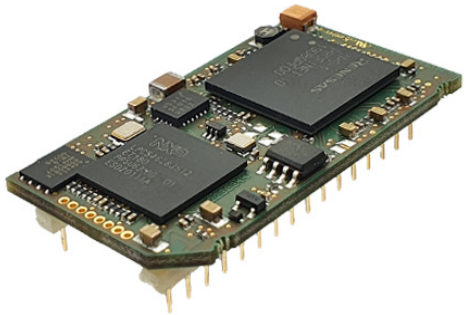
Deutschmann

your ticket to all buses

Instruction Manual
Universal Fieldbus-Gateway
UNIGATE® IC2 - PROFINET 2Port



Deutschmann Automation GmbH & Co. KG
www.deutschmann.com | wiki.deutschmann.de

Manual Art.-No.: V4475E

1 General introduction 9

2 The UNIGATE® IC2 10

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® IC2 hardware survey 11

3 Hardware design 12

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 (SPI: SCK) 13
3.2.7 -Reset In 13
3.2.8 Error LED 14
3.2.9 LED-LINK1 14
3.2.10 LED-LINK2 14
3.2.11 State-LED 14
3.2.12 -Config Mode 14
3.2.13 DbgTX, DbgRX 14
3.2.14 TE 14
3.2.15 TX, RX 15
3.3 Software 16
3.4 Basic line of proceeding 16
3.5 Connection examples 17
3.6 Layout examples 19
3.7 Handling (mounting the UNIGATE® IC2 on the carrier board) 23
3.8 Requirement for Ethernet-Interface 24

4 The serial interface 25

4.1 Overview 25
4.2 Initialization of the serial interface 25
4.3 Use of the serial interface 25
4.4 Further operation modes 25

5 Synchronous serial interface 26

5.1 Shift register operation 26
5.2 SPI mode 26

6 The Debug-interface 27

6.1 Overview of the Debug-interface 27
6.2 Starting in the Debug-mode 27
6.3 Communication parameter for the Debug-interface 27
6.4 Possibilities with the Debug-interface 27

6.5 Commands of the Debug-interface . . . 27
7 Script and configuration . . . 28
7.1 Overview . . . 28
7.2 The configuration mode . . . 28
7.3 Update the script . . . 28
7.4 Configuration of the UNIGATE® IC2 . . . 30
7.4.1 Ethernet . . . 30
7.4.2 RS232/RS485/RS422 . . . 31
8 Generating a script . . . 32
8.1 What is a script? . . . 32
8.2 Memory efficiency of the programs . . . 32
8.3 What can you do with a script device? . . . 32
8.4 Independence of buses . . . 32
8.5 Further settings at the IC2 Gateway . . . 32
8.6 The use of the Protocol Developer . . . 33
8.7 Accuracies of the baud rates at UNIGATE® IC2 . . . 33
8.8 Script processing times . . . 33
9 FTP-Server . . . 35
9.1 Users, Passwords & Groups . . . 35
9.2 File – „users“ . . . 35
9.2.1 Format . . . 35
9.2.2 Permissions . . . 35
9.2.3 Load . . . 36
9.3 Script update via FTP . . . 36
9.4 Updating the system configuration via FTP . . . 36
9.5 Use FTP server . . . 36
10 WEB server . . . 37
10.1 Server-Side-Includes (SSI) . . . 37
10.2 HTML forms (CGI) . . . 37
10.3 Example . . . 37
11 File system . . . 38
11.1 Directory /etc . . . 38
11.2 Directory /srv . . . 38
12 PROFINET . . . 39
12.1 Data exchange mode . . . 39
12.2 PROFINET address assignment . . . 39
12.3 PROFINET device name . . . 39
12.4 Device description file . . . 39
12.5 MRP (Media Redundancy Protocol) . . . 40
12.6 System redundancy S2 . . . 40
12.7 Possible data lengths . . . 40
12.8 Startup phase . . . 40
12.9 Literature . . . 40

13 Error handling at UNIGATE® IC2 41
14 Firmware-update 42
14.1 Overview 42
14.2 Execution of the firmware-update 42
14.3 Adjusting the firmware-update-mode 42
14.3.1 Adjustment by hardware 42
14.4 Note on safety 42
14.5 Operation mode of the IC2 42
15 Technical data 43
15.1 Mechanics of the UNIGATE® IC2 43
15.1.1 General dimensions of UNIGATE® IC2 43
15.2 Technical data UNIGATE® IC2 - PROFINET 2Port 45
16 Accessory 46
16.1 FirmwareDownloadTool (FDT) 46
16.2 Protocol Developer 46
16.3 Developer Kit UNIGATE® IC-AB IC 46
16.3.1 Developer Board UNIGATE® IC2-AB 46
16.3.2 Quick start 47
17 Appendix 48
17.1 Explanations of the abbreviations 48
18 Servicing 50
18.1 Returning a device 50
18.2 Downloading PC software 50

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 – 2025. 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® IC2 by Deutschmann Automation any more. All digital functions, software, stack and driver as well as optocoupler are integrated on a UNIGATE® IC2 in correspondence with the standard. In addition to the reduction of the required size, also different fieldbusses can easily be integrated.

Through the flexible firmware of UNIGATE® IC2 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® IC2 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.

The UNIGATE® IC2 is a further development of the 1st series of the UNIGATE® IC. Equipped with a Cortex-M4 controller, the UNIGATE® IC2 achieves much higher transfer rates when communicating via SPI or UART. In addition, the scripting processing time (depending on the commands used) is reduced by a factor of about 50 - 80. Thus, script execution times in the microsecond range are possible.
The UNIGATE® IC2 is also PIN-compatible with the 1st series of the UNIGATE® IC.

2 The UNIGATE® IC2

2.1 Technical introduction

The UNIGATE® IC2 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® IC2 and the required plug connectors.

2.2 Availability

The module is available as PROFINET. 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® IC2s under:
http://www.deutschmann.com.

2.3 Firmware

UNIGATE® IC2 is programmed via scripts. On principle any script, that has been developed for a UNIGATE® SC, can also be operated on the UNIGATE® IC2.

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® IC2. For more information on this serial interface see chapter 4.

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 IC2 is also operable autonomously without that controller.

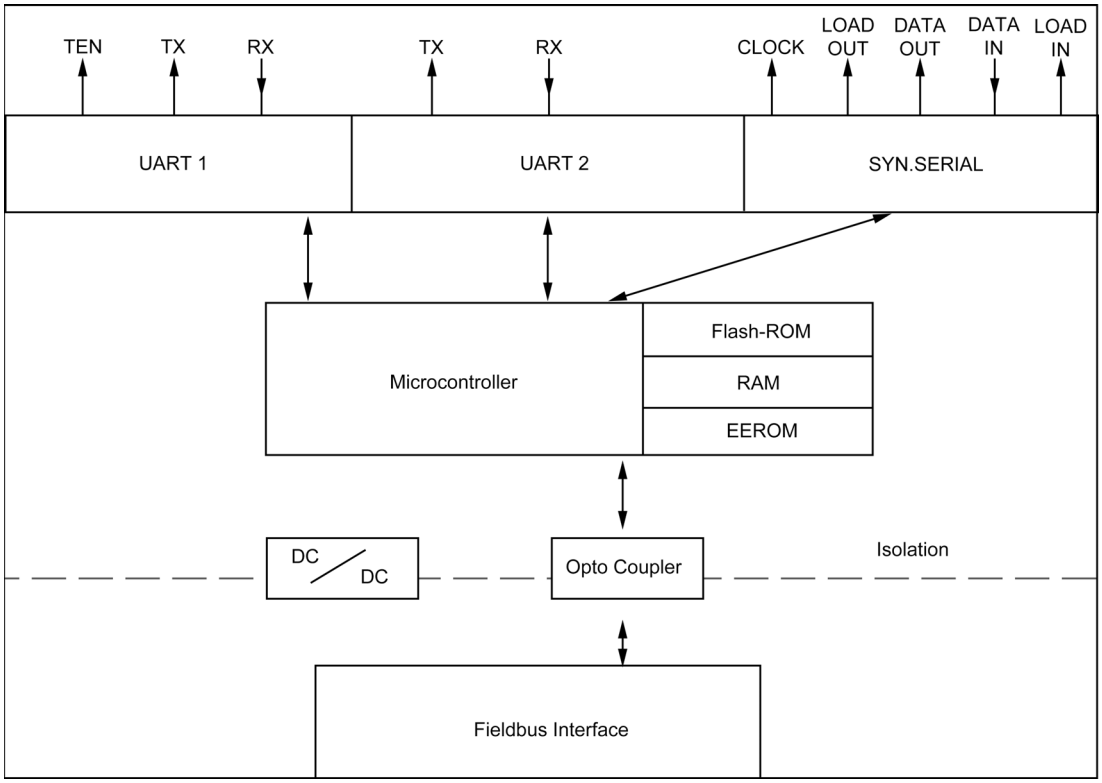
2.6 The Debug-interface

The UNIGATE® IC2 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® IC2 hardware survey

The hardware of the UNIGATE® IC2 consists of some few standard components. The picture below shows the functional structure of the IC2.



3 Hardware design

This chapter gives basic advise, that is required in order to load UNIGATE® IC2 into your own hardware designs. In the following all ports of UNIGATE® IC2 are described in detail.

3.1 Ports

UNIGATE® IC2 features 32 pins in its layout as a DIL 32 component.The exact mechanical dimensions can be taken from chapter 15.

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% < 460mA	Vcc	+3.3V voltage supply	
2	IN_{Logic}	-BE	boot enable	
3	OUT_{Driver}	Load out (SS0-)	strobe signal for synchronous, serial interface	
4	OUT_{Driver}	Data out (SS1-)	output data for synchronous, serial interface	
5	IN_{Logic}	Data in (MISO)	input data of the synchronous, serial interface	internal pull up
6	OUT_{Logic}	Load in (MOSI)	strobe signal input data of the synchronous, serial interface	
7	OUT_{Driver}	Clock (SCK)	clock pulse signal for synchronous, serial interface	
8	IN_{Reset}	-Reset in	reset-input of the IC2	internal pull up
9*	connected to pin 1	Vcc	+3.3V voltage supply	
10	according to norm	Error-LED	Anode of the red LED	
11	according to norm	LED-LINK1	Ethernet-link pulse found / Networkdata traffic Port 1	
12	nc	nc	not connected	
13	according to norm	RD1-	Ethernet-signal RD- Port1	
14	according to norm	RD1+	Ethernet-signal RD+ Port1	
15	according to norm	TD1-	Ethernet-signal TD- Port1	
16	according to norm	TD1+	Ethernet-signal TD+ Port1	
17	according to norm	TD2+	Ethernet-signal TD+ Port2	
18	according to norm	TD2-	Ethernet-signal TD- Port2	
19	according to norm	RD2+	Ethernet-signal RD+ Port2	
20	according to norm	RD2-	Ethernet-signal RD- Port2	
21	nc	nc	not connected	
22	according to norm	LED-LINK2	Ethernet-link pulse found / Network data traffic Port2	
23	nc	nc	not connected	
24*	connected to pin 32	GND	Ground supply voltage of the IC2	
25		State-LED	Bus status LED (Anode of the green LED)	
26	IN_{Logic}	-Config Mode	Signal to start in configuration mode	internal pull up
27	OUT_{Logic}	DbgTX	serial Debug TX	
28	IN_{Logic}	DbgRX	serial Debug RX	internal pull up
29	IN_{Logic}	RX	serial data RX	internal pull up
30	OUT_{Logic}	TX	serial data TX	
31	OUT_{Logic}	TE	Transmit enable	
32	GND	GND	Ground supply voltage of the IC2	

* The supply voltage is 3.3V +/- 5 %, max. 460 mA DC.

3,3V±5% < 460mA	V_{IL}	V_{IH}
IN_{Reset}	< 0,7V	>2,4V
IN_{Logic}	< 0,8	>2,0V
	V_{OL}	V_{OH}
OUT_{Driver}	< 0,4V / 6mA	>2,7V / 6mA

3.2.1 -Boot enable

The IC2 is started in the firmware update mode with the level GND during the power up process. See also chapter 12.

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 (SPI: SCK)

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® IC2, 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 Error LED

A red LED can be connected to this line (see chapter 3.6). It can have the following conditions:

Off	No error
On	Error

The Error LED can be activated by a system error (see chapter 13) or by commands in the script (see online help of the Protocol Developer).

3.2.9 LED-LINK1

A green LED can be connected to this line (see chapter 3.6). This is directly accessed by the PROFINET processor and is glowing when the Gateway on Port 1 is an operational network (it will receive link pulses).

Off	No Ethernet connection
On	Working network (Link)
Flash 1 Hz	PROFINET participant flash test

3.2.10 LED-LINK2

A green LED can be connected to this line (see chapter 3.6). This is directly accessed by the PROFINET processor and is glowing when the Gateway on Port 1 is an operational network (it will receive link pulses).

Off	No Ethernet connection
On	Working network (Link)
Flash 1 Hz	PROFINET participant flash test

3.2.11 State-LED

A green LED can be connected to this line (see chapter 3.6). It shows the bus status and can have the following conditions:

Off	PROFINET Device not started (Bus-start not executed)
Flash 10Hz	System-Error or warning (see chapter 13)
Flash 1 Hz	PROFINET Device started, no PROFINET connection yet. (waiting for connection)
On	PROFINET connection established

3.2.12 -Config Mode

If the pin has the level GND, then the IC2 starts in the configuration mode.

3.2.13 DbgTX, DbgRX

They are transmission line and receive line as well of the IC2's Debug-interface. For the function description of the Debug-interface see chapter 6.

3.2.14 TE

The transmit enable signal allows the connection of RS485 drivers to the IC2's serial interface. The signal is set to High whenever the IC2 sends via the line TX.

3.2.15 TX, RX

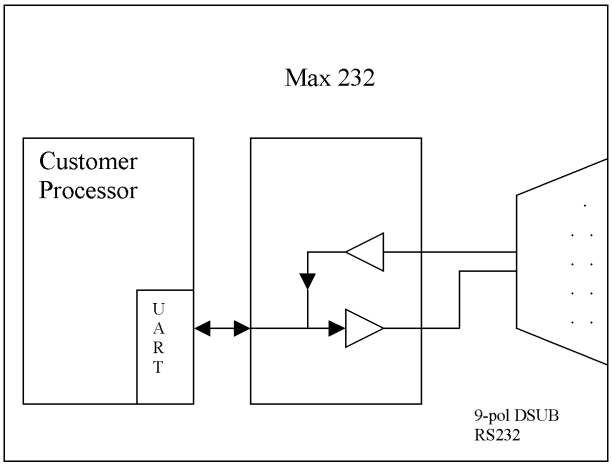
Transmission and receive line of the serial interface. This interface is programmable in accordance with the description in chapter 4.

3.3 Software

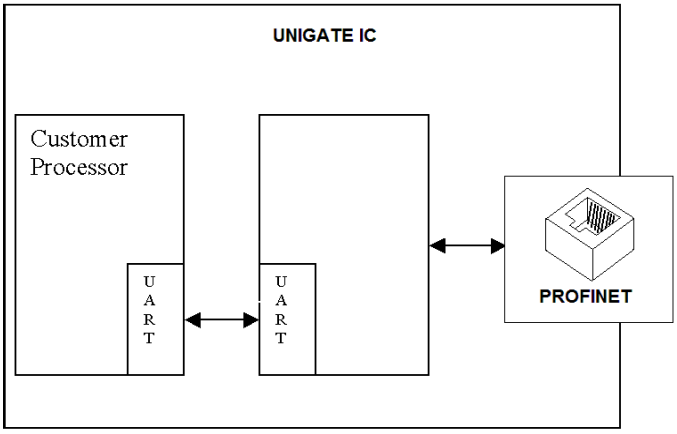
The software executes script-commands, which in turn control the IC2's hardware and they process their complete protocol by software. The script itself can be generated by the company Deutschmann Automation or with the software Protocol Developer by yourself. For a detailed description of the script.commands 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® IC2.



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

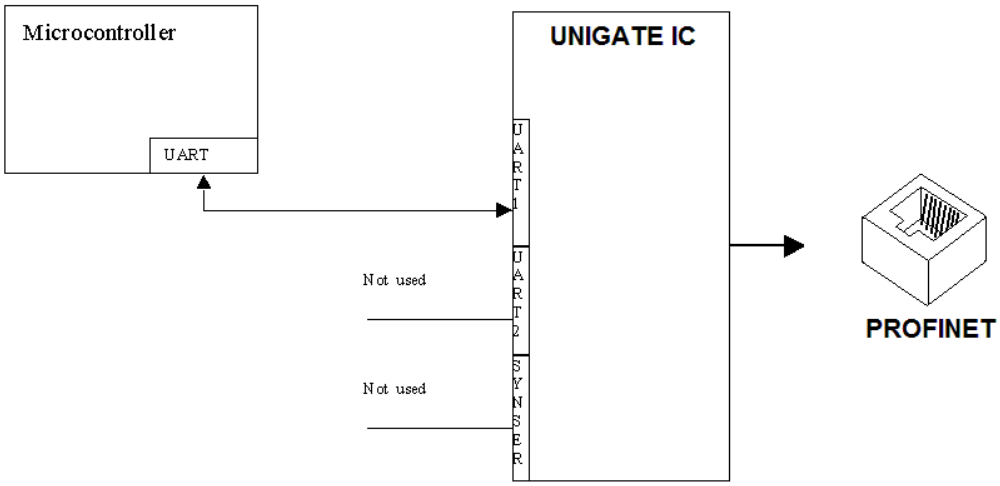


After the RS232-driver has been replaced by the UNIGATE® IC2, PROFINET is available at an RJ45-socket.

3.5 Connection examples

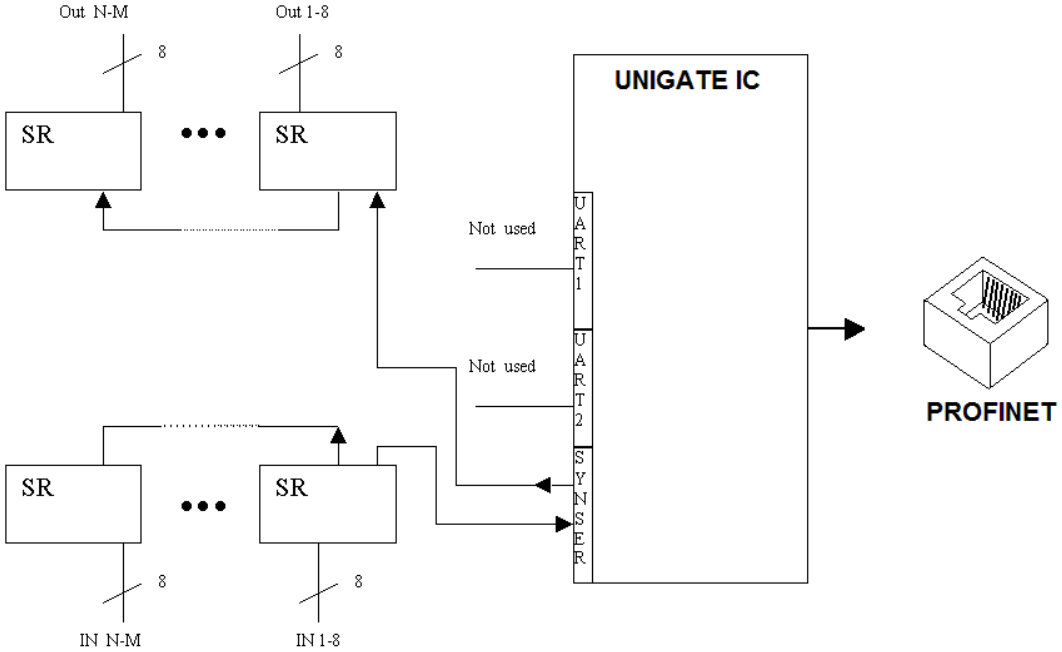
Here you will find some advise that offers help for a re-design. In the following several versions are listed, that should make it easier for you to decide.

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



The UNIGATE® IC2 independently processes the communication with the customer's device via the TTL-interface.

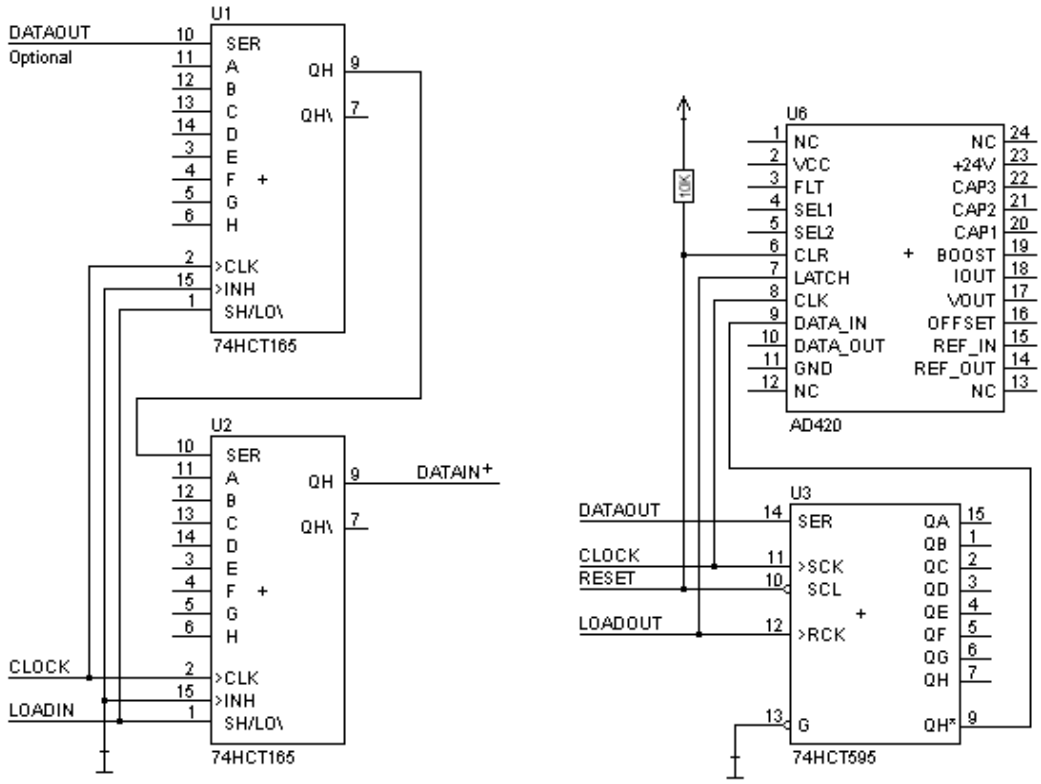
Version 2: Use of UNIGATE® IC2 for digital or analog I/O-modules



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 IC2.

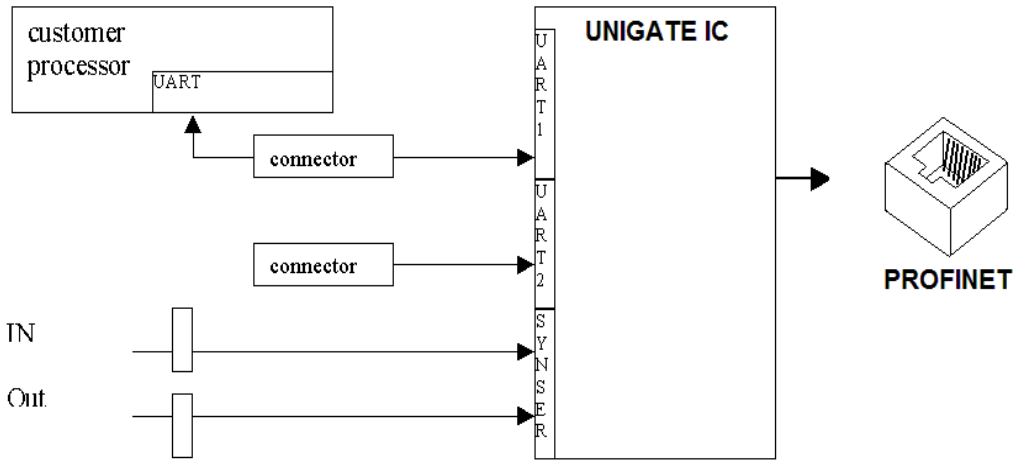


Version 3: Example for digital I/Os

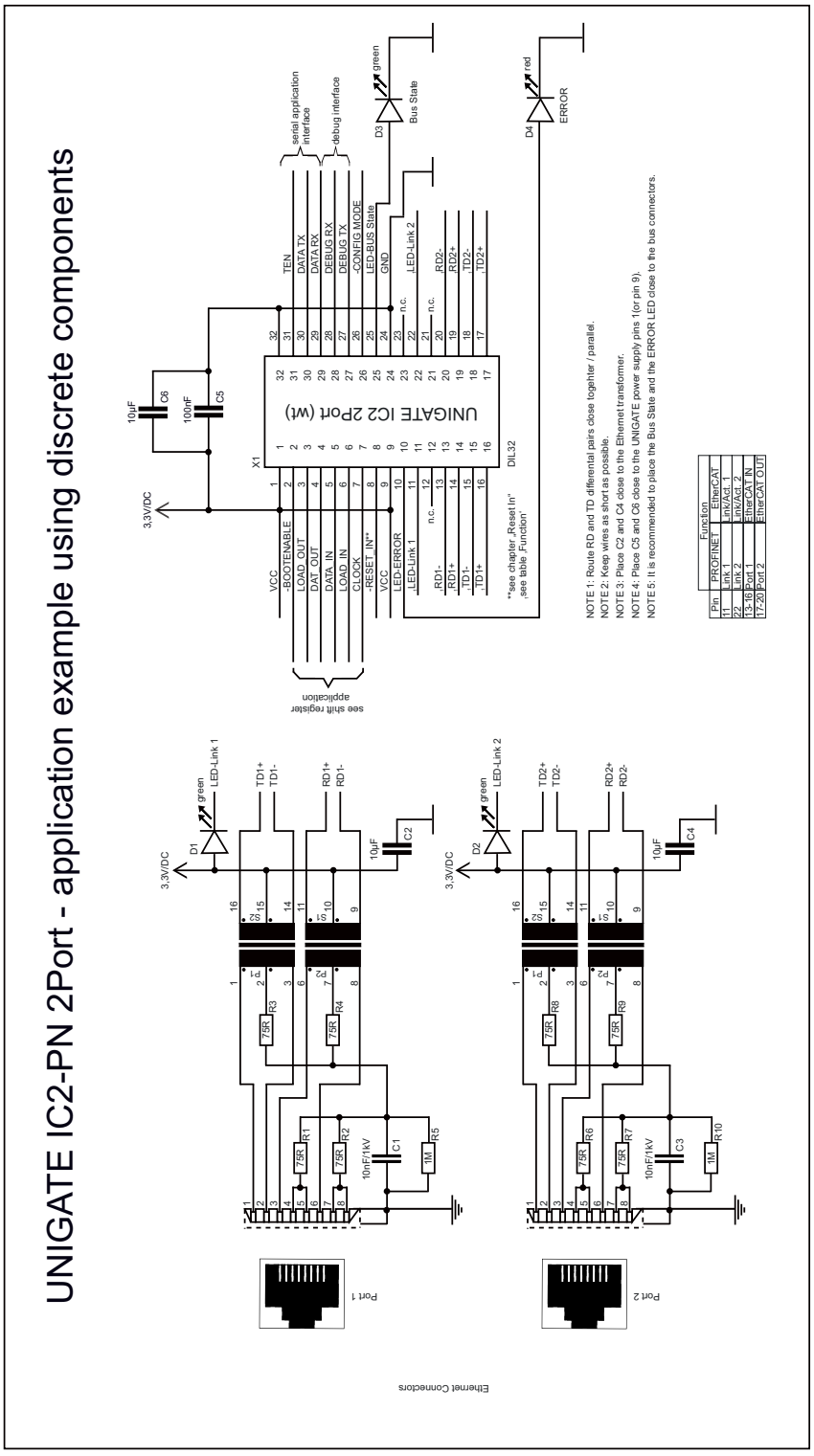
The serial synchronous and the asynchronous interface as well can be operated by UNIGATE® IC2 at the same time. Here the possibility results that an existing application can be extended by additional digital or analog I/Os.

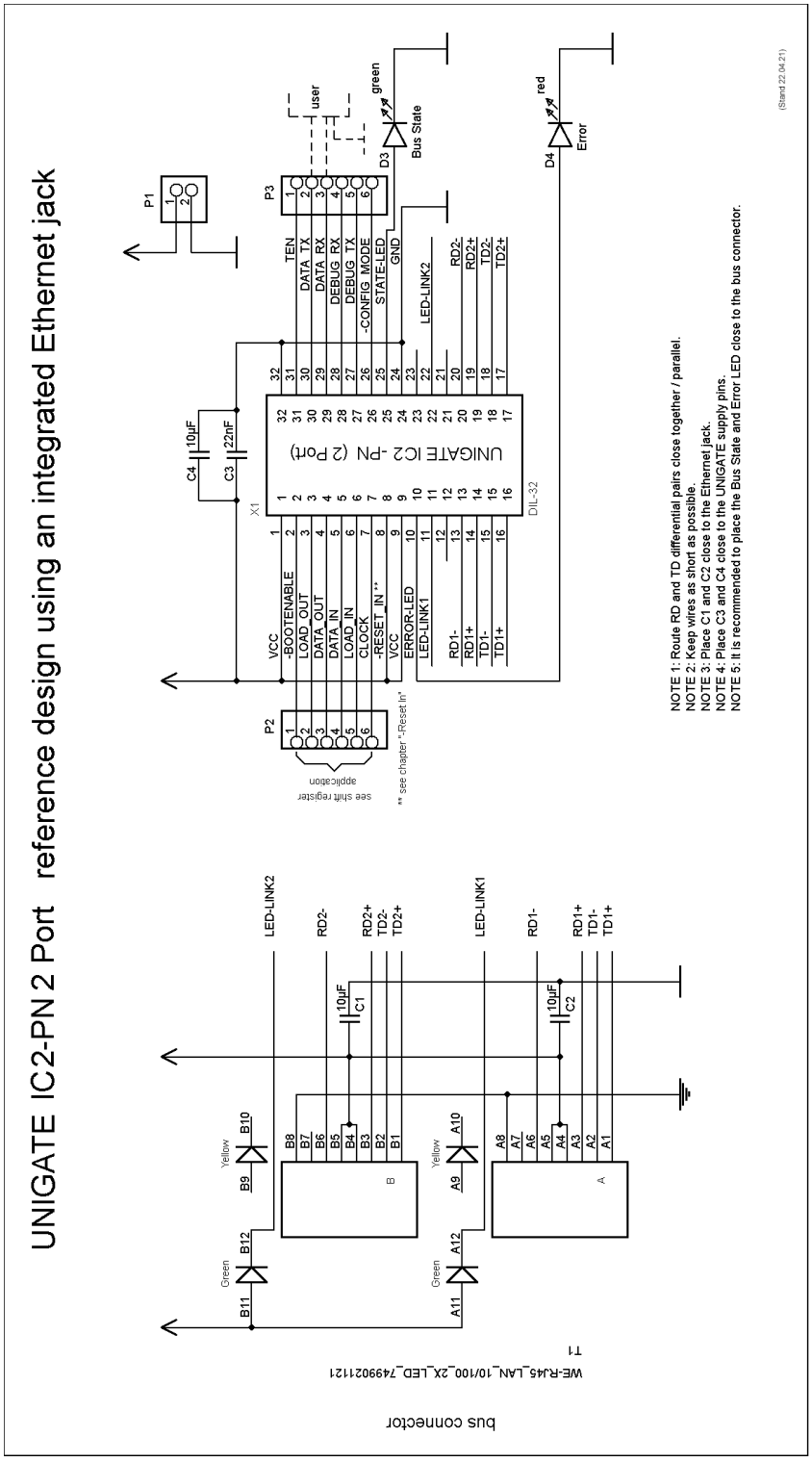
In chapter 5.2 you find an example for a script, that operates these I/Os.

Valid for all versions: A designated 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.

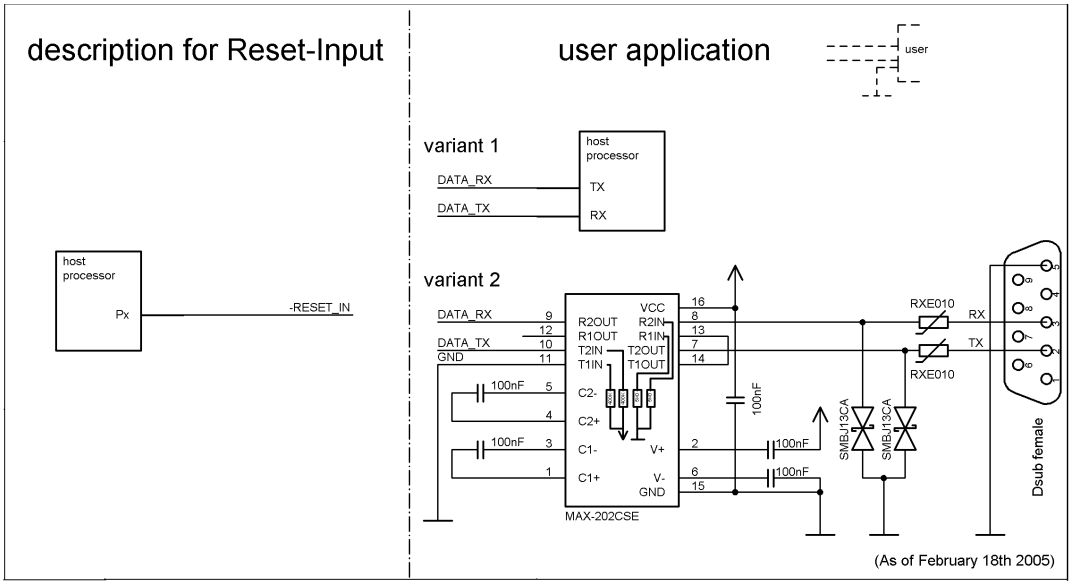


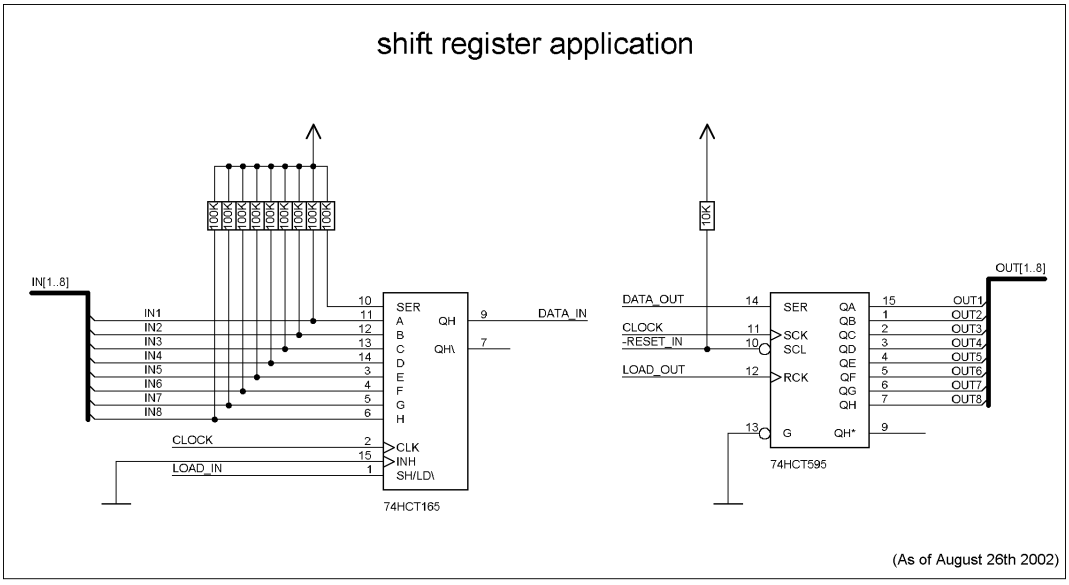
3.6 Layout examples

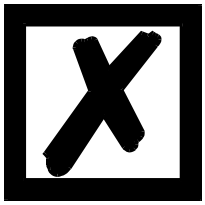



UNIGATE IC2-PN 2 Port reference design using an integrated Ethernet jack
UNIGATE IC2 -PN (2 Port)
DIL-32
X1
P1
P2
P3
C1 10µF
C2 10µF
C3 22nF
C4 10µF
D3
Bus State
green
D4
Error
red
user
VCC
-BOOTENABLE
LOAD_OUT
DATA_OUT
DATA_IN
LOAD_IN
CLOCK
-RESET_IN **
ERROR-LED
LED-LINK1
RD1-
RD1+
TD1-
TD1+
TEN
DATA_TX
DATA_RX
DEBUG_RX
DEBUG_TX
-CONFIG_MODE
STATE-LED
GND
LED-LINK2
RD2-
RD2+
TD2-
TD2+
** see chapter "Reset in"
see shift register application
NOTE 1: Route RD and TD differential pairs close together / parallel.
NOTE 2: Keep wires as short as possible.
NOTE 3: Place C1 and C2 close to the Ethernet jack.
NOTE 4: Place C3 and C4 close to the UNIGATE supply pins.
NOTE 5: It is recommended to place the Bus State and Error LED close to the bus connector.
(Stand 22.04.21)
Yellow
Green
bus connector
WE-RJ45_LAN_10/100_2x_LED_749902112 1
T1

When connecting the UG-IC2 and the RJ45-connector it is important to make sure the signal lines RD1-/RD1+ resp. TD1-/TD1+ also RD2-/RD2+ resp. TD2-/TD2+ run as short a possible, parallel and equal in length. Because this is for analog signals, it is also advisable not to place digital lines in close proximity.







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.

Note:
The following recommendations result from the reference design of the Micrel Switch used in the UNIGATE® IC2-PN 2Port:

Compatible isolation transformers:

- Pulse H1102
- Transpower HB726
- Bel Fuse S558-5999-U7
- YCL PT163020
- DELTA LF8505

3.7 Handling (mounting the UNIGATE® IC2 on the carrier board)

Depending on the application and the expected shock- and vibration-conditions you can generally choose from the following possibilities for the UNIGATE® IC2's installation on the carrier board:

- Mounting on a socket in the carrier board. If necessary solder the UNIGATE® IC2 to 2 or 4 pins in the socket. Normally the IC2 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® IC2 was plugged in the socket pull an isolated wire over the IC2 and solder it on the carrier board at the specified holes.
- Fasten the UNIGATE® IC2 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® IC2 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

Since there is an extremly large number of suppliers for RJ45 connectors (with and without integrated Ethernet transformer), or single Ethernet transformers on the market, 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 ± 5%
Open Circuit Inductance 350 uH (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 connector with transformer
We mainly use: Order-No. 615 008 145 521

RJ45 connector (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

Further 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® IC2 and the microcontroller 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® IC2's serial interface constitutes the basis for it. The UNIGATE® IC2 allows to connect of controllers with a baud rate of 110 Baud – 6 MBaud. The baud rate 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® IC2, the module carries out actions independently, in order to identify data from the connected device. For customers who already have a software-adaptation at he company Deutschmann Automation, this protocol as well or a Script after an adaptation can be processed by the IC2.

In any case the IC2 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® IC2 is used to connect clocked shift registers or components that have a Serial Peripheral Interface (SPI).

It allows
- the expansion of the IC2 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 given 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`“.

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.

5.2 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 33 Mbit/s as master, 10 Mbit/s as slave. For details please see the IC2-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.

6 The Debug-interface

6.1 Overview of the Debug-interface

The UNIGATE® IC2 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® IC2 (power up) the firmware will output the binary character 0 (0x00) after a self-test was carried out on this interface. If the IC2 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® IC2, as well as the configuration, can be replaced or updated via the serial interface (application) in the configuration mode.

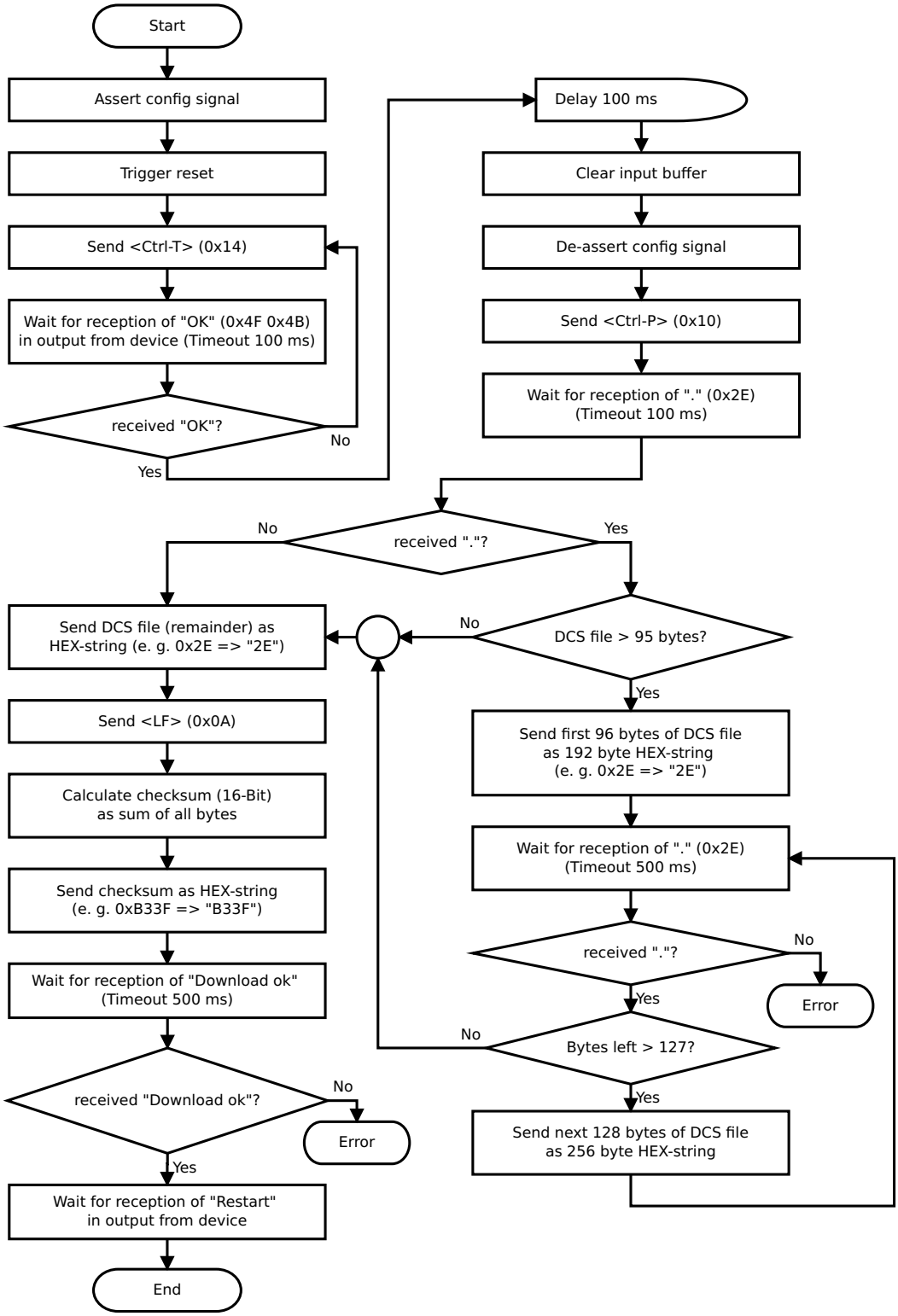
7.2 The configuration mode

If the pin „ConfigMode“ pulled to GND during the PowerUp or Reset, then the UNIGATE® IC2 starts in the configuration mode. In this mode it is possible to communicate with the IC2 without processing the regular software. In this mode it is possible to change the UNIGATE® IC2’s settings of the standard software or to write a new script in the UNIGATE® IC2. 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:
IC2-PN2P V0.1.0[44] (c)dA[220MHz] Script(C:131/16192,V:0/16384)="Leer" Author="Deutschmann Automation GmbH" Version="1" Date=21.08.2001 SN=47110010 MAC=00-14-11-8C-E5-9F IP=0.0.0.0 Name=""

7.3 Update the script

- The preferred option is to insert the UNIGATE® IC2 into the base board from Deutschmann (Developer Board UNIGATE® IC2-AB) and use the Deutschmann tools (software WINGATE with „Write Script“ under „File“ or with the software ScriptProgramTool).
- You can also automatically replace the script in your application from your host (see flowchart).

Script Download



The operational sequence is as follows:

The Gateway has to be in the config-mode.
The script-download is initiated with "Ctrl-P (=0x10)".
After that the data follows byte by byte as ASCII-hex-characters.
The download is terminated with a "LF (=0x0A)".
Afterwards the word-checksum follows as ASCII-hex-characters.

The Gateway responds with a clear text reply to that download and carries out a warm start.

Example:
The following 4-bytes script is supposed to be downloaded: 0x01 0x12 0x5A 0x23
The sum of the bytes is 0x0090 as checksum.
Then the following sequence is to be sent:

1. 0x10 Ctrl-P
2. 0x30 '0'
3. 0x31 '1'
4. 0x31 '1'
5. 0x32 '2'
6. 0x35 '5'
7. 0x41 'A'
8. 0x32 '2'
9. 0x33 '3'
10. 0x0A LF
11. 0x30 '0'
12. 0x30 '0'
13. 0x39 '9'
14. 0x30 '0'

Gateway's reply: "Download ok"

7.4 Configuration of the UNIGATE® IC2

UNIGATE® IC2 is delivered with an empty script.
The configuration of the UNIGATE® IC2 - PROFINET is restricted to the setting of the Ethernet-address. (see also chapter 12.2)

7.4.1 Ethernet

- Baud rate: 100 MBit/s
- IP-address (Gateway)
- Ethernet (MAC-address)
- IP-destination host

7.4.2 RS232/RS485/RS422

- RS-ype: 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 IC2 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 software tool Protocol Developer can be downloaded from our website http://www.deutschmann.de.
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.

8.7 Accuracies of the baud rates at UNIGATE® IC2

The baud rate of the serial interface is derived from the processor's crystal frequency. The deviation of the real baud rate from the set baud rate is always less than 1.1%.

8.8 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 Sript 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

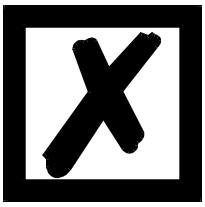
From experience approximately 10 µs can be calculated for each Script line. This value confirmed itself again and again in many projects as a standard value. He is always quite accurate if the processor has enough time available for the Script processing.
By means of the tasks mentioned above, the following 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 transfered 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.

9 FTP-Server

This UNIGATE® has an integrated FTP server that enables access to the file system of the device. The FTP server is password protected.
In addition to the predefined user name "admin" and the password "nimda", others can also be stored.



If the stored user name and/or password is lost, access can be made via the predefined user name "admin" and the password "nimda" if the UNIGATE® is started in configuration mode.
However, the user names and passwords stored in the "users" file cannot be readout. The file can only be deleted or replaced.

By default, the FTP server is disabled in Run mode. This can be activated by script via the FTPServer parameter.

Note: Access in run mode is also only possible with your own user name and password. The predefined user name "admin" and the corresponding password "nimda" cannot be used for this.

9.1 Users, Passwords & Groups

The ‘users’ file is used to manage users, passwords and groups. This file must be added to the file system under /etc. The ‘users’ file is not present in the delivery state.

Note: For firmware versions < 1.6.0, this file must be added to the file system under etc/ftpd/.

9.2 File – „users“

9.2.1 Format

The composition of the "users" file consists of: username:password:group

- Username: 1 to 19 characters
- Password: 1 to 19 characters
- Group (optional): at least 1 character

Note: The "users" file can be a maximum of 254 bytes.

Allowed characters are numbers, special characters and letters in upper and lower case. The colon (:) is automatically a separator and can therefore not be used as a character for the user name, password and group.

9.2.2 Permissions

The structure of the "users" file consists of 1. the user name; 2. the password and 3. the assigned group (optional)

Example 1: Deutschmann:Automation:admin -> The user has access to the "/etc" directory.

Example 2: Deutschmann:Automation -> The user does not have access to the "/etc" directory.

Note: Only users in the admin group have access to the /etc directory and its contents. Access is limited to "Write". Other groups can be defined; however, no rights can be assigned to them. If the customer has created their own "user" file, they can also open and read it.

9.2.3 Load

To upload the "users" file, the UNIGATE® must be started in configuration mode. The FTP access takes place via the predefined user name "admin" and the password "nimda" if the UNIGATE® is still in the delivery status or the user name and/or the password has been lost.

9.3 Script update via FTP

The **script (.dss)** created via the Protocol Developer software **must be saved as a compiled script (.dcs)** under the file name "script.dcs". This file can then be transferred to the UNIGATE®. In the file system, the "script.dcs" file must be stored at a higher level, i.e. not in the "etc" or "srv" folders. When the write process is complete, the FTP connection can be disconnected. The UNIGATE® must then be restarted. During the start process, the UNIGATE® takes over the script contained in the file and then deletes the file from the file system.

9.4 Updating the system configuration via FTP

The device configuration called up via the WINGATE configuration software (*.gwc) must be saved under the file name "system.gwc". This file can then be transferred to the UNIGATE®. When the writing process is complete, the FTP connection can be closed. The UNIGATE® must then be restarted. During the start process, the UNIGATE® accepts the device configuration contained in the file and then deletes the file.

9.5 Use FTP server

The FTP server is only suitable for "single user access". It is recommended to set FTP programs such as "FileZilla" to only one simultaneous connection. This significantly reduces the network load.

10 WEB server

This UNIGATE® has an integrated web server. When delivered, the file system of the UNIGATE® contains an HTML page on which the system parameters of the device are displayed.
The web page can be changed as desired by the customer or replaced with their own web page. This can then be written to the file system via FTP (see Chapter 9).

The WEB server is disabled by default in Run mode. This can be activated by script via the Web-Server parameter.

10.1 Server-Side-Includes (SSI)

The "Server Side Includes" (SSI) functionality is available to also be able to generate dynamic web pages that display process data, for example.
The web server only searches files with the extension "shtml" or "shtm" following SSI instructions. In all other HTML files, SSI instructions are ignored.

When a client requests a web page that contains SSI directives, the web server replaces the directives with the appropriate values and sends the generated page to the client. SSI instructions are noted within HTML comments and have the following syntax: <%command%>

A complete list of all commands can be obtained with the SSI command <%ListSsiFunctions%> or by calling the sample HTML page ssiList.shtml. be called.
Additional script variables can be made known to the web server with the script command “SetHT-MLOutputString” for the output of any process data. The command assigns a name to the variable, which can then be used in the following command: <%DisplayScriptVar:variable name%>

10.2 HTML forms (CGI)

By supporting HTML forms, the web server allows the design of interactive web pages. For example, input masks for process values can be implemented in this way.
The script command "SetHTMLInputString" assigns a name to a script variable, which can then be used in the HTML form.

10.3 Example

The interaction between gateway script and HTML page can be seen in the example HTML page "ssi.shtml" or "cgi.shtml" and the example script "example_Set_HTML_String.dss".
The sample HTML page is pre-installed on the Unigate in the /srv/www directory.
The "example_Set_HTML_String.dss" file can be found in the Script Interpreter.

11 File system

The UNIGATE® has a file system that is used by the integrated FTP server (see chapter 9) and the integrated webserver (see chapter 10). The directories /etc and /srv are stored in the delivery state.

11.1 Directory /etc

User names and passwords can be stored in this directory. To do this, the "users" file must be added to the /etc/ftpd/ directory. Further information can be found in the chapter.

11.2 Directory /srv

A webpage can be stored in this directory. To do this, the subdirectory /www must be created. The files can then be added to a web page under /srv/www. When delivered, the Deutschmann website is stored under /srv/www.

12 PROFINET

If we are speaking of PROFINET, then the 100 MBit/s-version is automatically meant, full-duplex and switched Ethernet. The end device's protocol is being converted in the UNIGATE® via a Script.
The UNIGATE® supports PROFINET I/O-RT.

12.1 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.

12.2 PROFINET address assignment

Note: on delivery, the Gateway has no IP address!
During normal operation (data exchange mode) the IP address is assigned to the Gateway by the Profinet-IO-controller (PLC). For it the gateway has a device name on which it is addressed. (see chapter 12.3). The IP address can be assigned manually or via a DHCP server.

12.3 PROFINET device name

Note: on delivery, the Gateway has no device name!
The device name is assigned via the configuration software of the Gateway.

Accordingly to the PROFINET specification, the following rules apply to the device name:

* It consists of one or more name parts seperated through a full stop
* the total length is from 1 up to 240 characters
* the length of a part of the name is 1 up to 63 characters
* a name consists only of lowercase letters, numbers and the hyphen.
* neither the first nor the last character of a part of the name is a hyphen.
* the first part of the name does not begin with „port-xyz or „port-xyz-abcde“, whereas a, b, c, d, e, x, y and z are seen as numerals.
* it doesn‘t have the form „k, l, m, n“, whereas k, l, m and n are numbers between 0 and 999

12.4 Device description file

The sample device description file provided (which is based on the XML-oriented GSDML standard) can be used as the basis for creating your own device description file. Deutschmann Automation also offers the creation as a service.

Note:
The Deutschmann webpage is loaded in the delivery status of the UNIGATE® IC2-PROFINET 2Port. This can be used to load the sample device description file. Further information on creating your own device description file can be found in the attached document 'gsdQuickStart'.

12.5 MRP (Media Redundancy Protocol)

The UNIGATE® IC2-PROFINET 2Port supports the MRP function. The device description file must contain the corresponding entry in order to be able to use the function.
The corresponding entry "MediaRedundancy" is already included in the sample device description file "GSDML-V2.35-Deutschmann-MyDevice-20210217.xml" from Deutschmann Automation.

12.6 System redundancy S2

The UNIGATE® IC2-PROFINET 2Port supports the function system redundancy S2. The device description file must contain the corresponding entry in order to be able to use the function. The corresponding entry "SystemRedundancy" is already included in the sample device description file "GSDML-V2.35-Deutschmann-MyDevice-20210217.xml" from Deutschmann Automation.

12.7 Possible data lengths

The table below shows the maximum transferable data:

Input data	max. 1024 Bytes	Variable: maximum value in this case
Output data	max. 1024 Bytes	Variable: maximum value in this case

12.8 Startup phase

The Gateway is parameterized and configured by the PROFINET-IO-controller during the startup phase. Only after a correct termination of the startup phase the data exchange with external devices will take place.

12.9 Literature

We recommend the book "Der Schnelleinstieg in PROFINET", author M. Popp and K. Weber, to help you quickly get to grips with the subject of PROFINET. The book (German language) can be ordered from the PROFIBUS User Organisation, Order No. 4.181.

Address:
PROFIBUS Nutzerorganisation e.V.
Haid-und-Neu-Str. 7
D-76131 Karlsruhe
Germany
Phone: +49 (0) 721 9658 590

13 Error handling at UNIGATE® IC2

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® IC2

14 Firmware-update

14.1 Overview

In the firmware-update-mode the firmware can be replaced via the UNIGATE® IC2's serial interface.

14.2 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 16).
A Firmware-update can be carried out in the following operational modes:
Configuration mode (via RS232 serial interface)
Firmware-update-mode (via RS232 serial interface)

14.3 Adjusting the firmware-update-mode

14.3.1 Adjustment by hardware

UNIGATE® IC2 can be brought to the firmware-update-mode by the hardware. For it the signal $\overline{\text{BE}}$ (boot enable) has to be pulled to the potential GND during the Power-up-process.

14.4 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.

14.5 Operation mode of the IC2

Data exchange mode
This mode is required for the regular use of the IC2. In this mode the IC2 will process all script-commands and normally exchange the corresponding user data. The bus as well is operated in this mode through the IC2.

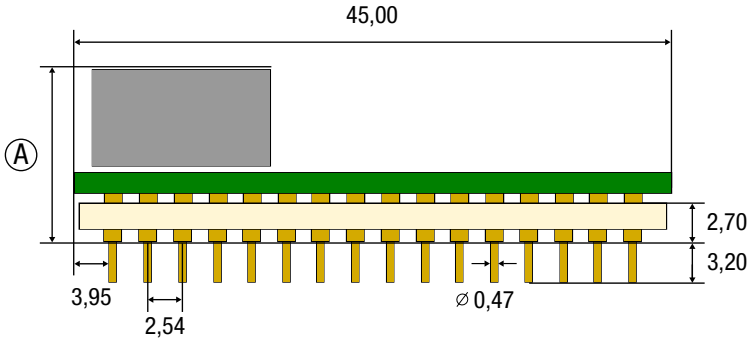
Configuration mode
In the configuration mode the UNIGATE® IC2 will carry out a self-test after the start (or after a reset). After a successful self-test it will wait for further commands. Here it is possible to load a translated Script (file extension *.dcs) into the unit or to configure the device.

15 Technical data

In this chapter you will find all necessary technical data on UNIGATE® IC2.
All measurements in mm.

15.1 Mechanics of the UNIGATE® IC2

15.1.1 General dimensions of UNIGATE® IC2



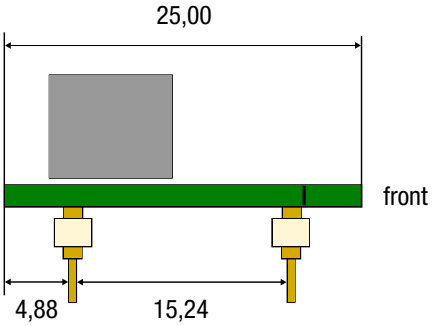
UNIGATE® IC2

	EtherCAT (wt*)	Ethernet TCP/IP + Modbus TCP	Ethernet TCP/IP (wt*) + Modbus TCP (wt*)	PROFIBUS DP	PRROFINET 2Port (wt*)
A= total height (mm)	8	11	8	13	8

*wt = without magnetics

Note: The total height of all UNIGATE® ICs can be found in the "Pinout list for UNIGATE® IC and UNIGATE® IC2". (Download PDF)

The pins of the UNIGATE® ICs are arranged in a grid dimension of 2.54 mm



DIP-Spacing Code 6

If you are also thinking of using other UNIGATE® IC variants, you must consider a maximum construction height (including pins) of ≤ 20 mm.

15.2 Technical data UNIGATE® IC2 - PROFINET 2Port

Characteristics	Explanation
Supply voltage	3.3V ± 5 %, max. 460 mA DC
Reverse polarity protection power supply	No
Interface	2 UART interfaces, 1 synchronous serial interface
IP-address	Assigned by the PROFINET-IO-controller (PLC) see chapter 12.2
Bus-baud rate	100 MBit/s
UART-baud rate	Up to 6 Mbaud (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 25 x 8 mm (WxDxH)
Installation	32 DIL
Weight	Approx. 9 g
Ambient temperature*	-40°C ..+85°C
Storage / transport temperature	-40°C..+100°C
Built-in position	Any

* When implementing a UNIGATE® IC in your electronics, not only the mechanical dimensions but also the thermal management must be taken into account. The ambient temperature specified in the manual does not take into account the specific installation situation (e.g. closed housing). External factors have a major influence on how well the heat is dissipated from the heat sources. When using the UNIGATE® IC in your electronics, adequate heat dissipation must therefore be ensured.

With the UNIGATE® IC2-PROFINET 2Port, the greatest heat development comes from the TPS-1 component on the top of the module.

16 Accessory

The following tools are available from Deutschmann Automation.

16.1 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 IC2. The software describes the procedure of an update itself.

16.2 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.

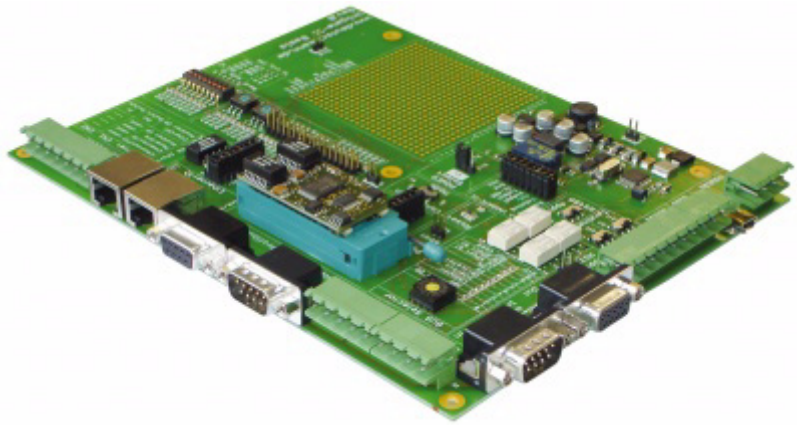
16.3 Developer Kit UNIGATE® IC-AB IC

The Developer Kit IC2 contains

- a Developer board UNIGATE® IC2-AB (see chapter 16.3.1)
- a plug-in power pack to supply the Developer Board
- connection cables for appl. RS232, Debug RS232 and appl. RS422/485
- USB-cable
- Software and documentation to complete the package.

16.3.1 Developer Board UNIGATE® IC2-AB

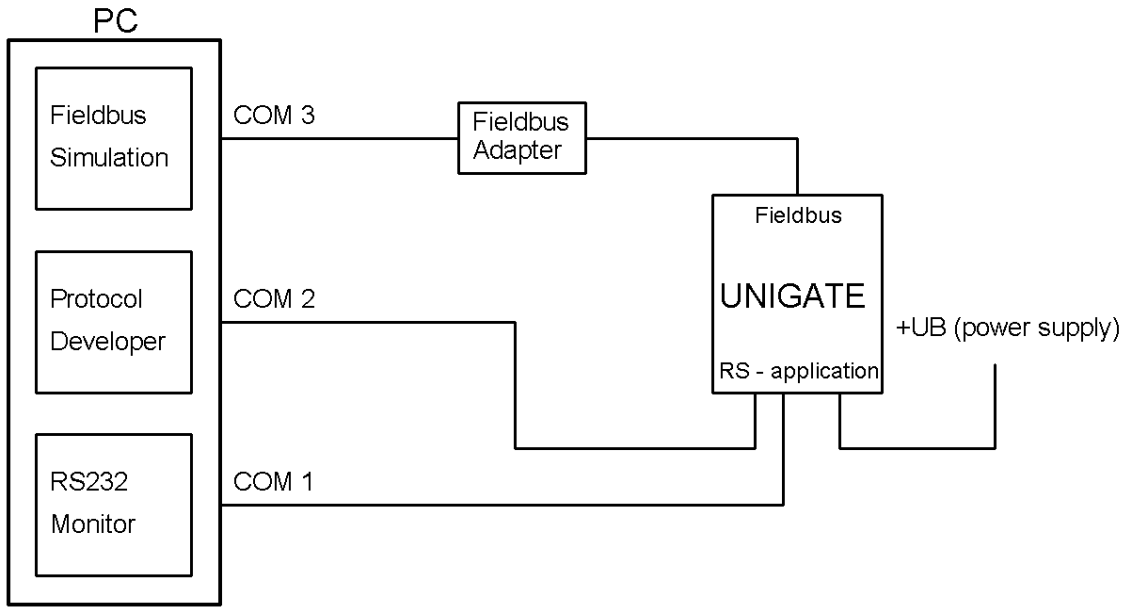
The Developer Board was developed so that the fast implementation of the Deutschmann All-in-one bus node UNIGATE® IC2 into your electronic system can be guaranteed. The board is suitable for all Fieldbuses and Industrial Ethernet Buses supported by Deutschmann Automation.



The required UNIGATE® IC2s 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).

16.3.2 Quick start



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

17 Appendix

17.1 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)
IC2	=	Product group IC2 (IC-design DIL32)
IO8	=	Option I/O8
16	=	Script memory expanded to 16KB
5V	=	Operating voltage 5V
3,.3V	=	Operating voltage 3.3V

Fieldbus

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/s
FEX	=	Ethernet 10/100 MBit/s-version X (see comparison table UNIGATE® IC for the respective product)
IB	=	Interbus
IBL	=	Interbus
LN62	=	LONWorks62
LN512	=	LONWorks512
ModTCP	=	ModbusTCP
MPI	=	Siemens MPI®
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

18 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 or directly in our Wiki on 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

Your request will be recorded in the Support center and will be processed by our Support Team as quickly as possible (Usually in 1 working day, rarely more than 3 working days.).

Technical Support hours 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 Support	+49 6434 9433-33
Fax sales department	+49 6434 9433-40
Fax Technical Support	+49 6434 9433-44
E-mail Technical Support	support@deutschmann.de

18.1 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 (±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.

18.2 Downloading PC software

You can download current information and software free of charge from our Internet server.
http://www.deutschmann.com