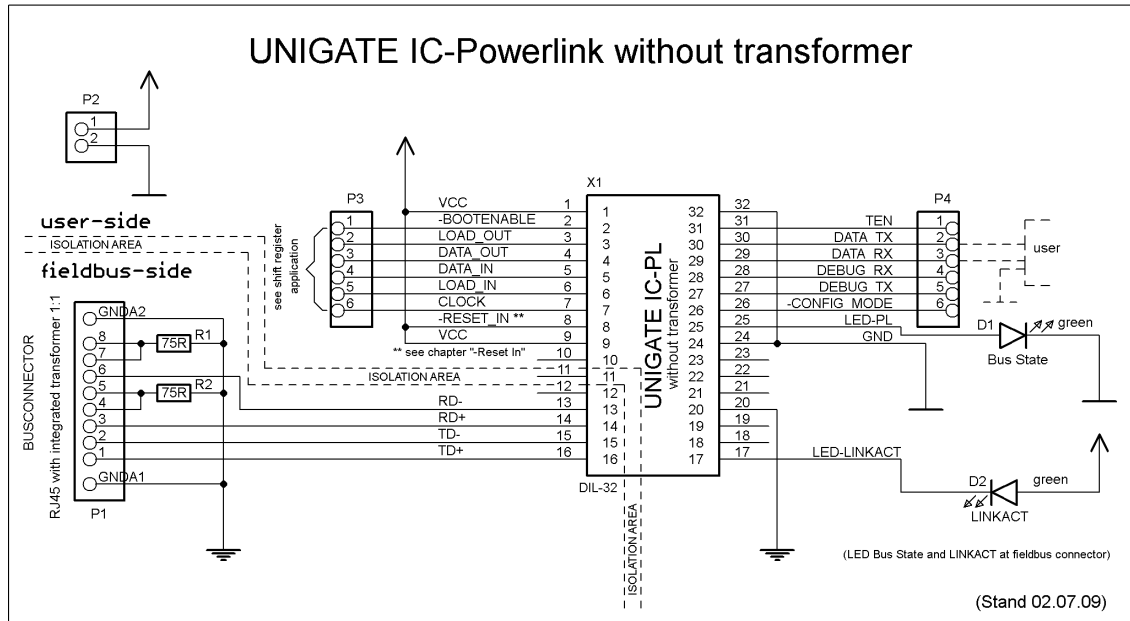
In chapter 5.2 you find an example for a script, that operates these I/Os.

Valid for all versions: A planed plug connection of the serial interface in the application offers the possibility of an update of the firmware or the software via an external connection.

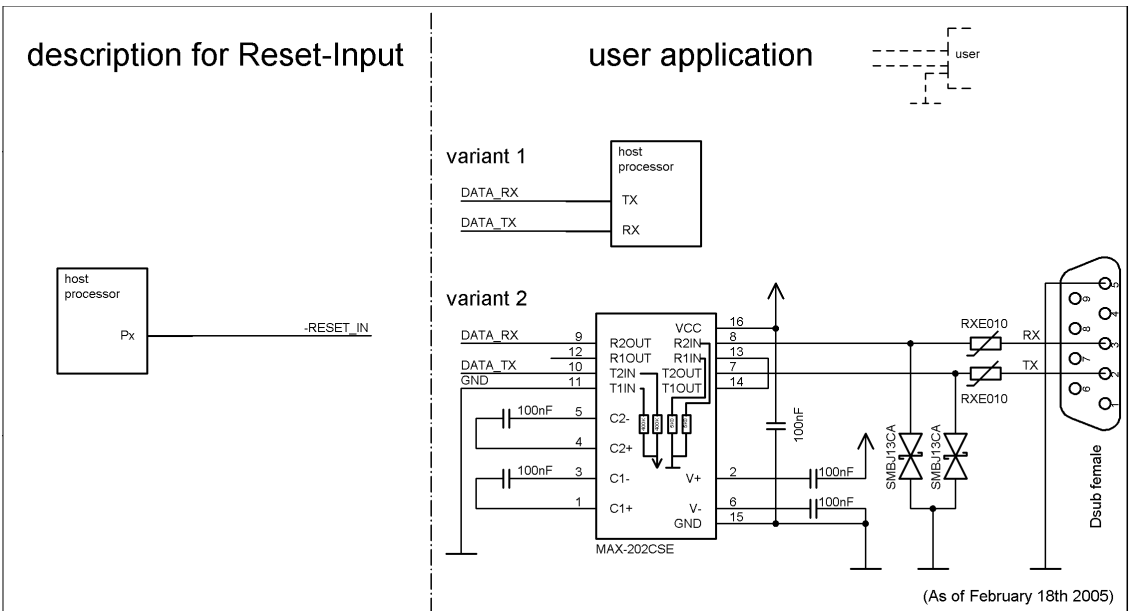


### 3.6 Layout examples

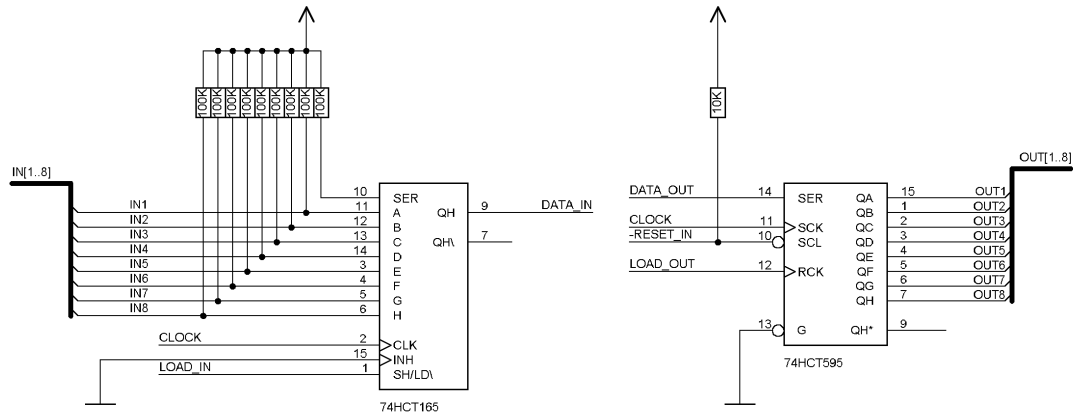




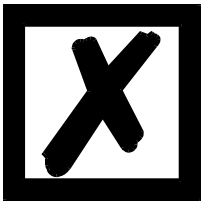
When connecting the UG-IC and the RJ45-connector it is important to make sure the signal lines RD-/RD+ resp. TD-/TD+ run as short as possible, parallel and equal in length. Because this is for analog signals, it is also advisable to not place digital lines in close proximity.



shift register application



(As of August 26th 2002)



The 74HCT595 used in this example has an undefined on-position, but therefor can set the outputs to the tri-state condition via the OutEnable-Pin 13. If it is more important to have a defined on-position in an application, and the OutEnable-pin is not necessary, the 74HCT594 can be used here.

### 3.7 Handling (mounting the UNIGATE® IC on the carrier board)

Depending on the application and the expected shock- and vibration-conditions you can choose from the following possibilities for the UNIGATE® IC's installation on the carrier board:

- Mounting on a socket in the carrier board. If necessary solder the UNIGATE® IC to 2 or 4 pins in the socket. Normally the IC can easily be pulled out after the soldering points have been removed.
- Make arrangements for two holes next to the socket in the layout. After the UNIGATE® IC was plugged in the socket pull an isolated wire over the IC and solder it on the carrier board at the specified holes.
- Fasten the UNIGATE® IC With a wire or a tie wrap on the socket.
- Manual soldering directly on the carrier board.
- Automatic soldering directly on the carrier board, whereas „selective“ soldering is essential (no wave soldering)

The advantage of the socketed variant is the easy download of Script- and Firmware-updates, if the carrier board is not designed for it. Besides, that way the Fieldbus can be changed easily by changing the UNIGATE® IC if the corresponding plug connectors are provided on the carrier board. Another advantage is, that - normally - only a reflow soldering of the carrier board is necessary.

The advantage of the soldered variant is, that the installation height is lower and a higher shock- and vibration-safety is provided.

### 3.8 Requirement for Ethernet-Interface

There is an extremely large number of suppliers for RJ45 connectors (with and without integrated Ethernet transformer), or single Ethernet transformers on the market, therefore it is not possible for us to test all possibilities. If you consider that there are various qualities (SMD / THT, with / without LED, vertical / horizontal, with / without EMC springs tab at the top / bottom, single / multiple times, etc.) with RJ45 connectors alone, it is probably understandable.

In General, RJ45 connectors and the external or built-in Ethernet transformer must meet the following properties:

#### **Required parameters for Ethernet Transformer:**

Turns ratio 1 CT: 1 CT  $\pm$  5%

Open Circuit Inductance 350  $\mu$ H (min) @ 10 mV, 10 kHz

Insertion Loss 1.1 dB (max)

1500 Vrms dielectric, 1 minute

#### **Required parameters for RJ45 connector:**

CAT5, shielded if possible

To enable a quick start in the selection of appropriate components, we have compiled a list of order numbers of Würth Elektronik below, where we source our components. Of course, other manufacturers are possible, so the following list is to be considered exemplary.

#### **Ethernet transformer 10/100 Base-T 1:1**

We mainly use: Order-No. 749 010 014

#### **RJ45 jack without transformer**

We mainly use: Order-No. 615 008 140 121

#### **RJ45 jack (single or multiple) with integrated transformer**

To be cost-effective and more flexible we don't use RJ45 connectors with integrated transformers. Technically, however, nothing speaks against it.

We exemplary tested: Order-No. 749 905 100 1

For more information:

We had bad experiences with SMT connectors in the past, because in the field they got demolished every now and then due to mechanical loads.

## 4 The serial interface

### 4.1 Overview

The serial interface is the most important connection between the UNIGATE<sup>®</sup> IC and the micro controller of your application. The interface is designed in a way so that your application at least does not have to be changed on the software-side. The wide range of services of the UNIGATE<sup>®</sup> IC's serial interface constitutes the basis for it. The UNIGATE<sup>®</sup> IC allows to connect of controllers with a baudrate of 110 baud to 625 kbaud. The baudrate for the communication itself is permanently stored in the module. The maximum size for IO-data can be read-out with the Script command "Get\_RSOutBufFree16".

Depending on the read-in Script of the UNIGATE<sup>®</sup> IC, the module carries out actions independently, in order to identify data from the connected device. For customers who already have a software-adaptation at the company Deutschmann Automation, this protocol as well or a Script after an adaptation can be processed by the IC.

Anyway, the IC will take over the communication with the fieldbus independently.

### 4.2 Initialization of the serial interface

The initialization of the interface is carried out by script-commands, such as "Set baudrate", "Set databits", "Set parity". For a detailed description of these commands see the online documentation for the Protocol Developer or the instruction manual for the Protocol Developer.

### 4.3 Use of the serial interface

The serial interface can freely be programmed by the user. Efficient script-commands for sending and receiving data are available; just to mention some possibilities: such as waiting with timeout for a character, waiting for a fixed number of characters or also sending and receiving data in the Modbus RTU. A reference to these commands is offered in the online documentation for the Protocol Developer as well as in the instruction manual for the Protocol Developer.

### 4.4 Further operation modes

In the configuration mode the serial interface also serves to configure the standard software or to carry out a Script-update. More details on that can be found in chapter 7.

## 5 Synchronous serial interface

The synchronous serial interface of the UNIGATE® IC is used to connect clocked shift registers or components that have a Serial Peripheral Interface (SPI).

It allows

- the expansion of the IC for digital inputs and outputs (for example for driving LEDs or for reading switch positions)
- communicate with microcontrollers or
- the control of DA and AD converters.

Connection examples are give in chapter 3.

By using the synchronous serial interface can realize products that can work without another microcontroller (stand-alone mode). Examples are sensor products or digital IO modules

### 5.1 Shift register operation

Before the interface can be used, it has to be initialized by setting various Script parameters. (see chapter 5.1.1)

The parameters „ShiftRegisterInputType“ and „ShiftRegisterOutputType“ allow the use of different shift register types, which differ in the polarity of the shift register signals. To use the shift register types 74595 and 74165, for example, the values „RiseClk\_RiseLoad“ and „RiseClk\_LowLoad“ can be set.

The shift register width is set by the parameter „ShiftRegisterInputBitLength“ and „ShiftRegisterOutputBitLength“ The maximum width is 256 bits.

The data exchange with the connected shift registers ensues with the commands „WriteShiftRegister“, „ReadShiftRegister“ or bidirectional with the command „ShiftRegisterDataExchange“. The clock rate is between 280 kHz and 320 kHz.

Further information on the commands and parameter values can be found in the Help section of the Protocol Developer Software. On request, the Deutschmann Script language can be complemented by additional parameter values in order to support other types of shift registers.

#### 5.1.1 Example-Script

Note: The script example refers to the circuit example in chapter 3.5.

```
var InBuffer: Buffer[2];
var OutBuffer: Buffer[2];
MoveConst( OutBuffer[0], #0x58#0x21 );

Set( ShiftRegisterInputType, RiseClock_FallLoad );
Set( ShiftRegisterOutputType, RiseClock_RiseLoad );
```



```
Set( ShiftRegisterInputBitLength, 16 );
Set( ShiftRegisterOutputBitLength, 16 );

WriteShiftRegister( OutBuffer[0] );
ReadShiftRegister( InBuffer[0] );

// Input data is now in the INBuffer
// 0x58 is applied to the outputs of the analog converter
// 0x21 at the shift register's outputs
```

## 5.2 SPI mode

Before the interface can be used in SPI mode, this must be initialized. The command `InitSPI` sets the operating type, the mode (signal polarity and phase) and the clock frequency.

The data exchange ensues with the command `ExchangeSPI`. The maximum clock frequency is between 1 and 5 MHz, depending on the hardware. For details please see the IC-Pinout list in the download area of our website.

Please also refer to the script commands documentation in the online help of the Protocol Developer.

### 5.2.1 Example-Script

```
var L_Freq      : long;
var b_Channel   : byte;
var w_Len       : word;
var a_BufOut    : buffer[100];
var a_BufIn     : buffer[100];

moveconst( L_Freq, 1000000); // 1 MHz
InitSPI( 1 , 0 , L_Freq );

moveconst( b_Chanel, 0 );
moveconst( w_Len, 11 );
moveconst( a_BufOut[0], "Hello World" );
ExchangeSPI( b_Channel , w_Len , a_BufOut[0] , a_BufIn[0] );
```

## 6 The Debug-interface

### 6.1 Overview of the Debug-interface

The UNIGATE<sup>®</sup> IC features a Debug-interface, that allows a step-by-step processing of a script. Normally this interface is only required for the development of a script.

### 6.2 Starting in the Debug-mode

When applying power to the UNIGATE<sup>®</sup> IC (power up) the firmware will output the binary character 0 (0x00) after a self-test was carried out on this interface. If the IC receives an acknowledgement via this interface within 500 ms, it is in the Debug-mode. The acknowledgement is the ASCII-character O (0x4F).

With the start in the Debug-mode the further execution of script-commands will be put to a stop.

### 6.3 Communication parameter for the Debug-interface

The Debug-interface is always operating with 9600 baud, no parity, 8 data bit, 1 stop bit. It is not possible to change this parameter in the Protocol Developer. Please consider the fact that these settings have to be in accordance with those of the PC-COM-interface and that the flow control (protocol) has to be set on „none“ there.

### 6.4 Possibilities with the Debug-interface

Usually the Protocol Developer is connected to the Debug-interface. With it a step-by-step processing of a script, monitoring jumps and decisions and looking at memory areas is possible. Moreover breakpoints can be set. It basically possesses all characteristics a software-development tool is typically supposed to have. However, it is also possible to carry out a Scrip-update via this interface.

### 6.5 Commands of the Debug-interface

The commands for the use of the Debug-interface are described in the online-documentation for the Protocol Developer as well as in the instruction manual Protocol Developer.

## 7 Script and configuration

### 7.1 Overview

The script stored in the UNIGATE® IC, as well as the configuration, can be replaced or updated via the serial interface (application) in the configuration mode.

### 7.2 The configuration mode

If the pin „ConfigMode“ pulled to GND during the PowerUp or Reset, then the UNIGATE® IC starts in the configuration mode. In this mode it is possible to communicate with the IC without processing the regular software. It is also possible to change the UNIGATE® IC's settings of the standard software or to write a new script in the UNIGATE® IC. It shows its start in the configuration mode by issuing a status message (it may take up to 1 minute), which might look as follows:  
IC-PL-SC V0.31[30] (c)dA Script(16k)=""Leer" Author=""Deutschmann Automation GmbH"  
Version=""1.0" Date=21.08.2001 SN=47110001 IP=10.10.10.10 MAC=00-14-11-15-19-81

### 7.3 Update the script

- The preferred option is to insert the UNIGATE® IC into the base board from Deutschmann (Developer Board UNIGATE® IC-AB) and use the Deutschmann tools (software WINGATE with „Write Script“ under „File“ or with the software ScriptProgramTool).
- On request the procedures, how to overwrite the included script can be disclosed by Deutschmann Automation in order to automatically replace the script in an application.
- It is also possible to change the Script via FTP (see chapter 9.1 'Script-update via FTP').



**Note:**

**After a change (download of a new configuration or a new Script) it takes up to one minute until the IC has saved this change and will be ready for operation.**

### 7.4 Configuration of the UNIGATE® IC

UNIGATE® IC is delivered with an empty script.

The configuration of the UNIGATE® IC - Powerlink is restricted to the setting of the Ethernet-address.

#### 7.4.1 Ethernet

- Protocol (UDP/IP or TCP/IP)
- Baud rate: 10 and 100 MBaud autodetect
- IP-address (Gateway)
- Ethernet (MAC-address)
- IP-destination host

#### 7.4.2 RS232/RS485/RS422

- RS-type: RS232
  - Start bit: 1
  - Data bits: 8
  - Stop bit: 1
  - Parity: None
  - Baud rate: 9600 Baud
- Default setting. This configuration can be changed via the Script.

## 8 Generating a script

### 8.1 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.

### 8.2 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.

### 8.3 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.

### 8.4 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.

### 8.5 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.

## **8.6 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.

## 9 FTP-Server

This UNIGATE<sup>®</sup> features an integrated FTP-Server, that can be accessed with the file system. The FTP-Server is password protected and can be addressed via the user name "deutschmann" and the password "deutschmann".

The following files that are located there on this file system MUST NOT be deleted or changed in no case since they are mandatory for the system:

- project.hex
- ftp\_accounts.txt
- script.sys

The remaining files belong to the WEB-Server. Further information can be found in the corresponding chapter "WEB-Server".

### 9.1 Script-update via FTP

The dcs-file generated by the Protocol Developer has to be stored as "script.dcs" by the FTP on the Gateway (subdirectory "flash"). When the Gateway starts up, it identifies, converts and integrates this file into the file "script.sys", where the Script is usually filed and then deletes the file "script.dcs".

### 9.2 System configuration update via FTP

A WINGATE gwc-file has to be stored on the Gateway as "SYSTEM.GWC" via FTP. When the Gateway is startet, it identifies this file, transfers the new configuration and then deletes the file "SYSTEM.GWC".

## 10 WEB-Server

This Gateway also contains a WEB-Server by default. In the initial state there is a welcome page on the Flashdisk, that presents the device's system parameters.

The WEB-pages can be changed by the customer as desired and via FTP they can be written on the Flashdisk.

The "*Server Side Include*"-functionality is made available in order to be able to generate dynamic WEB-pages that display process data on the WEB-page or pass on data from the WEB-page to the process, which means that placeholders are used for the process data on the HTML-page.

### Data exchange through Server Side Include (SSI)

In case the UNIGATE detects the variable's placeholder, that is described by a variable's name (see below), it places the corresponding String at the variable's place. The other way round Strings, that are passed on via „POST“ by the HTML-page with the corresponding variable name, are copied in to the corresponding Script buffer.

The Syntax looks as follows:

```
<?--#exec cmd_argument='xxxxx'-->
```

For xxxxx the following expressions are possible:

- DisplayFWVersion
- DisplayBLVersion
- DisplaySerial
- DisplayMacID
- DisplayStationName
- DisplayStationType
- DisplayVendorID
- DisplayDeviceID
- DisplayIP
- DisplaySubnet
- DisplayGateway

These expressions are self-explanatory and they reflect the corresponding values that have been preset by the Firmware.

The following expression is required for the exchange of any process data between the Script in the Gateway and the WEB-page:

- DisplayScriptVar, Variablenname



A good example for the interaction between Gateway-Script and HTML-page can be taken from the Example-HTML-page "ssi.html" and the Example-Script "example\_Set\_HTML\_String.dss", that can both be found in the download area on our homepage at [www.deutschmann.com](http://www.deutschmann.com).

## 11 Powerlink

UNIGATE® IC-Powerlink serves to adapt a serial interface to Powerlink CN (Controlled Nodes). The end device's protocol is being converted in the UNIGATE® via a Script.

### 11.1 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. This mode is always active as long as the gateway is not in the configuration mode, the firmware update- or Debugmode. In the data exchange mode the gateway will carry out the downloaded Script.

### 11.2 Powerlink address-assignment

The Powerlink address can either be assigned through WINGATE or the Script.

### 11.3 Project planning

Use any project planning tool for project planning.

A sample XDD-file can be found on the Internet ([www.deutschmann.com](http://www.deutschmann.com)) free of charge. Generating the XDD-file is also provided as service from Deutschmann Automation.

### 11.4 Possible data lengths

In der folgenden Tabelle sind die maximal zu übertragenden Daten tabellarisch dargestellt:

Nachricht	Anzahl Nachrichten	Nachrichtenlänge
Rx PDO	1	max. 1514 Bytes
Tx PDO	1	max. 1514 Bytes
SDO	max. 256 Stück	max. Variablenlänge pro SDO = 256 Bytes

### 11.5 Literature

We recommend to check out the website [www.ethernet-powerlink.org](http://www.ethernet-powerlink.org) to help you quickly get to grips with the subject of Powerlink.

## 12 Error handling at UNIGATE® IC

A distinction can be made between two categories of system-errors:

Serious errors (1-4): 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.

The system-error can be read-out via the Script.

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

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

Table 1: Error handling at UNIGATE® IC

## 13 Firmware-update

### 13.1 Overview

UNIGATE® IC has a memory for the firmware. It can be replaced via the UNIGATE® IC's Powerlink-interface.

### 13.2 Execution of the firmware-update

The safest way for the firmware-update is the use of the basic board combined with the software "FDT.EXE" (firmware-download-tool). These tools are available from Deutschmann Automation (see chapter 15).

A Firmware-update can be carried out in the following operational modes:

Data exchange mode (via the IP-address UNIGATE®)

Konfiguration mode (via the IP-address UNIGATE®)

Firmware-update-mode (via the fixed IP-address 10.10.10.10)

### 13.3 Adjusting the firmware-update-mode

#### 13.3.1 Adjustment by hardware

UNIGATE® IC can be brought to the firmware-update-mode by the hardware. For it the signal  $\overline{BE}$  (boot enable) has to be pulled to the potential GND during the Power-up-process.

#### 13.3.2 Adjustment by software

If the UNIGATE® IC is in the configuration mode (see chapter 7.2) it can be brought to the firmware-update-mode interactively through the command CTRL-F (0x06). After sending the command a security query follows, that has to be answered with J or N (J = Yes, N = No). After a positive confirmation the IC is re-started in the firmware-update-mode.

### 13.4 Note on safety

The firmware-update should only be carried out when there is no other possibility left. A firmware-update-process that has already been started CANNOT be undone. With it the previously used firmware is permanently unusable.

### 13.5 Operation mode of the IC

#### Data exchange mode

This mode is required for the regular use of the IC. In this mode the IC will process all script-commands and normally exchange the corresponding user data. The bus as well is operated in this mode through the IC.

#### Configuration mode

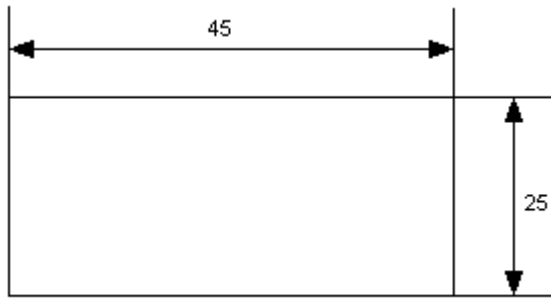
In the configuration mode the UNIGATE® IC will carry out a self-test after the start (or after a reset). After a successful self-test it will wait for further commands. Here it is possible to load a translated Script into the unit or to configure the device.

## 14 Technical data

In this chapter you will find all necessary technical data on UNIGATE® IC.  
All measurements in mm.

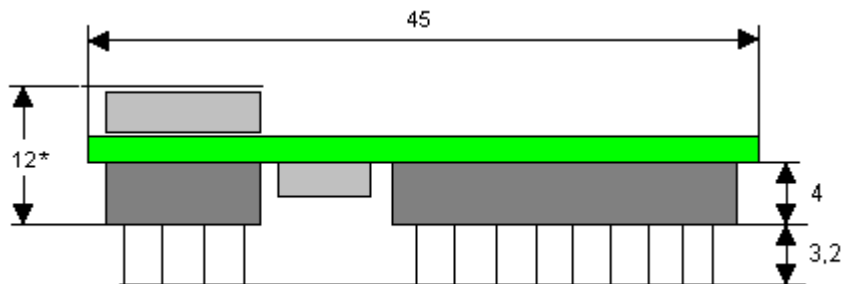
### 14.1 Mechanics of the UNIGATE® IC

#### 14.1.1 General dimensions of UNIGATE® IC



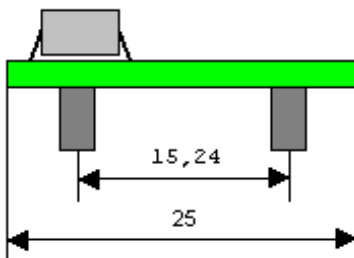
Max. height = 20 mm, including pin

#### 14.1.2 Dimensions UNIGATE® IC (Powerlink only)



\*) Version "without transformer" = 7

The pins of UNIGATE® IC - Powerlink are arranged with a grid spacing of 2.54 mm.



In case you intend to use other fieldbus ICs, the maximum overall height of  $\leq 20$  mm (including pins) has to be taken into consideration.

## 14.2 Technical data UNIGATE® IC - Powerlink

Characteristics	Explanation
Supply voltage	3.3V ± 5 %, max. 500 mA DC
Interface	2 UART interfaces, 1 synchronous serial interface
Physical separation Powerlink-side	Standard
IP-address	Set via WINGATE or via Script
Bus-baud rate	10 and 100 MBaud autodetect
UART-baud rate	Up to 625 Kbaud (adjustable via script)
Supported Ethernet protocols	UDP/IP, TCP/IP
Technology	32-bit-processor
Others	I/Os, analogous signals, shift registers, LEDs, switches and so on can be connected externally
Dimensions	45 x 12 x 25 mm (WxHxD) version "without transformer": 45 x 7 x 25 (WxHxD)
Installation	32 DIL
Weight	Approx. 9 g
Operating temperature	-40°C ..+85°C
Storage / transport temperature	-40°C..+100°C
Built-in position	Any

## 15 Accessory

The following tools are available from Deutschmann Automation.

### 15.1 Adapter RS232

In an application the adapter RS232 offers the possibility to replace an existing driver MAX 232 (only in DIL-16-housing) by this adapter. This board allows the use of the IC according to chapter 3.4. Please note that with it the Ethernet does not offer a connection conforming to the standards. With a plug adapter, however, at least the operation of Ethernet is possible.

The hardware is only meant for development purposes. It offers the possibility to make an existing application capable for bus connection in no time and to test the IC's utilizability and functionality.

### 15.2 Adapter RS485

From the functionality's point of view the RS485 adapter is the same as the RS232 adapter. It offers the possibility to replace a module LS 176 (only in DIL-8-housing) by the IC.

There are the same restrictions as for the RS232 adapter.

### 15.3 FirmwareDownloadTool (FDT)

The FirmwareDownloadTool is available for download from our homepage: it is required for an update of the firmware. Condition for it is, that a PC can be connected to the serial of the IC. The software describes the procedure of an update itself.

### 15.4 Protocol Developer

The Protocol Developer is the development environment for scripts, that also contain the Debugger. This software package also contains the documentation to all script-commands. This software is available for download from our homepage at <http://www.deutschmann.com>. The instruction manual for the Protocol Developer, which is available in pdf-format, gives further advise on how to use the software.

### 15.5 Developerkit UNIGATE® IC-AB IC

The Developerkit IC contains

- a Developerboard UNIGATE® IC (see chapter 15.5.1)
- a plug-in power pack to supply the Developerboard
- connection cables for appl. RS232, Debug RS232 and appl. RS422/485
- USB-cable
- Software and documentation to complete the packet.

#### 15.5.1 Developerboard UNIGATE® IC-AB

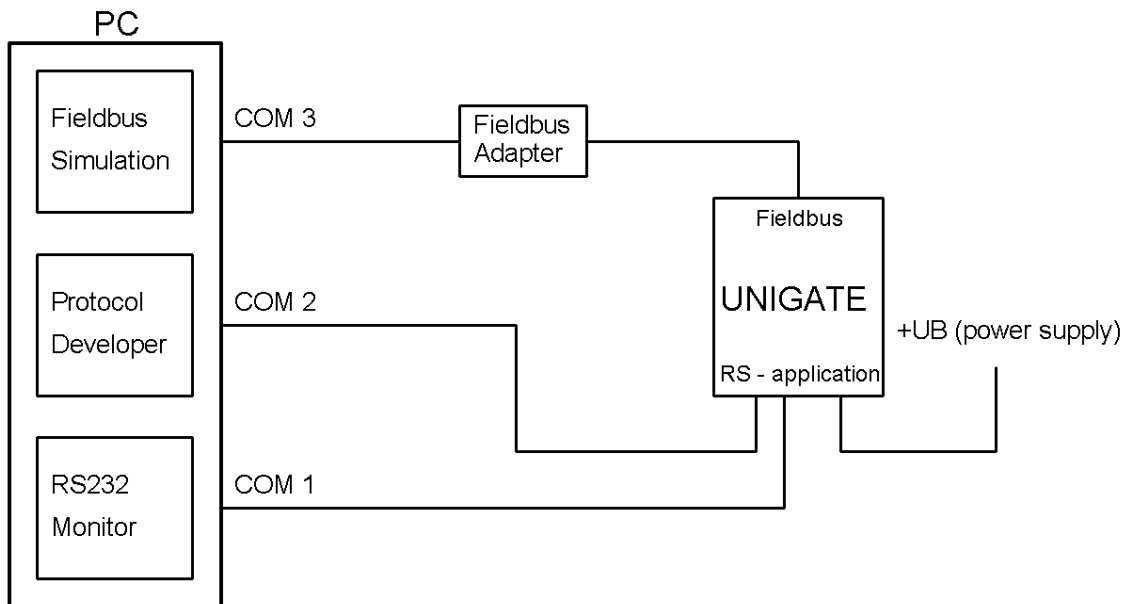
The Developer Board was developed so that the fast implementation of the Deutschmann All-in-one bus node UNIGATE® IC into your electronic system can be guaranteed. The board is suitable for all Fieldbuses and Industrial Ethernet Buses supported by Deutschmann Automation.



The required UNIGATE® IC / ICs are ordered separately. The required voltage (5V or 3.3V, depending on the version) can be adjusted. An RS232-interface or a USB-connection is available for the connection to the PC (Debug-interface).

The application can be connected either through the USB, RS232, RS485 or the RS422. The bus-connections according to standard or market standard are available to test the respective bus-side. Optionally Deutschmann Add-on packages (bus-master simulation) are available. The board contains 32 bit input and 16 bit output, provided with one LED each. Different connectors allow an easy coupling to your processor. A hole matrix field with the most important signals (voltage, IOs) allows a customized hardware extension (e. g. to connect a D/A converter).

**15.5.2 Quick start**



For a transparent data exchange you will find example scripts for the respective Fieldbus under "File->New" in the Protocol Developer.



## 16 Appendix

### 16.1 Explanations of the abbreviations

#### General

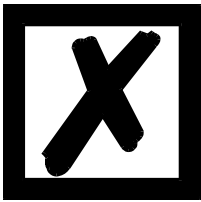
CL	=	Product group CL (Compact Line)
CM	=	Product group CM (CANopen Line)
CX	=	Product group CX
EL	=	Product group EL (Ethernet Line)
FC	=	Product group FC (Fast Connect)
GT	=	Galvanic separation RS-side
GY	=	Housing color gray
MB	=	Product group MB
RS	=	Product group RS
SC	=	Product group SC (Script)
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)
IO8	=	Option I/O8
16	=	Script memory expanded to 16KB
5V	=	Operating voltage 5V
3,3V	=	Operating voltage 3.3V

#### Fieldbus

ASI	=	AS-Interface (AS-i)
BI	=	BACnet/IP
BMS	=	BACnet MSTB
CO	=	CANopen
C4	=	CANopen V4
C4X	=	CANopen V4-version X (see comparison table UNIGATE <sup>®</sup> 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 <sup>®</sup> IC for the respective product)
IB	=	Interbus
IBL	=	Interbus
LN62	=	LONWorks62
LN512	=	LONWorks512
ModTCP	=	ModbusTCP
MPI	=	Siemens MPI <sup>®</sup>
PL	=	Powerlink

PN	=	PROFINET-IO
PBDP	=	PROFIBUS DP
PBDPL	=	PROFIBUS DP-version L (see comparison table UNIGATE® IC for the respective product)
PBDPX	=	PROFIBUS DP-version X (see comparison table UNIGATE® IC for the respective product)
PBDPV0	=	PROFIBUS DPV0
PBDPV1	=	PROFIBUS DPV1
RS	=	Serial RS232/485/422

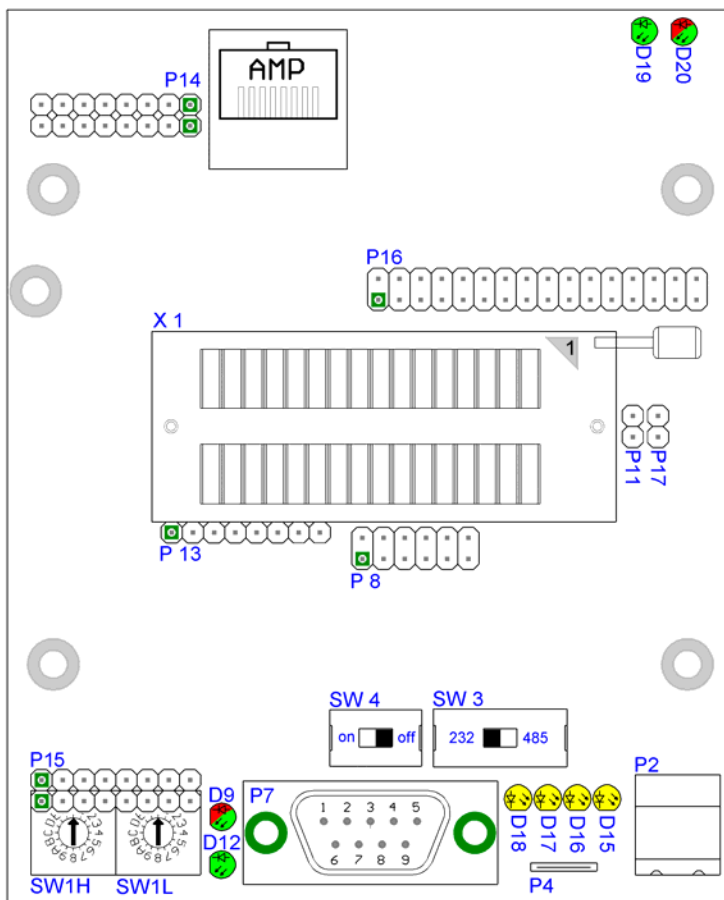
## 16.2 Basis board



The basis board that is described in this chapter was supplied until the end of 2008.

A new board is available since the beginning of 2009 (see chapter 15.5.1).

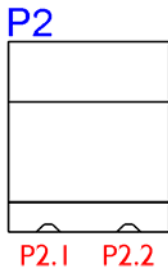
### 16.2.1 Overview basis board for PN, EI + PL 3.3V



**Slot X 1 (ZIF-socket)**

PIN 1 of the IC is located up at the lever of the ZIF-socket.

Never place the IC into the socket back to front!

**P 2**

Pin	Signal
Pin 1	24 V DC
Pin 2	Ground

The basis board is supplied with voltage through this plug connector.

**P 3 (AMP)**

Powerlink plug connector, for the assignment of the connector see chapter 16.2.2.2.

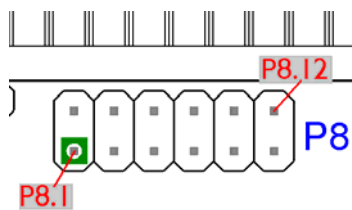
**P 4**

Earth terminal 6.3 mm for basis board.

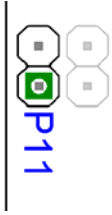
**P 7**

This plug is the basis board's serial connection to the customer's device and the connection to the PC (Debug-interface).

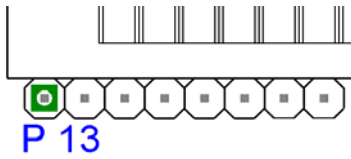
For the pin assignment see chapter 16.2.2.1.

**P 8**

The illustration shows the arrangement of the pins. On this connector strip the signals of the serial connection between IC and RS-drivers are wired. For an initial development you will probably also use an existing driver in your application. In order to exchange it later on, you can also directly take the signals of the serial interface here.

**P 11**

Force Boot. By setting this bridge the Pin BE is dragged to Ground. For the function see chapter 13.3.1.

**P 13**

Status signal of the IC

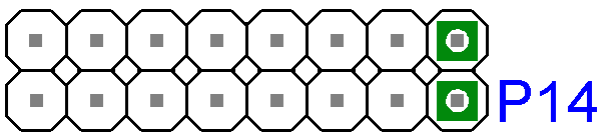
Plug connector P 13

Pin	Signal
1	Vcc
2	Gnd
3	-RESET
4	RX of the IC (TTL-level)
5	TX of the IC (TTL-level)
6	TE Pin IC (TTL-level)
7	TX Debug of the IC (TTL-level)
8	RX Debug of the IC (TTL-level)

**P 14**

Input shift register

For a detailed assignment and for information on which pin is assigned to which bits, see also chapter 5.

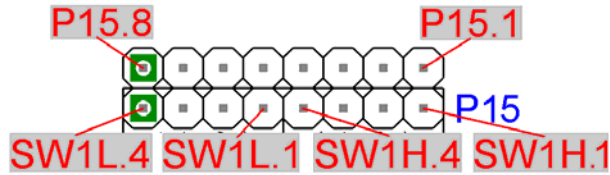


Connection	Pin	Meaning
P 14	1	Input 9
	.....	.....
	8	Input 16

**P 15, SW1H, SW1L**

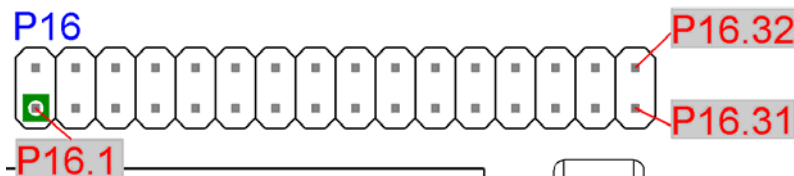
Input shift register

Basically the same applies as for P 14, with the exception that different input bits of the shift registers are wire.



Connection	Pin	Meaning
P 12	1	Input 1
	.....	.....
	8	Input 8
SW1H	1	Input 17
	.....	.....
	4	Input 20
SW1L	1	Input 21
	.....	.....
	4	Input 24

**P 16**



All digital outputs of the shift registers are available here. Additionally the LEDs D9, D15..D18, D20 are connected to the shift registers.

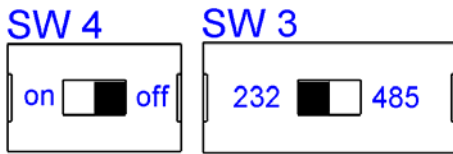
**P 17**



With P17 the UNIGATE® IC can be brought into the configmode. If the jumper is plugged and if the UNIGATE® IC is restarted (by power off and power on or by reset), then the UNIGATE® IC will start in the configmode. In order to use the configmode with Deutschmann software tools the interface of the board has to be in RS232-position and the PC has to be connected with the „normal“ interface, where otherwise your application is connected to. See also chapter 7.2.

**SW1H, SW1L, SW5H, SW5L**

The rotary switches SW1H, SW1L, SW5H, SW5L are plugged into the base boards and can be removed if required. As a default the rotary switches are plugged in and can be read in through the basis board’s shift registers base boards (see also chapter 5 for it).

**SW3, SW4**

These switches are required for the setting of the serial interface. The switch SW3 is used to switch between interface RS232 and RS485. This is the interface, the customer's device is connected to. The Debug-interface always has RS232-level.

The switch SW4 is of importance only, when it is an RS485-interface. Then this switch can be used to connect the termination of the RS485-bus.

Each switch position can be taken from the illustration.

**D12**

Power LED

This LED is always supposed to be shining statically green when the board is supplied with voltage.

**D9, D15..D18, D20**

LEDs that are connected to the shift register components.

**D19**

LED LINKACT

This LED is directly controlled by the processor and shines when the gateway is in an operable network (link-pulses are received) and it flickers when network data traffic takes place.

**16.2.2 Connectors of the basis board****16.2.2.1 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 P7 (9-pin Sub-D, plug)

Pin No.	Name	Function
1		
2	Rx/RS485- / RS422- (Tx)	Receive signal customer's device
3	Tx/RS485+ / RS422+ (Tx)	Transmit signal customer's device
4	Tx / Diag	Transmit signal Debug interface
5	GND RS	Ground connection, reference for PIN 2+3+6+7
6	RS422- (Rx)	
7	RS422+ (Rx)	
8	Not connected	Not connected
9	Rx / Diag	Receive signal Debug interface

**Attention:**

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

Pin assignment P2 (2-pin screw-type/plug connector)

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

**16.2.2.2 Powerlink connector**

The connector for connection to Powerlink (RJ45) is located on the upper side of the device.

Pin No.	Name	Function
1	TD+	Transmission line+
2	TD-	Transmission line -
3	RD+	Receive line+
4	n. c.	Not connected
5	n. c.	Not connected
6	RD-	Receive line-
7	n. c.	Not connected
8	n. c.	Not connected

**16.2.2.3 Power supply of the basis board**

The device must be powered with 10.8-30 VDC.

The power is supplied through the separate 2-pin screw-type/plug connector.

Please pay attention to the fact that devices of the series UNIGATE<sup>®</sup> cannot be operated with AC voltage.

**16.2.2.4 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 Powerlink cable shield is not electrically connected to the shield signal of the electronic circuitry for reasons relating to interference immunity.

**16.2.2.5 Rotary coding switches**

The rotary coding switches are socketed and can be taken off, in order to alternatively use the pins of the shift register.

**16.2.2.6 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.

**16.2.2.7 Slide switch (RS485 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.



## 17 Servicing

Should questions arise that are not covered in this manual you can find further information in our

- FAQ/Wiki area on our homepage [www.deutschmann.com](http://www.deutschmann.com) or directly in our Wiki on [www.wiki.deutschmann.de](http://www.wiki.deutschmann.de)

If your questions are still unanswered 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 (CET).

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

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

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

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

### 17.1 Downloading PC software, XDD-file, Script examples etc.

You can download the current version of WINGATE<sup>®</sup>, a topical sample-XDD-file and Script examples free of charge from our Internet server.

Here you will also find the software tool Protocol Developer for UNIGATE<sup>®</sup> SC and IC.

<http://www.deutschmann.com>

