



**Deuschmann**

*your ticket to all buses*

**Bedienerhandbuch  
Universelles Feldbus-Gateway  
UNIGATE<sup>®</sup> IC2 - Fast Ethernet**



**Deuschmann Automation GmbH & Co. KG**  
[www.deuschmann.de](http://www.deuschmann.de) | [wiki.deuschmann.de](http://wiki.deuschmann.de)





<b>1</b>	<b>Allgemeine Einführung</b>	<b>9</b>
<b>2</b>	<b>Das UNIGATE® IC2.</b>	<b>10</b>
2.1	Technische Einführung	10
2.2	Verfügbarkeit	10
2.3	Firmware	10
2.4	Die serielle Standardschnittstelle	10
2.5	Die synchrone serielle Schnittstelle	10
2.6	Die Debug Schnittstelle	10
2.7	UNIGATE® IC2 Hardwareübersicht	11
<b>3</b>	<b>Hardware Design</b>	<b>12</b>
3.1	Anschlüsse	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	LED-LINKACT	14
3.2.9	Error-LED	14
3.2.10	LED-FE	14
3.2.11	-Config Mode	14
3.2.12	DbgTX, DbgRx.	14
3.2.13	TE	14
3.2.14	TX, RX.	14
3.3	Software	14
3.4	Prinzipielle Vorgehensweise	15
3.5	Anschlussbeispiele	15
3.6	Layout-Beispiele	19
3.7	Handling (Befestigung des UNIGATE® IC2 auf der Trägerplatine)	21
3.8	Anforderung an Ethernet-Schnittstelle	22
<b>4</b>	<b>Die serielle Schnittstelle</b>	<b>23</b>
4.1	Übersicht	23
4.2	Initialisierung der seriellen Schnittstelle	23
4.3	Benutzung der seriellen Schnittstelle	23
4.4	Weitere Betriebsmodi	23
4.5	UART	23
<b>5</b>	<b>Synchrone serielle Schnittstelle</b>	<b>24</b>
5.1	Schieberegisterbetrieb	24

---

5.1.1	Script-Beispiel . . . . .	24
5.2	SPI-Betrieb . . . . .	25
5.2.1	Script-Beispiel . . . . .	25
<b>6</b>	<b>Die Debug Schnittstelle . . . . .</b>	<b>26</b>
6.1	Übersicht über die Debug Schnittstelle . . . . .	26
6.2	Start im Debugmode . . . . .	26
6.3	Kommunikationsparameter für die Debugschnittstelle . . . . .	26
6.4	Möglichkeiten mit der Debug Schnittstelle . . . . .	26
6.5	Befehle der Debug Schnittstelle . . . . .	26
<b>7</b>	<b>Script und Konfiguration . . . . .</b>	<b>27</b>
7.1	Übersicht . . . . .	27
7.2	Der Konfigurationsmodus . . . . .	27
7.3	Script updaten . . . . .	27
7.4	Konfiguration des UNIGATE® IC2 . . . . .	29
7.4.1	Ethernet . . . . .	29
7.4.2	RS232/RS485/RS422 . . . . .	30
<b>8</b>	<b>Erstellung eines Scripts . . . . .</b>	<b>31</b>
8.1	Was ist ein Script . . . . .	31
8.2	Speichereffizienz der Programme . . . . .	31
8.3	Was kann man mit einem Script Gerät machen . . . . .	31
8.4	Unabhängigkeit von Bussen . . . . .	31
8.5	Weitere Einstellungen am IC2 Gateway . . . . .	32
8.6	Die Benutzung des Protocol Developer . . . . .	32
8.7	Genauigkeiten der Baudraten bei UNIGATE® IC2 . . . . .	32
8.8	Scriptarbeitungszeiten . . . . .	32
<b>9</b>	<b>Ethernet . . . . .</b>	<b>34</b>
9.1	Netzwerkadressen, Netzwerkverbindungen . . . . .	34
9.2	TCP-Verbindungsaufbau . . . . .	34
9.2.1	TCP-Server-Kanalmodus . . . . .	34
9.3	Datenaustausch . . . . .	34
9.3.1	TCP oder UDP? . . . . .	34
9.4	Datenaustauschmodus (data exchange mode) . . . . .	34
9.4.1	Direkter Datenaustausch . . . . .	35
9.4.2	MODBUS-TCP. . . . .	35
9.4.3	Server . . . . .	35
9.4.3.1	Webserver. . . . .	35
9.4.3.2	FTP-Server . . . . .	35
9.4.3.3	Script Update via FTP. . . . .	37
9.4.3.4	Systemkonfiguration Update via FTP . . . . .	37
9.4.3.5	Dateisystem . . . . .	37

9.4.3.6	TCP-Server-Kanalmodus . . . . .	37
9.4.3.7	UDP-Kanalmodus . . . . .	37
9.4.3.8	Raw-Kanalmodus . . . . .	37
9.4.3.9	BusStatus . . . . .	37
9.5	Einstellen der Ethernet-Adresse . . . . .	37
<b>10</b>	<b>Fehlerbehandlung beim UNIGATE® IC2 . . . . .</b>	<b>39</b>
<b>11</b>	<b>Firmware Update. . . . .</b>	<b>40</b>
11.1	Übersicht . . . . .	40
11.2	Einstellen des Firmware-Update-Modus . . . . .	40
11.2.1	Einstellung per Hardware . . . . .	40
11.2.2	Einstellung per Software. . . . .	40
11.3	Durchführung des Firmware-Updates . . . . .	40
11.4	Sicherheitshinweis . . . . .	40
11.5	Betriebsmodi des IC2 . . . . .	40
<b>12</b>	<b>Technische Daten . . . . .</b>	<b>41</b>
12.1	Mechanik des UNIGATE® IC2 . . . . .	41
12.1.1	UNIGATE® IC2 allgemeine Abmessungen . . . . .	41
12.1.2	Abmessungen UNIGATE® IC2 (nur Fast Ethernet). . . . .	41
12.2	Technische Daten UNIGATE® IC2 - Fast Ethernet . . . . .	42
<b>13</b>	<b>Zubehör . . . . .</b>	<b>43</b>
13.1	Adapter RS232 . . . . .	43
13.2	Adapter RS485 . . . . .	43
13.3	FirmwareDownloadTool (FDT) . . . . .	43
13.4	Protocol Developer . . . . .	43
13.5	Developerkit UNIGATE® IC-AB . . . . .	43
13.5.1	Developerboard UNIGATE® IC-AB . . . . .	43
13.5.2	Schnellstart . . . . .	44
<b>14</b>	<b>Anhang . . . . .</b>	<b>45</b>
14.1	Erläuterungen der Abkürzungen . . . . .	45
<b>15</b>	<b>Service . . . . .</b>	<b>47</b>
15.1	Einsendung eines Gerätes . . . . .	47
15.2	Download von PC-Software . . . . .	47



#### Haftungsausschluss

Wir haben den Inhalt der Druckschrift auf Übereinstimmung mit der beschriebenen Hard- und Software geprüft. Dennoch können Abweichungen nicht ausgeschlossen werden, so dass wir für die vollständige Übereinstimmung keine Gewähr übernehmen. Die Angaben in der Druckschrift werden jedoch regelmäßig überprüft. Notwendige Korrekturen sind in den nachfolgenden Auflagen enthalten. Für Verbesserungsvorschläge sind wir dankbar.

#### Copyright

Copyright (C) Deutschmann Automation GmbH & Co. KG 1997 – 2020. All rights reserved.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung Ihres Inhalts sind nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder PM-Eintragung.



## 1 Allgemeine Einführung

In der Vergangenheit erforderte die Integration eines Feldbusanschlusses einen enormen Aufwand der Entwicklungsingenieure. Durch die vielfältigen Kommunikationstechniken ist es nicht damit getan, die richtige Kombination von Kommunikationshardware zusammenzustellen; die verschiedenen Busse erfordern durch ihre Normen und Grundlagen auch entsprechendes Können der Ingenieure.

Im Falle des Deutschmann UNIGATE® IC2 trifft das nicht mehr zu. Alle digitalen Funktionen, Software Stack und Treiber sowie Optokoppler entsprechend der Norm sind alle auf einem IC2 integriert. Zusätzlich zur Verringerung der benötigten Größe können auch verschiedene Feldbusse einfach integriert werden.

**Durch die flexible Firmware des UNIGATE® IC2 sind auf der Kundenseite keinerlei Änderungen der Software notwendig!**

Deutschmann Automation ist seit 1997 im Bereich der Feldbusgateways erfahren; aus dieser enormen Erfahrung heraus resultiert das UNIGATE® IC2 als eine konsequente Fortführung dieser erfolgreichen Produktlinie.

### Terminologie

Im gesamten Dokument und in allen Teilen der zu verwendenden Software werden die Begriffe Input und Output benutzt. Input und Output sind mehrdeutig; jeweils abhängig von der Sichtweise. Wir sehen den Feldbus als zentrale Schnittstelle und als integralen Bestandteil Ihres Gerätes; daher bezeichnen wir an allen Stellen - unabhängig vom verwendeten Bus - die Daten aus der Sicht des Slave, also Input Daten als Daten vom Master zum Slave.

### Darstellung von Zahlen

Zahlen im dezimalen Format werden immer sowohl ohne Prefix als auch ohne Suffix dargestellt. Hexadezimale Zahlen werden immer durch das Prefix 0x gekennzeichnet.

Das UNIGATE® IC2 ist eine Weiterentwicklung der 1. Baureihe des UNIGATE® IC. Das mit einem Cortex-M4-Controller ausgestattete UNIGATE® IC2 erreicht wesentlich höhere Übertragungsraten bei der Kommunikation über SPI oder UART. Zudem verringert sich die Script-Abarbeitungszeit (in Abhängigkeit der verwendeten Befehle) in etwa um den Faktor 50 - 80. Somit sind Script-Abarbeitungszeiten im Mikrosekundenbereich möglich. Die Abarbeitungszeit einer Script-Zeile liegt bei ca. 10 µs

Das UNIGATE® IC2 ist außerdem PIN-kompatibel zur 1. Baureihe des UNIGATE® IC.

## 2 Das UNIGATE® IC2

### 2.1 Technische Einführung

Das Deutschmann Automation UNIGATE® IC2 beinhaltet alle für die Kommunikation in einem Feldbus nötigen Komponenten in einer einzelnen Baugruppe, das bedeutet, dass ein Entwickler sich nicht mehr um diese Details zu kümmern braucht; es ist lediglich ein Hardware Redesign notwendig, um das UNIGATE® IC2 und die nötigen Steckverbinder zu integrieren.

### 2.2 Verfügbarkeit

Das Modul ist als Ethernet verfügbar. Weitere Feldbusse sind in Arbeit bzw. Planung; diese werden sich nur in den Anschlüssen des Busses unterscheiden. Die allgemeinen Pins 1 - 9 sowie 24 und 26 - 32 werden ihre Bedeutung auch für weitere Feldbusimplementierungen unverändert behalten. Eine aktuelle Liste für alle UNIGATE® IC2 finden Sie unter <http://www.deutschmann.de>

### 2.3 Firmware

UNIGATE® IC2 wird über Scripte programmiert. Im Prinzip ist jedes Script, das für ein UNIGATE® IC entwickelt wurde, auch auf dem UNIGATE® IC2 lauffähig.

### 2.4 Die serielle Standardschnittstelle

Intelligente Geräte, die bereits einen Microcontroller oder Microprozessor haben, verfügen im allgemeinen auch über eine serielle, asynchrone Schnittstelle mit TTL Pegeln. Diese wird direkt mit der TTL Schnittstelle des UNIGATE® IC2s verbunden. Weitere Informationen zu dieser seriellen Schnittstelle finden Sie im Kapitel 4.

### 2.5 Die synchrone serielle Schnittstelle

Zusätzlich zu der Standardschnittstelle gibt es auch die Möglichkeit der synchronen Datenein- und Ausgabe. So können zum Beispiel digitale IOs über Schieberegisterbausteine oder auch analoge IOs über DA-Wandler mit serielltem Ein-/Ausgang angeschlossen werden. Für synchrone IOs können maximal 256 Signale benutzt werden (256 Bit). Schaltungsbeispiele finden Sie im Kapitel 3.5 und Softwarebeispiele finden Sie im Kapitel 5. Diese Schnittstelle kann ebenso dazu genutzt werden Bausteine bzw. Geräte mit SPI-Schnittstelle anzubinden. Es ist auch möglich z. B. digitale oder analoge E/A-Module zu bauen, ohne dass das Kundengerät über einen eigenen Controller verfügt; das Feldbus IC2 ist auch ohne diesen autonom lauffähig.

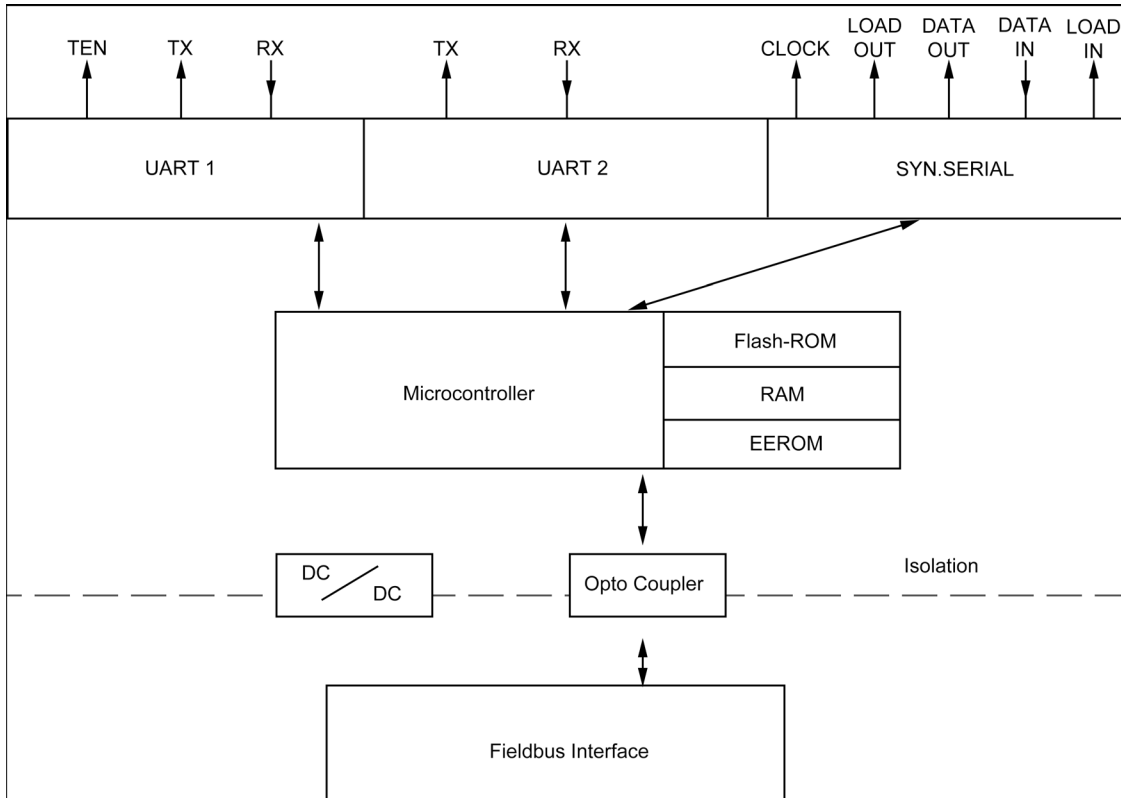
### 2.6 Die Debug Schnittstelle

Das UNIGATE® IC2 verfügt über eine Debug Schnittstelle, die es erlaubt, ein Script schrittweise abzuarbeiten und auch Daten zu beobachten oder zu manipulieren. Dies ist für die Entwicklung eines Scripts unerlässlich. Normalerweise wird ein Script mit der Software Protocol Developer entwickelt. Details hierzu finden Sie im Handbuch Protocol Developer.

**Alle Schnittstellen können unabhängig voneinander zur selben Zeit benutzt werden.**

### 2.7 UNIGATE® IC2 Hardwareübersicht

Die Hardware des UNIGATE® IC2 besteht aus einigen wenigen Standardbauteilen. Die folgende Grafik zeigt den funktionellen Aufbau des IC2.



### 3 Hardware Design

In diesem Kapitel werden die grundlegenden Hinweise gegeben, die nötig sind, um das UNIGATE® IC2 in eigene Hardware Designs zu übernehmen. Es werden im folgenden alle Anschlüsse des UNIGATE® IC2s ausführlich beschrieben.

#### 3.1 Anschlüsse

Das UNIGATE® IC2 hat 32 Pins im Layout wie ein DIL 32 Baustein. Pins 10 - 12 sowie 21 - 23 sind wegen der elektrischen Isolation nicht belegt. Die genauen mechanischen Abmessungen können Kapitel 12 entnommen werden.



Für ALLE 32 Pins sind im Layout Bohrungen vorzusehen.

#### 3.2 Pinout

Pin	Technische Spezifikationen	Name	Beschreibung	Anmerkung
1	3,3V ± 5% < 210mA	Vcc	+ 3,3 V Spannungsversorgung	
2	IN <sub>Logic</sub>	-BE	Boot enable	
3	OUT <sub>Driver</sub>	Load out (SS0-)	Übernahmesignal der Ausgangsdaten der synchronen seriellen Schnittstelle	
4	OUT <sub>Driver</sub>	Data out (SS1-)	Ausgangsdaten für synchrone, serielle Schnittstelle	
5	IN <sub>Logic</sub>	Data in (MISO)	Eingangsdaten der synchronen, seriellen Schnittstelle	intern pulled up mit 10 kΩ
6	OUT <sub>Logic</sub>	Load in (MOSI)	Übernahmesignal der Eingangsdaten der synchronen seriellen Schnittstelle	intern pulled up mit 10 kΩ
7	OUT <sub>Driver</sub>	Clock (SCK)	Taktsignal für synchrone, serielle Schnittstelle	
8	IN <sub>Reset</sub>	-Reset In	Reset-Eingang des IC2	
9	connected to pin 1	Vcc	+ 3,3 V Spannungsversorgung	
10			verbunden mit Pin18	
11			verbunden mit Pin 17	
12	nc	nc	nicht belegt	
13	according to norm	RJ45_6	Ethernet-Signal RD-	galvanisch getrennt Isolationsspannung 1000 Vrms
14	according to norm	RJ45_3	Ethernet-Signal RD+	galvanisch getrennt Isolationsspannung 1000 Vrms
15	according to norm	RJ45_2	Ethernet-Signal TD-	galvanisch getrennt Isolationsspannung 1000 Vrms
16	according to norm	RJ45_1	Ethernet-Signal TD+	galvanisch getrennt Isolationsspannung 1000 Vrms
17	according to norm	LED-LINKACT	Ethernet-Linkpulse gefunden / Netzwerk Datenverkehr	ab Hardware-Rev. B Anschluss siehe Kapitel 3.6
18	according to norm	Error-LED	Anode der roten LED (wird zur Zeit nicht unterstützt)	(wird zur Zeit noch nicht unterstützt) Anschluss siehe Kapitel 3.6
19	according to norm	nc	nicht belegt	
20	according to norm	Ground (PE)	Erde	ab Hardware-Rev. B
21	nc	nc	nicht belegt	
22	nc	nc	nicht belegt	
23			verbunden mit Pin 20	

24	connected to pin 32	GND	Ground Spannungsversorgung des IC2	
25		LED-FE	Bus Status LED (Anode der grünen LED)	
26	IN <sub>Logic</sub>	-Config Mode	Signal zum Starten im Konfigurationsmodus	intern pulled up mit 10 k $\Omega$
27	OUT <sub>Logic</sub>	DbgTX	serial Debug TX	
28	IN <sub>Logic</sub>	DbgRX	serial Debug RX	intern pulled up mit 10 k $\Omega$
29	IN <sub>Logic</sub>	RX	serielle Daten RX	intern pulled up mit 10 k $\Omega$
30	OUT <sub>Logic</sub>	TX	serielle Daten TX	
31	OUT <sub>Logic</sub>	TE	Transmit enable	
32	GND	GND	Ground Spannungsversorgung des IC2	

Die Ethernet Signale sind galvanisch getrennt. Die Isolationsspannung beträgt 1000 Vrms.

	V <sub>IL</sub>	V <sub>IH</sub>
IN <sub>Reset</sub>	< 0,7V / 0,5mA	>2,3V / 5 $\mu$ A
IN <sub>Logic</sub>	< 1V / 5 $\mu$ A	>2,3V / 5 $\mu$ A
	V <sub>OL</sub>	V <sub>OH</sub>
Out	< 0,45V / 4mA	> 2,8V / 4mA

### 3.2.1 -Boot enable

Mit dem Pegel GND während dem Power Up Vorgang wird das IC2 im Firmware Update Mode gestartet. Siehe auch Kapitel 11.

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

Übernahmesignal für die synchrone serielle Schnittstelle. Mit der positiven Flanke an diesem Ausgang werden Daten von den angeschlossenen Schieberegistern an die physikalischen Ausgänge übernommen.

Im SPI-Modus dient dieser Pin als Low-aktives Slave-Select-Signal.

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

Auf dieser Leitung werden die Daten auf der synchronen seriellen Schnittstelle ausgegeben. Das höchstwertige Bit der Daten wird zuerst ausgegeben.

Im SPI-Modus dient dieser Pin als Low-aktives Slave-Select-Signal.

### 3.2.4 Data In (SPI: MISO)

Über dieses Signal werden die Daten auf der synchronen seriellen Schnittstelle eingelesen. Das höchstwertige Bit der Daten wird zuerst erwartet.

Im SPI-Modus dient dieser Pin zur Datenübertragung vom Slave zum Master.

### 3.2.5 Load In (SPI: MOSI)

Dieser Pin ist das Übernahmesignal für die Eingangsdaten der synchronen seriellen Schnittstelle.

Im SPI-Modus dient dieser PIN zur Datenübertragung vom Master zum Slave.

### 3.2.6 Clock (SPI: SCK)

Dieses Signal ist die Taktleitung für die synchrone serielle Schnittstelle. Dies Signal gilt gleichermaßen für die Datenein- und Ausgabe.

### 3.2.7 -Reset In

- Ein Resetgenerator ist on board, somit wird der Reset-Eingang im Normalfall nicht benötigt. Der Reset-Eingang muss in diesem Fall mit VCC verbunden werden, um Störeinkoppelungen zu vermeiden (siehe Kapitel 3.6).

- Muss die Kundenapplikation einen Reset des UNIGATE® IC2 auslösen, so kann der Reset-Eingang statt mit VCC auch mit einem Reset-Ausgang der Kundenapplikation verbunden werden. Dabei müssen die in Kapitel 3.2 angegebenen Spezifikationen des Reset-Signals eingehalten werden. Der Reset-Impuls sollte mind. 10 µs lang sein!

### **3.2.8 LED-LINKACT**

An diese Leitung kann eine grüne LED angeschlossen werden (siehe Kapitel 3.6). Diese leuchtet, wenn sich das Gateway an einem arbeitsfähigen Netz befindet (es werden Link-Pulse empfangen) und blinkt bei Netzwerk Datenverkehr (Activity). Es erfolgt keine Unterscheidung zwischen 10 und 100 MBaud.

### **3.2.9 Error-LED**

Diese Leitung wird zur Zeit noch nicht unterstützt und ist nur für zukünftige neue Projekte gedacht.

### **3.2.10 LED-FE**

An diese Leitung kann eine rote LED angeschlossen werden (siehe Kapitel 3.6). Diese leuchtet im Zustand "Ethernet ok" und blinkt im Zustand "Config-Mode".

### **3.2.11 -Config Mode**

Mit dem Pegel GND an diesem Pin startet das IC2 im Konfigurationsmode.

### **3.2.12 DbgTX, DbgRx**

Dies sind sowohl Sendeleitung als auch Empfangsleitung der Debug Schnittstelle des IC2. Zur Funktionsbeschreibung der Debug Schnittstelle siehe Kapitel 6.

### **3.2.13 TE**

Das Transmit enable Signal ermöglicht den Anschluss von RS485 Treibern an der seriellen Schnittstelle des IC2. Das Signal wird immer dann auf High geschaltet, wenn das IC2 über die Leitung TX sendet.

### **3.2.14 TX, RX**

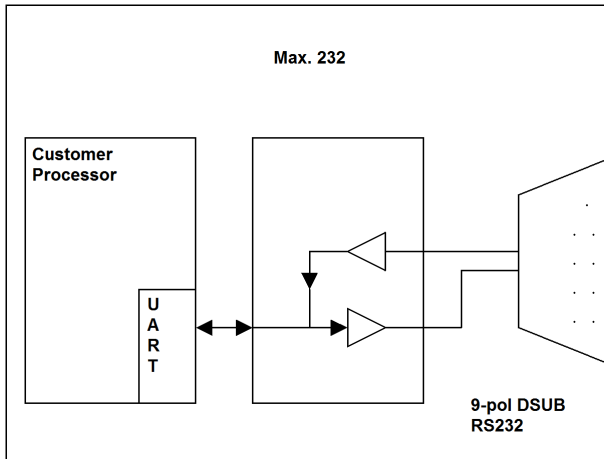
Sende- und Empfangsleitungen der seriellen Schnittstelle. Diese Schnittstelle ist entsprechend der Beschreibung in Kapitel 4 frei programmierbar.

## **3.3 Software**

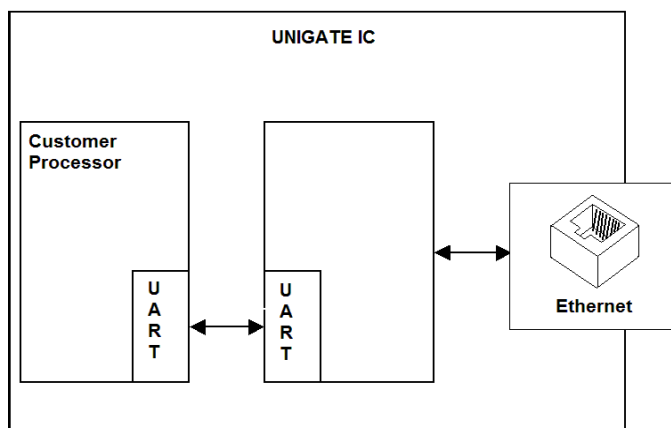
Die Software führt Scriptbefehle aus, diese wiederum steuern die Hardware des IC2 und wickeln per Software ihr komplettes Protokoll ab. Das Script selbst kann durch die Firma Deutschmann oder mit der Software Protocol Developer von Ihnen erstellt werden. Für eine detaillierte Beschreibung der Scriptbefehle des Protocol Developer siehe Benutzerhandbuch Protocol Developer und die Online-Dokumentation zu den Scriptbefehlen.

### 3.4 Prinzipielle Vorgehensweise

Theoretisch ist es ausreichend, den in Ihrer Anwendung enthaltenen RS232-Treiber durch das UNIGATE<sup>®</sup> IC2 zu ersetzen.



Ihr Gerät, das im wesentlichen wie oben aufgebaut sein dürfte wird nun modifiziert, so dass an der 9-pol Buchse der Ethernet zur Verfügung steht. Um die Belegung normgerecht zu halten, ist allerdings ein Hardware-Redesign nötig.



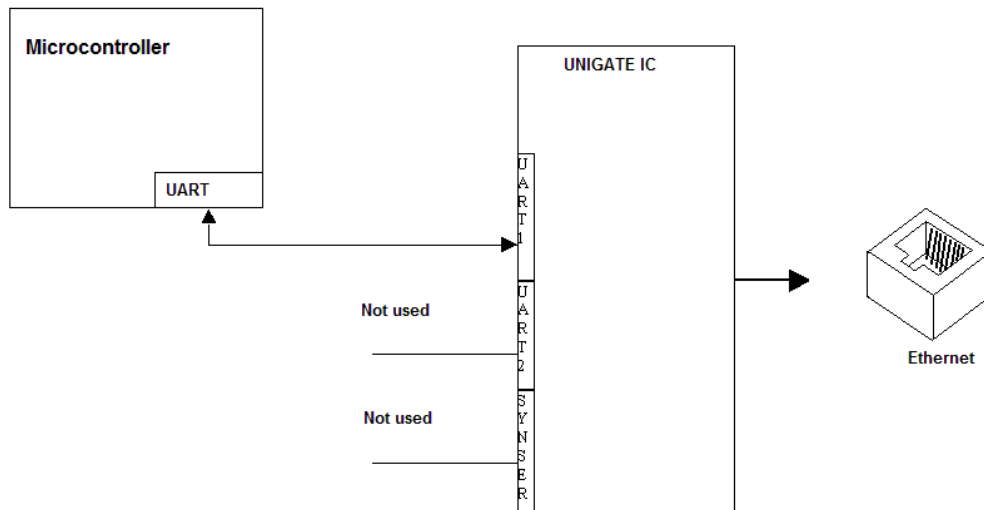
Nach dem Austausch des RS232 Treibers durch das UNIGATE<sup>®</sup> IC2 steht an einer RJ45-Buchse Ethernet zur Verfügung.

Deutschmann bietet auch eine entsprechende Adapterplatine an; mit ihr können bestehende Geräte ohne Redesign umgerüstet werden; siehe hierzu Kapitel 13.

### 3.5 Anschlussbeispiele

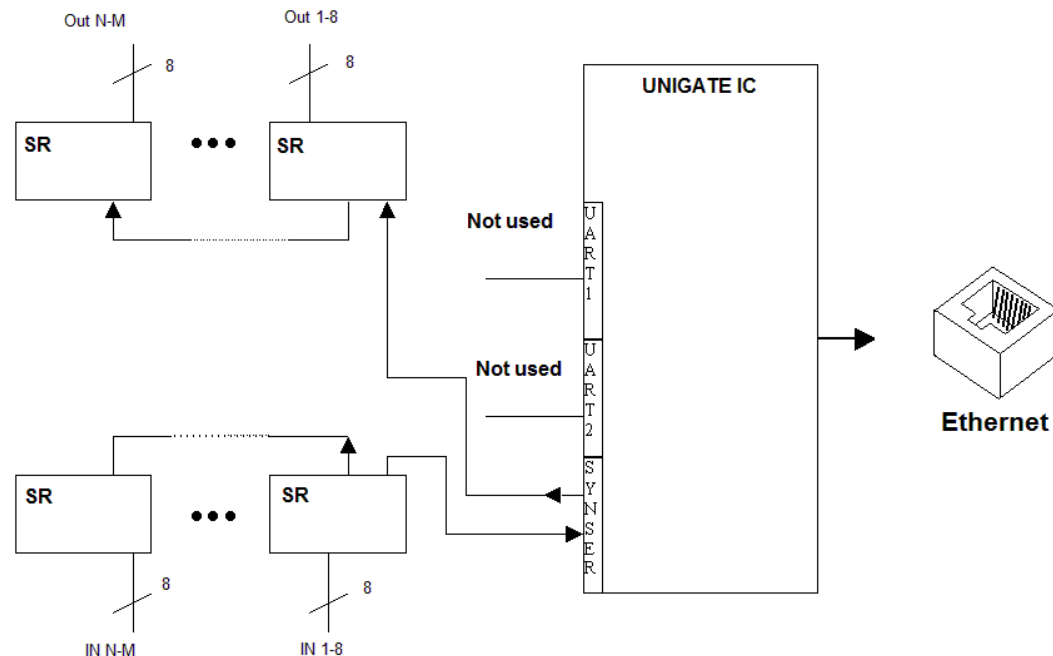
Hier finden Sie einige Hinweise, die Ihnen für ein Redesign helfen. Im folgenden sind einige Varianten aufgeführt, die Ihnen die Entscheidung etwas leichter machen sollen.

Variante 1: Einsatz als reines Koppelmodul für den Bus



Das UNIGATE<sup>®</sup> IC2 wickelt selbständig die Kommunikation mit dem Kundengerät über die TTL-Schnittstelle ab.

Variante 2: Einsatz des UNIGATE<sup>®</sup> IC2 für digitale oder analoge E/A-Module

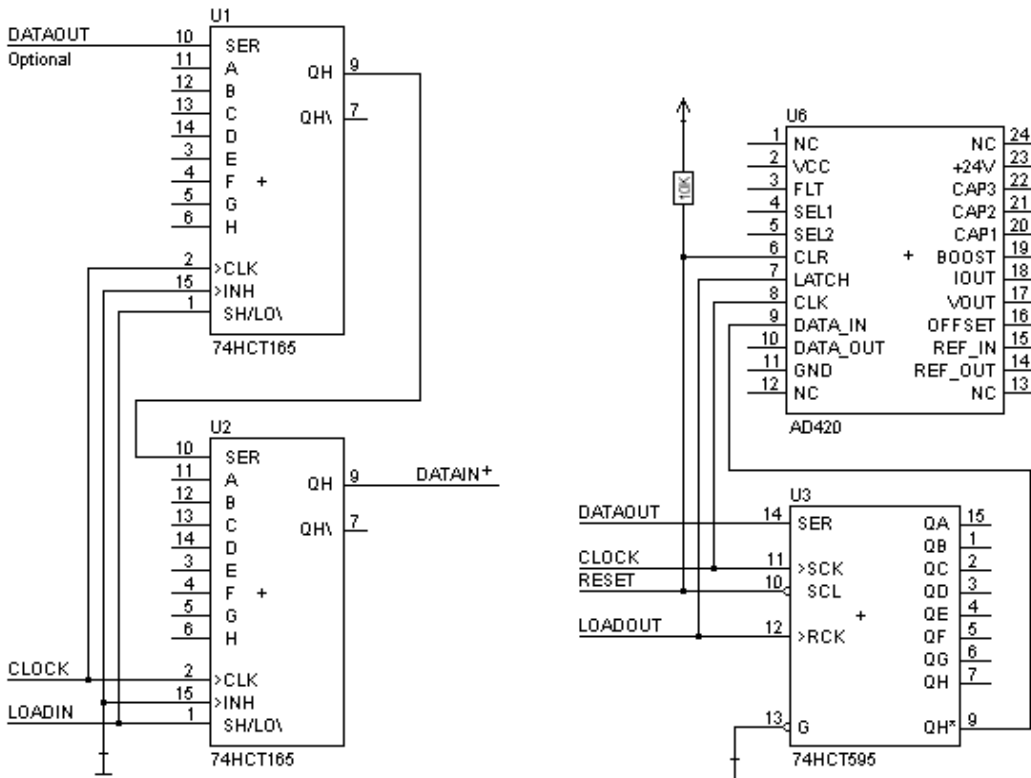


Hier wird lediglich die synchrone serielle Schnittstelle benutzt, die asynchrone serielle Schnittstelle ist hier im Prinzip ohne Bedeutung. Falls sie das Script jedoch in Ihrer fertigen Anwendung programmieren möchten, empfiehlt sich der Einsatz einer Pfostenleiste für die asynchrone Schnittstelle, Sie können damit die ISP Programmierung vornehmen.



**Es ist für diese Betriebsart kein zusätzlicher Controller auf Ihrer Applikation notwendig!**

Das folgende Schaltbild dient als Beispiel wie Schieberegister Bausteine an das IC2 angekoppelt werden können.

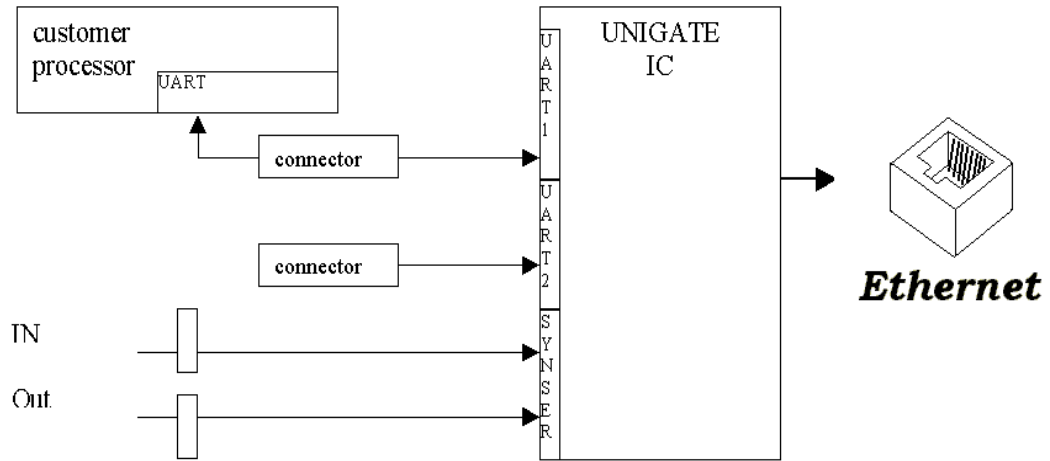


Variante 3: Beispiel für digitale I/Os

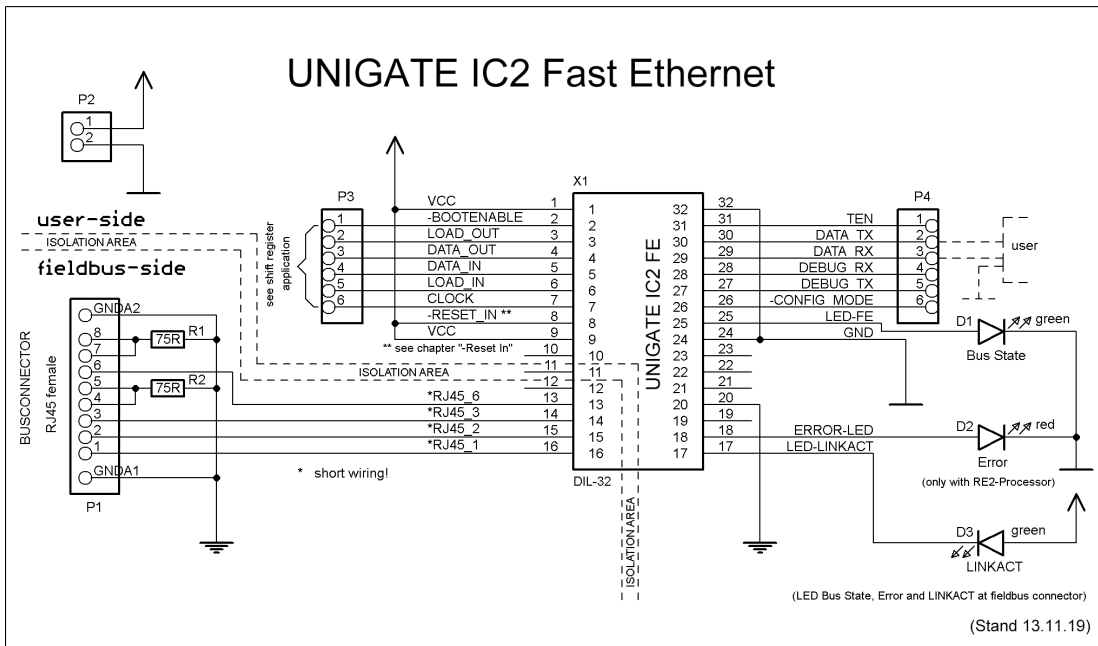
Die serielle synchrone als auch die asynchrone Schnittstelle kann vom UNIGATE® IC2 zur selben Zeit bedient werden. Hieraus ergibt sich die Möglichkeit, eine bestehende Applikation um zusätzliche digitale oder analoge E/As zu erweitern.

Im Kapitel 5.1.1 finden Sie ein Beispiel für ein Script, das diese E/As bedient.

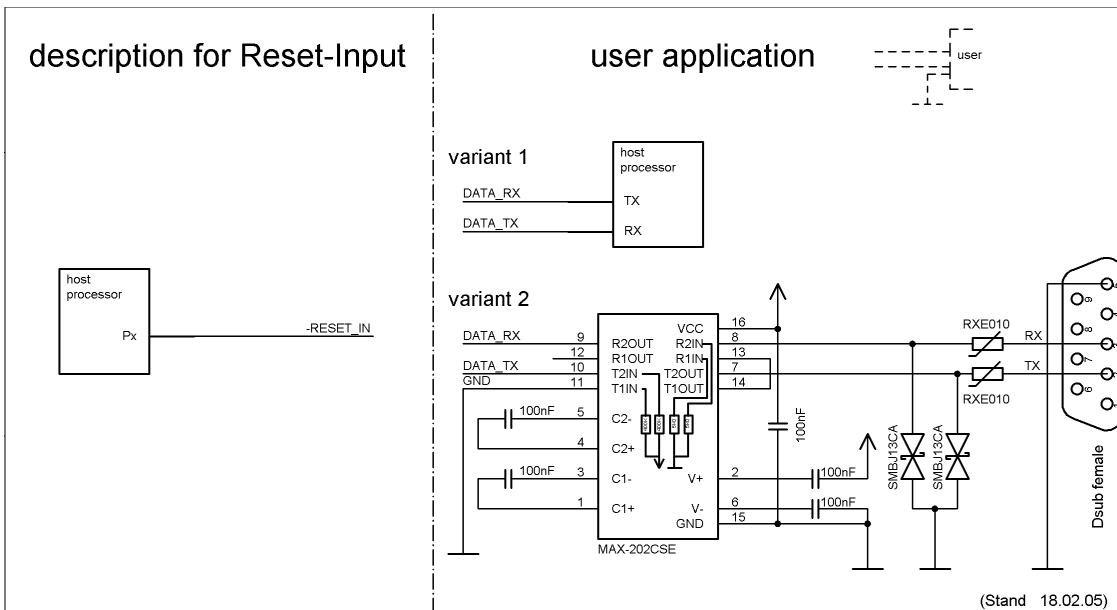
Für alle Varianten gilt: Eine vorgesehene Steckverbindung der seriellen Schnittstelle in der Anwendung schafft die Möglichkeit, über eine externe Verbindung ein Update der Firmware oder der Software durchzuführen.

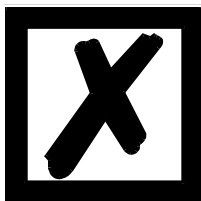
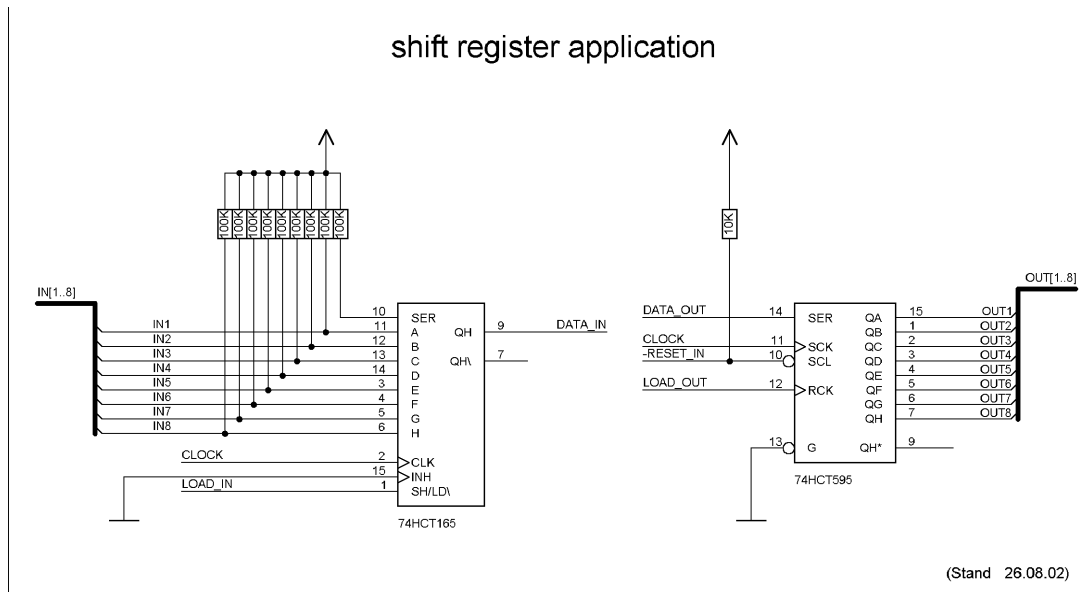


### 3.6 Layout-Beispiele



Bei der Verbindung zwischen UG-IC2 und RJ45-Buchse unbedingt darauf achten, dass die Signalleitungen RD-/RD+ bzw. TD-/TD+ möglichst kurz, parallel und gleichlang ausgeführt werden. Da es sich hier um Analogsignale handelt, ist es auch empfehlenswert, digitale Leitungen nicht in unmittelbarer Nähe zu platzieren.





Beim in diesem Beispiel eingesetzten 74HCT595 ist der Einschaltzustand undefiniert, dafür können aber über den OutEnable-Pin 13 die Ausgänge in den Tri-State Zustand versetzt werden. Ist es in einer Applikation wichtiger einen definierten Einschaltzustand zu bekommen, und der OutEnable-Pin nicht notwendig, kann hier z.B. der 74HCT594 verwendet werden.

**Hinweis:**

Aus dem Datenblatt des Wiznet W1000, der im UNIGATE IC-FE eingesetzt wird, ergibt sich folgende Empfehlung:

**Transformer Eigenschaften**

Parameter	Transmit End	Receive End
Turn Ratio	1:1	1:1
Inductance	350 uH	350 uH
Symmetrical TX & RX channels for auto MDI/MDIX capability		

### 3.7 Handling (Befestigung des UNIGATE® IC2 auf der Trägerplatine)

Generell gibt es zur Montage des UNIGATE® IC2's auf der Trägerplatine folgende Möglichkeiten, die in Abhängigkeit von der Applikation und der zu erwartenden Schock- und Vibrationsverhältnisse ausgewählt werden können:

- Aufstecken auf einen Sockel im Trägerboard. Eventuell das UNIGATE® IC2 im Sockel an 2 oder 4 Pins festlöten. Das IC2 lässt sich in der Regel dann nach Entfernen der Lötunkte problemlos wieder herausziehen.
- Im Layout neben dem Sockel zwei Löcher vorsehen. Nach dem Aufstecken des UNIGATE® IC2's auf den Sockel einen isolierten Draht über das IC2 ziehen und in den vorgesehenen Löchern der Trägerplatine festlöten.
- Mit einem Draht oder Kabelbinder das UNIGATE® IC2 auf der Fassung fixieren.
- Manuelles Einlöten direkt auf dem Trägerboard.
- Automatisches Einlöten direkt auf dem Trägerboard, wobei dann "selectiv" gelötet werden muss (kein Wellenlöten)

Die gesockelte Variante hat den Vorteil, dass Script- oder Firmware-Updates leicht eingespielt werden können, wenn das nicht auf dem Trägerboard mit vorgesehen ist. Ausserdem kann so durch Wechsel des UNIGATE® IC2 leicht der Feldbus gewechselt werden, wenn die entsprechenden Steckverbinder auf dem Trägerboard vorgesehen sind. Ein weiterer Vorteil ist, dass - in der Regel - die Trägerplatine dann nur Reflow gelötet werden muss.

Die gelötete Variante hat den Vorteil, dass die Bauhöhe geringer ist, und eine höhere Schock- und Vibrationssicherheit gegeben ist.

### 3.8 Anforderung an Ethernet-Schnittstelle

Da es auf dem Markt eine extrem große Anzahl von Anbietern für RJ45-Buchsen (mit und ohne integriertem Ethernet-Übertrager), bzw. einzelnen Ethernet-Übertragern gibt, ist es uns nicht möglich, alle Möglichkeiten zu testen. Wenn man bedenkt, dass es alleine bei RJ45-Buchsen die verschiedensten Ausführungen (SMD/THT, mit/ohne LED, stehend/liegend, mit/ohne EMV-Federn, Tab oben/unten, einfach/mehrfach usw.) gibt, ist das sicherlich verständlich. Generell müssen die RJ45-Buchsen und der externe oder integrierte Ethernet-Übertrager folgende Eigenschaften erfüllen:

#### **Geforderte Parameter für Ethernet-Übertrager:**

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)

Dielectric 1500 Vrms, 1 minute

#### **Geforderte Parameter für RJ45-Buchse:**

CAT5, möglichst geschirmt

Um einen schnellen Einstieg bei der Auswahl der entsprechenden Komponenten zu ermöglichen, haben wir nachfolgend eine Auflistung mit Artikelnummern von Würth-Elektronik zusammengestellt, wo wir unsere Komponenten beziehen. Selbstverständlich sind auch andere Hersteller möglich, so dass die nachfolgende Auflistung nur beispielhaft zu betrachten ist.

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

Wir verwenden hauptsächlich: Order-Nr.: 749 010 014

#### **RJ45-Buchse ohne Übertrager**

Wir verwenden hauptsächlich: Order-Nr.: 615 008 140 121

#### **RJ45-Buchse (einfach oder mehrfach) mit integriertem Übertrager**

Wir setzen, um kostengünstiger und flexibler zu sein, keine RJ45-Buchsen mit integriertem Übertragen ein. Technisch spricht aber nichts dagegen.

So haben wir beispielhaft getestet: Order-Nr.: 749 905 100 1

Weitere Hinweise:

Mit SMD-Buchsen haben wir in der Vergangenheit schlechte Erfahrungen gemacht, da diese in der Praxis durch mechanische Belastungen doch immer mal wieder abgerissen wurden.

## 4 Die serielle Schnittstelle

### 4.1 Übersicht

Die serielle Schnittstelle ist die wichtigste Verbindung zwischen dem UNIGATE® IC2 und dem Microcontroller Ihrer Anwendung. Die Schnittstelle ist so ausgelegt, dass Ihre Anwendung zumindest auf der Softwareseite nicht geändert werden muss; der große Leistungsumfang der seriellen Schnittstelle des UNIGATE® IC2 bietet hierfür die Grundlage. Das UNIGATE® IC2 ermöglicht den Anschluss von Controllern mit einer Baudrate von 110 Baud – 625 kBaud. Die Baudrate für die Kommunikation selbst ist im Modul permanent gespeichert. Die maximale Größe für EA-Daten kann mit dem Scriptbefehl `Get RSOutBufFree16` ausgelesen werden.

Je nach eingespieltem Script des UNIGATE® IC2 führt das Modul selbständig Aktionen aus, um vom angeschlossenen Gerät Daten zu ermitteln. Für Kunden, die bereits eine Software-Adaption für ein UNIGATE® bei der Fa. Deutschmann besitzen, kann auch dieses Protokoll bzw. Script nach einer Anpassung vom IC2 abgearbeitet werden.

In jedem Fall wird das IC2 die Kommunikation mit dem Feldbus eigenständig übernehmen.

### 4.2 Initialisierung der seriellen Schnittstelle

Die Initialisierung der Schnittstelle erfolgt durch Scriptbefehle, wie z.B. "Set Baudrate", "Set Databits", "Set Parity". Für eine detaillierte Beschreibung dieser Befehle siehe die Online-Dokumentation zum Protocol Developer bzw. das Handbuch zum Protocol Developer.

### 4.3 Benutzung der seriellen Schnittstelle

Die serielle Schnittstelle kann frei durch den Benutzer programmiert werden. Es stehen leistungsfähige Script Befehle zum Senden und Empfangen der Daten zur Verfügung; beispielhaft seien nur einige Möglichkeiten genannt, z.B. auf ein Zeichen mit Timeout zu warten, auf eine feste Anzahl Zeichen zu warten oder auch Daten im Modbus RTU Modus zu Senden und zu Empfangen. Eine Referenz zu diesen Befehlen bietet die Online-Dokumentation zum Protocol Developer sowie das Handbuch zum Protocol Developer.

### 4.4 Weitere Betriebsmodi

Die serielle Schnittstelle dient in den Modi Konfigurationsmodus sowie Firmware-Update-Modus auch zum Konfigurieren der Standardsoftware bzw. zum Durchführen eines Firmware Update. Details hierzu finden Sie im Kapitel 11.5.

### 4.5 UART

Kein 9-Bit Betrieb möglich.

## 5 Synchrone serielle Schnittstelle

Die synchrone serielle Schnittstelle des UNIGATE® IC2 dient der Anbindung von getakteten Schieberegistern oder Bauteilen, die über ein *Serial Peripheral Interface (SPI)* verfügen. Sie ermöglicht die Erweiterung des IC2 um digitale Ein- und Ausgänge (z. B. zur Ansteuerung von LEDs oder zum Einlesen von Schalterstellungen), die Kommunikation mit Mikrocontrollern oder die Ansteuerung von DA- oder AD-Wandlern. Anschlussbeispiele sind in Kapitel 3 gegeben. Durch die Verwendung der synchronen seriellen Schnittstelle lassen sich Produkte realisieren, die ohne einen weiteren Mikrocontroller auskommen (Stand-Alone-Betrieb). Beispiele hierfür sind Sensorprodukte oder digitale EA-Module.

### 5.1 Schieberegisterbetrieb

Bevor die Schnittstelle verwendet werden kann, muss sie zunächst durch das Setzen diverser Script-Parameter initialisiert werden (Kapitel 5.1.1).

Die Parameter `ShiftRegisterInputType` und `ShiftRegisterOutputType` erlauben die Verwendung verschiedener Schieberegistertypen, die sich in der Polarität der Schieberegistersignale unterscheiden. Zur Verwendung der Schieberegistertypen 74595 und 74165 können beispielsweise die Werte `RiseClk_RiseLoad` und `RiseClk_LowLoad` gesetzt werden. Die Schieberegisterbreite wird über die Parameter `ShiftRegisterInputBitLength` und `ShiftRegisterOutputBitLength` eingestellt. Die Maximalbreite beträgt je 256 Bit. Der Datenaustausch mit den angeschlossenen Schieberegistern erfolgt mit den Befehlen `WriteShiftRegister`, `ReadShiftRegister` oder bidirektional mit dem Befehl `ShiftRegisterDataExchange`. Die Taktrate bewegt sich zwischen 280 kHz und 320 kHz.

Weiterführende Informationen zu den Befehlen und Parameterwerten finden Sie in der Hilfe der Protocol Developer Software. Auf Wunsch kann die Deutschmann Script-Sprache um zusätzliche Parameterwerte ergänzt werden, um weitere Schieberegistertypen zu unterstützen.

#### 5.1.1 Script-Beispiel

Hinweis: das Script-Beispiel bezieht sich auf das Schaltungsbeispiel in Kapitel 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] );

// Eingangsdaten befinden sich jetzt in InBuffer
// 0x58 liegt am Analogwandler an
// 0x21 an den Ausgängen des Schieberegisters
```



## 5.2 SPI-Betrieb

Zur Nutzung der Schnittstelle im SPI-Betrieb muss diese zunächst initialisiert werden. Mit dem Befehl `InitSPI` werden die Betriebsart, der Modus (Signalpolarität und Phase) und die Taktfrequenz festgelegt.

Der Datenaustausch erfolgt mit dem Befehl `ExchangeSPI`. Die maximale Taktfrequenz beträgt 12 MHz. Details entnehmen Sie bitte der IC2-Pinout-Liste im Download-Bereich unserer Webseite.

Bitte beachten Sie auch die Dokumentation der Script-Befehle in der Hilfe des Protocol Developers.

### 5.2.1 Script-Beispiel

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

## **6 Die Debug Schnittstelle**

### **6.1 Übersicht über die Debug Schnittstelle**

Das UNIGATE® IC2 bietet eine Debug Schnittstelle, die es ermöglicht, ein Script schrittweise auszuführen. Diese Schnittstelle ist im Normalfall nur für die Entwicklung eines Scripts nötig.

### **6.2 Start im Debugmode**

Mit dem Anlegen von Spannung an das UNIGATE® IC2 (Power up) wird die Firmware nach Ausführung eines Selbsttests auf dieser Schnittstelle das binäre Zeichen 0 (0x00) ausgegeben. Wenn das IC2 innerhalb von 500 ms eine Quittung über diese Schnittstelle erhält, befindet es sich im Debugmode. Die Quittung ist das ASCII-Zeichen O (0x4F).

Mit dem Start im Debugmode wird die weitere Ausführung von Scriptbefehlen unterbunden.

### **6.3 Kommunikationsparameter für die Debugschnittstelle**

Die Debugschnittstelle arbeitet immer mit 9600 Baud, no Parity, 8 Datenbit, 1 Stopbit. Eine Änderung dieser Parameter ist im Protocol Developer nicht möglich. Bitte achten Sie darauf, dass diese Einstellungen mit denen der PC-COM-Schnittstelle übereinstimmen und dort die Flusststeuerung (Protokoll) auf „keine“ steht.

### **6.4 Möglichkeiten mit der Debug Schnittstelle**

Normalerweise wird an der Debug Schnittstelle der Protocol Developer angeschlossen. Mit ihm ist es möglich ein Script schrittweise auszuführen, Sprünge und Entscheidungen zu beobachten sowie Speicherbereiche anzusehen. Außerdem können Haltepunkte gesetzt werden, er besitzt also im Prinzip alle Eigenschaften die ein Software-Entwicklungswerkzeug typischerweise besitzt. Es ist jedoch auch möglich über diese Schnittstelle ein Script Update durchzuführen.

### **6.5 Befehle der Debug Schnittstelle**

Die Befehle zum Benutzen der Debug Schnittstelle sind dem Handbuch Protocol Developer zu entnehmen.

## 7 Script und Konfiguration

### 7.1 Übersicht

Das im UNIGATE® IC2 gespeicherte Script und die Konfiguration können über die serielle Schnittstelle (Applikation) im Konfigurationsmodus ersetzt oder upgedated werden.

### 7.2 Der Konfigurationsmodus

Wird der Pin "ConfigMode" während des PowerUp oder Reset auf GND gezogen, startet das UNIGATE® IC2 im Konfigurationsmodus. In diesem Modus besteht die Möglichkeit mit dem IC2 zu kommunizieren, ohne dass die normale Software ausgeführt wird. Man kann in diesem Modus Einstellungen der Standardsoftware des UNIGATE® IC2 verändern oder auch ein neues Script in das UNIGATE® IC2 schreiben. Das UNIGATE® IC2 zeigt seinen Start im Konfigurationsmodus durch die Ausgabe einer Statusmeldung an, die z. B. so aussehen kann:

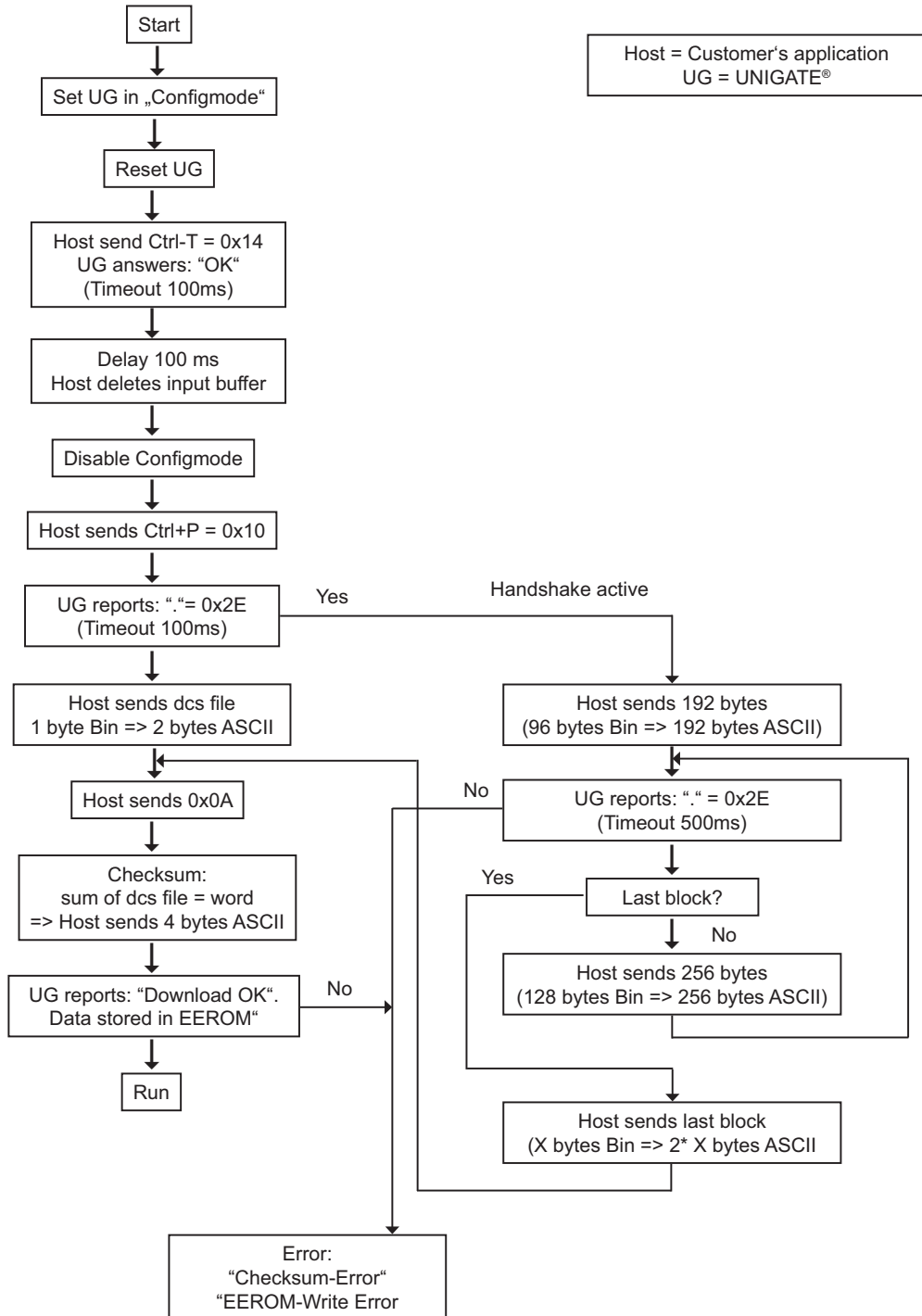
```
IC2FE V1.0.0[44] (c)dA[120MHz] Script(C:190/3968,V:118/16384)="Leer" Author="Deutschmann Automation GmbH" Version="1.0.2" Date=06.11.2019 SN=47110001 IP=0.0.0.0 MAC=00-14-11-01-02-03 Data-Flash=1MByte
```

Beim UNIGATE® IC2 - Fast Ethernet beschränkt sich die Konfiguration auf das Einstellen der Ethernet-Adresse. Siehe auch Kapitel 9.5.

### 7.3 Script updaten

- Die bevorzugte Variante ist die, dass das UNIGATE® IC2 in der von Deutschmann erhältlichen Basisplatine (Developerboard UNIGATE® IC-AB) eingesetzt ist und die Deutschmann Tools (Software WINGATE mit "Write Script" unter "File" oder mit der Software ScriptProgramTool) benutzt werden.
- Sie können das Script auch in Ihrer Applikation von Ihrem Host automatisch ersetzen. Im folgenden Ablaufdiagramm ist der Handshake aufgeführt.

**Script Download**



**Der Ablauf ist wie folgt:**

Das Gateway muss sich im Config-Mode befinden.  
Der Script-Download wird eingeleitet mit "Ctrl-P (=0x10)".  
Danach folgen die Daten byteweise als ASCII-Hex-Zeichen.  
Der Download wird beendet mit einem "LF (=0x0A)".  
Danach folgt noch die Wort-Checksumme als ASCII-Hex-Zeichen.

Das Gateway antwortet mit einer Klartextantwort auf diesen Download und führt eine Warmstart durch.

**Beispiel:**

Es soll folgendes 4 Byte Script heruntergeladen werden: 0x01 0x12 0x5A 0x23  
Die Summe der Bytes ergibt 0x0090 als Checksumme  
Dann muss folgende Sequenz geschickt werden:

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-Antwort: "Download ok"

## 7.4 Konfiguration des UNIGATE® IC2

Das UNIGATE® IC2 wird mit einem Leer Script ausgeliefert.  
Die Konfiguration beschränkt sich beim UNIGATE® IC2 - Ethernet auf die Einstellung der IP-Konfiguration und die Wahl von DHCP.

### 7.4.1 Ethernet

- Protokoll (UDP/IP oder TCP/IP), bis zu 4 Verbindungen gleichzeitig
- Baudrate: 10 MBaud / 100 MBaud autodetect
- IP-Adresse (Gateway)
- Ethernet (MAC-Adresse)
- IP-Zielhost

#### 7.4.2 RS232/RS485/RS422

- RS-Typ: RS232
- Startbit: 1
- Datenbits: 8
- Stopbit: 1
- Parity: Kein
- Baudrate: 9600 Baud

Default-Einstellung. Diese Konfiguration kann über das Script geändert werden.

## 8 Erstellung eines Scripts

### 8.1 Was ist ein Script

Ein Script ist eine Anreihung von Befehlen, die in exakt dieser Reihenfolge ausgeführt werden. Dadurch dass auch Mechanismen gegeben sind, die den Programmfluss im Script kontrollieren, kann man auch komplexere Abläufe aus diesen einfachen Befehlen zusammenbauen.

Das Script ist speicherorientiert. Das bedeutet, dass alle Variablen sich immer auf einen Speicherbereich beziehen. Allerdings brauchen Sie sich beim Entwickeln eines Scripts nicht um die Verwaltung des Speichers zu kümmern; das übernimmt der Protocol Developer für Sie.

### 8.2 Speichereffizienz der Programme

Ein Scriptbefehl kann z. B. eine komplexe Checksumme wie eine CRC-16 Berechnung über Daten ausführen. Für die Codierung dieses Befehls sind als Speicherbedarf (für den Befehl selbst) lediglich 9 Byte nötig. Dies ist nur möglich, indem diese komplexen Befehle in einer Bibliothek enthalten sind.

Ein weiterer Vorteil dieser Bibliothek ist, dass die zu Grunde liegenden Funktionen bereits seit Jahren im praktischen Einsatz sind und deswegen als fehlerarm bezeichnet werden können. Da diese Befehle auch im für den Controller nativen Code vorhanden sind, ist auch das Laufzeitverhalten des Scripts an dieser Stelle günstig.

### 8.3 Was kann man mit einem Script Gerät machen

Unsere Script Geräte sind in der Lage eine Menge von Befehlen abzuarbeiten. Ein Befehl ist dabei immer eine kleine fest umrissenen Aufgabe. Alle Befehle lassen sich in Klassen oder Gruppen einsortieren. Eine Gruppe von Befehlen beschäftigt sich mit der Kommunikation im allgemeinen, die Befehle dieser Gruppe befähigen das Gateway Daten sowohl auf der seriellen Seite als auch auf der Busseite zu senden und zu empfangen.

### 8.4 Unabhängigkeit von Bussen

Prinzipiell sind die Scripte nicht vom Bus abhängig, auf dem sie arbeiten sollen, d. h. ein Script, das auf einem PROFIBUS Gateway entwickelt wurde, wird ohne Änderung auch auf einem Interbus Gateway laufen, da sich diese Busse von der Arbeitsweise sehr stark ähneln. Um dieses Script auch auf einem Ethernet Gateway abzuarbeiten, muss man evtl. noch weitere Einstellungen im Script treffen, damit das Script sinnvoll ausgeführt werden kann.

Es gibt keine festen Regeln, welche Scripte wie richtig arbeiten. Beim Schreiben eines Scripts sollten Sie beachten, auf welcher Zielhardware Sie das Script ausführen wollen, um die nötigen Einstellungen für die jeweiligen Busse zu treffen.

Beim Fast Ethernet Gateway gibt es 4 Kommunikationskanäle.

Für die Kanäle sind jeweils im Script folgende Einstellungen vorzunehmen:

- Die Portnummer (Destination-Port und Source-Port)
- Bei Clientbetrieb TCP oder bei UPD die Destination-IP-Adresse
- Der Mode oder das Protokoll des Kanals:

- 0 = TCP-Server
- 1 = TCP-Client
- 2 = UDP
- 3 = Kanal nicht benutzt (closed)

Ohne diese Einstellungen lässt sich zu dem Gateway über das Ethernet keine Verbindung aufmachen. Nur ICMP-Datenverkehr (PING) ist möglich.

## 8.5 Weitere Einstellungen am IC2 Gateway

Die meisten Geräte benötigen keine weiteren Einstellungen außer denen, die im Script selbst getroffen sind. Allerdings gibt es auch Ausnahmen hierzu. Diese Einstellungen werden mit der Software WINGATE getroffen. Wenn Sie bereits unsere Serie UNIGATE® kennen, wird Ihnen die Vorgehensweise hierbei bereits bekannt sein. Beispielhaft sei hier die Einstellung der IP-Adresse und der Net-Mask eines Ethernet-Gateways genannt. Diese Werte müssen fix bekannt sein und sind auch zur Laufzeit nicht vorhanden. Ein weiterer Grund für die Konfiguration dieser Werte in WINGATE ist folgender: Nach einem Update des Scripts bleiben diese Werte unangetastet, d. h. die einmal getroffenen Einstellungen sind auch nach einer Änderung des Scripts weiterhin vorhanden.

Nur so ist es auch möglich, dass das gleiche Script auf verschiedenen Ethernet-Gateways arbeitet, die alle eine unterschiedliche IP-Adresse haben.

## 8.6 Die Benutzung des Protocol Developer

Das Softwaretool Protocol Developer kann von unserer Internetseite <http://www.deutschmann.de> heruntergeladen werden.

Es ist als Werkzeug zum einfachen Erstellen eines Scripts für unsere Script Gateways gedacht; seine Bedienung ist genau darauf ausgerichtet. Nach dem Start des Programms wird das zuletzt geladene Script erneut geladen, sofern es nicht der erste Start ist.

Windows typisch können Script Befehle per Maus oder Tastatur hinzugefügt werden. Soweit für den entsprechenden Befehl definiert und notwendig wird der Dialog zu dem entsprechenden Befehl angezeigt, und nach dem Eingeben der Werte wird automatisch der richtige Text in das Script eingefügt. Das Einfügen von neuen Befehlen durch den Protocol Developer erfolgt so, dass niemals ein existierender Befehl überschrieben wird. Generell wird ein neuer Befehl vor dem eingefügt, auf dem momentan der Cursor positioniert ist. Selbstverständlich können die Befehle auch einfach per Tastatur geschrieben werden, oder bereits geschriebene Befehle bearbeitet werden.

## 8.7 Genauigkeiten der Baudraten bei UNIGATE® IC2

Die Baudrate der seriellen Schnittstelle wird aus der Quarzfrequenz des Prozessors abgeleitet. Die Abweichung der realen Baudrate zur eingestellten Baudrate ist immer kleiner 1,1%.

## 8.8 Scriptarbeitungszeiten

Das Script wird vom Protocol Developer übersetzt, und der dabei erzeugte Code in das Gateway geladen. Der Prozessor im Gateway interpretiert nun diesen Code. Dabei gibt es Befehle, die sehr schnell abgearbeitet werden können (z. B. "Set Parameter"). Es gibt aber auch Befehle, die länger dauern (z. B. das Kopieren von 1000 Bytes). Somit differiert die Abarbeitungszeit zunächst einmal durch die Art des Scriptbefehls. Wesentlich stärker wird die Abarbeitungszeit der Scriptbefehle aber bestimmt durch die Prozessorzeit, die für diesen Prozess zur Verfügung steht. Da der Prozessor mehrere Aufgaben gleichzeitig ausführen muss (Multitasking-System), steht nur ein Teil der Prozessorleistung für die Scriptabarbeitung zur Verfügung. Folgende Tasks - in der Reihenfolge der Priorität - werden auf dem Prozessor ausgeführt:

- Senden und Empfangen von Daten an der Debug-Schnittstelle (nur wenn Protocol Developer auf PC gestartet ist)
- Senden und Empfangen von Daten an der RS-Schnittstelle
- Senden und Empfangen von Daten an der Feldbus-Schnittstelle
- Durch Systemtakt (1ms) gesteuerte Aufgaben (z. B. Blinken einer LED)
- Abarbeitung des Scriptes



Aus der Praxis heraus kann man ganz grob mit 10 µs pro Scriptzeile rechnen. Dieser Wert hat sich über viele Projekte hinweg immer wieder als Richtwert bestätigt. Er stimmt immer dann recht gut, wenn der Prozessor noch genügend Zeit für die Scriptabarbeitung zur Verfügung hat. An Hand der oben aufgelisteten Tasks kann man folgende Empfehlungen formulieren, um eine möglichst schnelle Scriptabarbeitung zu bekommen:

- Debugschnittstelle deaktivieren (ist im Serieneinsatz der Normalfall)
- Datenlast auf der RS-Schnittstelle möglichst klein halten. Dabei ist nicht die Baudrate das Problem, sondern die Anzahl Zeichen, die pro Sekunde übertragen werden.
- Datenlast auch auf der Feldbusseite nicht unnötig gross machen. Insbesondere bei azyklischen Busdaten, diese möglichst nur bei Änderung schicken. Die Datenlänge bei Bussen die auf eine feste Länge konfiguriert werden (z.B. PROFIBUS) nur so gross wählen, wie unbedingt notwendig.

Sollte trotz dieser Massnahmen die Abarbeitungszeit zu gross sein, besteht die Möglichkeit, einen kundenspezifischen Scriptbefehl zu generieren, der dann mehrere Aufgaben in einem Scriptbefehl abarbeitet. Wenden Sie sich dazu bitte an unsere Support-Abteilung.

## 9 Ethernet

Jeder Ethernet-Clientrechner mit TCP/IP Netzwerkprotokoll in einem 10/100 Mbaud Ethernet kann Daten mit dem Gateway austauschen.

### 9.1 Netzwerkadressen, Netzwerkverbindungen

Nach dem Einschalten wartet das Gateway auf Datenverkehr mit einem Netzwerkteilnehmer. Das Gateway ist dabei immer bereit auf ICMP-Messages zu reagieren (z. B. **PING** Request und Reply). Ebenfalls sind die Funktionen des ARP-Protokolls verfügbar.

### 9.2 TCP-Verbindungsaufbau

In der typischen Anwendung im TCP/IP Protokoll geht das Gateway nach dem Einschalten in den Zustand "Passive Open" unter einer in der Konfiguration festgelegten IP-Adresse und Portnummer. Ein Client-Rechner kann eine TCP/IP Verbindung mit dem Gateway unter dem in der Konfiguration festgelegten Port öffnen. Nach erfolgreichem Verbindungsaufbau können dann Daten ausgetauscht werden.

#### 9.2.1 TCP-Server-Kanalmodus

Eine TCP-Verbindung, bei der der IC2-FE der Server ist, wird nach 65635 Sekunden (18,2 Stunden) geschlossen, wenn in dieser Zeit keine Daten übertragen wurden.

### 9.3 Datenaustausch

Bei dem Protokoll TCP/IP steht nach dem Aufbau einer Verbindung zwischen dem Gateway und dem angeschlossenen Ethernet-Client eine Vollduplex Datenverbindung zur Verfügung. Der Datenstrom wird in beiden Richtungen durch das TCP-Protokoll abgesichert.

Bei Betrieb mit UDP/IP-Protokoll wird keine Verbindung vor dem Übertragen von Daten aufgebaut. Hier ist Anwendung für die Steuerung des Datenflusses, sowie für die Sicherung der Daten verantwortlich.

#### 9.3.1 TCP oder UDP?

TCP ist ein verbindungsorientiertes Protokoll: während der Datenübertragung gibt es hier eine feste logische Verbindung zwischen Client und Server. TCP verfügt über alle Mechanismen, die notwendig sind, um eine Verbindung zu öffnen, einen fehlerfreien Datentransfer über das Netzwerk sicherzustellen und die Verbindung wieder zu schließen. Zu diesem Zweck werden von der Protokollsoftware für jede Verbindung eigene Puffer angelegt und verwaltet. Sollen sehr viele UNIGATE®-Server gleichzeitig von einem Rechner aus angesprochen werden, können im Extremfall Speichergrenzen in dem Rechner erreicht werden.

Im Gegensatz dazu verfügt UDP nicht über Mechanismen, um Datenpakete zu wiederholen oder den Empfang der Daten in der richtigen Reihenfolge zu gewährleisten. Die Datenübertragung mit UDP ist empfehlenswert, wenn ein übergeordnetes Protokoll zwischen dem Endgerät am UNIGATE®-Server und der Applikation auf der TCP/IP-Station ohnehin eine fehlerfreie Übertragung sicherstellt.

### 9.4 Datenaustauschmodus (data exchange mode)

Das Gateway muss sich in diesem Modus befinden, damit ein Datenaustausch zwischen der RS-Seite des Gateways und dem Feldbus möglich ist. Dieser Modus ist immer dann aktiv, wenn das Gateway sich nicht im Konfigurations- oder Testmodus befindet. Im Datenaustauschmodus wird das Gateway das eingestellte Protokoll mit den durch WINGATE voreingestellten Parametern ausführen.

### 9.4.1 Direkter Datenaustausch

Die Daten von der RS-Schnittstelle werden ohne Veränderung über eine TCP/IP Verbindung im Ethernet abgebildet.

### 9.4.2 MODBUS-TCP

Das Modbus-Protokoll ist als Script für die serielle Schnittstelle für alle Script-Gateways bereits verfügbar. Bei dem Fast-Ethernet-Gateway ist nun die zusätzliche Funktion einstellbar, dass die Modbus-Daten als TCP-Modbus-Datenpakete auf der Ethernet-Seite zur Verfügung gestellt werden können.

(MODBUS-TCP)

Dazu ist der für diese Eigenschaft verwendete Kanal im Script als MODBUS-TCP-Kanal einzurichten. Für diese Funktion ist ein laufendes Modbus-Script erforderlich.

### 9.4.3 Server

#### 9.4.3.1 Webserver

Es sind Webseiten über einen im Script als Webserverkanal eingerichteten Kommunikationskanal abrufbar. Dabei wird das interne Dateisystem genutzt. Es können mehrere Webseiten platziert werden, bzw. Webseiten, Bilddateien und Java-Applets abgerufen werden.

Vom Browser werden für das Laden des Applets und der Bilder zusätzliche Verbindungen geöffnet.

Zusätzlich muss gewährleistet sein, dass im Dateisystem Dateien vorhanden sind, die als Webseite abrufbar sind.

Der Webserver läuft unabhängig vom Script.

Der Webserver ist nur im Run-Mode aktiv. Nicht im Config.-Mode. Der Webserver kann per Script über den Parameter WebServer aktiviert (Wert: 1) und wieder deaktiviert (Wert: 0) werden.

Der Webserver belegt keinen der 4 möglichen Kanäle mehr, wenn sie aktiviert sind.

#### 9.4.3.2 FTP-Server

Der FTP-Server ist sowohl im Config.-Mode, als auch im Run Mode aktiv. Per Script kann über den Parameter FTPServer aktiviert (Wert: 1) und wieder deaktiviert (Wert: 0) werden.

Der FTP-Server ist passwortgeschützt. Mit dem vordefinierten Benutzernamen „deutschmann“ und dem Passwort „deutschmann“ kann eine Verbindung hergestellt werden

Der FTP-Server belegt keinen der 4 möglichen Kanäle mehr, wenn sie aktiviert sind.

Der FTP-Server bietet keine gesonderte Unterstützung für Leerzeichen in Dateinamen. Es ist aber möglich, Dateien mit Leerzeichen im Namen auf den/vom FTP-Server zu übertragen, wenn die Leerzeichen vom FTP-Client korrekt behandelt werden. Das ist aber immer abhängig vom verwendeten Programm.

Umlaute sollten in Dateinamen nicht verwendet werden. Eine Verbindung zum FTP-Server kann z.B. mit Kommandozeilen-Clients (MSYS64, Windows 10) durchgeführt werden. Nachfolgend eine Liste mit Server-Befehlen:

### Supported FTP Commands

Code	Command	Description
USER	User Name	Starts login with name identifying the user.
PASS	Password	Continues login with the user's password.
QUIT	Logout	Closes the user connection.
SYST	System	Identifies the operating system at the server.
NOOP	No Operation	Sends an OK reply.
XNOP	No Operation	Same as NOOP
PWD	Print Working Directory	Returns the name of the current working directory.
XPWD	Print Working Directory	Same as PWD
CWD	Change Working Directory	Changes the current working directory of the user.
XCWD	Change Working Directory	Same as CWD
CDUP	Change Directory Up	Changes working directory to the parent of the current directory.
XCUP	Change Directory Up	Same as CDUP
MKD	Make Directory	Creates a sub directory in the current working directory.
XMKD	Make Directory	Same as MKD
RMD	Remove Directory	Removes the directory.
XRMD	Remove Directory	Same as RMD
Type	Representation Type	Supports ASCII and Image types.
PORT	Data Port	Specifies the data port to be used in data connection.
PASV	Passive	Requests the server to listen on a data port and wait for a connection.
LIST	List	Sends a directory listing to the user.
NLST	Name List	Sends a directory listing to the user.
RETR	Retrieve	Sends a file content to the user.
STOR	Store	Saves a captured user file on server.
APPE	Append	Appends a captured user file to an existing file on server.
DELE	Delete	Deletes a specified file from server.
RNFR	Rename From	Specifies the name of existing file to rename (must be followed by RNTO).
RNTO	Rename To	Renames an existing file to new name.
HELP	Help	Returns a list of supported commands.
SIZE	Size	Returns the size of a specified file.
MDTM	Last-modified Time	Returns last-modified time of a specified file.

#### Verfügbare Funktionen:

- PUT = Datei anlegen und schreiben
- GET = Datei lesen
- DIR = Directory lesen
- DEL = Datei löschen
- DEL \*.\* = Löschen aller Dateien und initialisieren (Formatieren) des Dateisystems.



#### Warnung:

Es werden auch die Script- und Config-Datei gelöscht.

Unterverzeichnisse sind nicht im Dateisystem vorgesehen. Es kann daher auch nicht auf Unterverzeichnisse umgeschaltet werden bzw. es können keine Unterverzeichnisse angelegt werden. Der FTP-Server läuft unabhängig vom Script. Der FTP-Server ist im Konfig-Mode automatisch verfügbar.

#### **9.4.3.3 Script Update via FTP**

Es muss das vom Protocol Developer erzeugte dcs-File als "script.dcs" per FTP auf dem Gateway gespeichert werden. Das Gateway erkennt dieses File beim Hochfahren, übernimmt es und löscht dann das File "script.dcs".

#### **9.4.3.4 Systemkonfiguration Update via FTP**

Es muss ein WINGATE gwc-File per FTP als "SYSTEM.GWC" auf dem Gateway gespeichert werden. Das Gateway erkennt dann beim Start dieses File, übernimmt die neue Konfiguration und löscht dann das File "SYSTEM.GWC".

#### **9.4.3.5 Dateisystem**

Es ist ein Dateisystem verfügbar. Auf dem Gateway wird ein zusätzlicher Speicherbaustein bestückt, in dem die Dateien des Dateisystems permanent gespeichert werden. Die Dateinamen dürfen bis zu 31 Zeichen lang sein. Die maximale Speicherkapazität beträgt derzeit 1 MByte. Auf Anfrage können bis zu 8 MByte bestückt werden.

#### **9.4.3.6 TCP-Server-Kanalmodus**

Eine TCP-Verbindung, bei der der IC2-FE der Server ist, wird nach 65635 Sekunden (18,2 Stunden) geschlossen, wenn in dieser Zeit keine Daten übertragen wurden.

#### **9.4.3.7 UDP-Kanalmodus**

Beim FEX war es möglich, einen Kanal mit Ziel-IP 0 und Port 0 zu initialisieren und so über diesen Socket UDP-Pakete von allen IPs und allen Ports zu empfangen. Das geht nicht mehr.

#### **9.4.3.8 Raw-Kanalmodus**

Im IC2-FE kann der Kanal 0 nur dann in den Raw-Modus versetzt werden, wenn die Sendung nicht ARP oder IP (v4 oder v6) ist. Die Kanalnummer ist egal, aber es kann weiter nur einen solchen Kanal geben.

#### **9.4.3.9 BusStatus**

Beim IC2-FE werden für die Funktionen Unigate-Scan, FTP- und Webserver keine Kanäle belegt. Dadurch ist ReadBusStatus nach BusStart 0x20222222, solange nicht durch das Skript weitere Kanäle initialisiert werden.

## **9.5 Einstellen der Ethernet-Adresse**

Festes Einstellen der Adresse über die Konfiguration

Das UNIGATE® IC2 muss sich im Konfigurationsmodus befinden (siehe Kapitel 7.2 "Der Konfigurationsmodus"). Jetzt ist es mit WINGATE möglich, die Adresse einzustellen. Diese Adresse bleibt so lange erhalten, bis sie erneut geändert wird.

### Scriptbeispiel für die Initialisierung des Ethernet

```
var wRemotePort: word; Moveconst(wRemotePort, 2001);
var wLocalPort: word; Moveconst(wLocalPort, 2000);
var lRemoteIP: long; Moveconst(lRemoteIP, 0xc0a82f0b); // 192.168.47.11

var lE3: long; Moveconst(lE3, 0xE3222222); // Kanal 0 verbunden, Rest
unbenutzt
var w0: word; Moveconst(w0, 0);

var bChannel: byte; Moveconst(bChannel, 0);
var lBusState: long;
var wInCount: word;
var wOutCount: word;
var aRecvData: buffer[100];
var aSendData: buffer[100];

// Kanal 0 als TCP-Client initialisieren
InitFastEthernet(bChannel, TCPIP_Client, wRemotePort, wLocalPort, lRe-
moteIP);

BusStart; // Ethernet aktivieren

:waitForConnection; // TCP-Verbindung prüfen, notfalls warten (bei UDP
unnötig)
Get(ReadBusState, lBusState);
if lBusState equal lE3 then :connected;
jump :waitForConnection;

:connected; // bei Verbindung auf Empfang prüfen
Get(AvailableBusData, wInCount);

if wInCount equal w0 then :send; // bei Empfang lesen und Auswerten
ReadBus(aRecvData[0], wInCount);
{hier Code ergänzen, um den Empfang auszuwerten}

:send; // etwas senden
Moveconst(wOutCount, 13);
Moveconst(aSendData[0], "Sendedaten...");
WriteBus(aSendData[0], wOutCount);

jump :waitForConnection; // typischerweise laufen Skripte als
Endlosschleife
```

## 10 Fehlerbehandlung beim UNIGATE® IC2

Es können zwei Kategorien der System-Errors unterschieden werden:

Schwere Fehler (1-4): In diesem Fall muss das Gateway aus- und wieder neu eingeschaltet werden. Tritt der Fehler erneut auf, muss das Gateway getauscht und zur Reparatur eingeschickt werden.

Warnungen (6-15): Diese Warnungen werden lediglich zur Information 1 Minute lang angezeigt und werden dann automatisch zurückgesetzt. Treten diese Warnungen häufiger auf, ist der Kundendienst zu verständigen.

Der System-Error kann über das Script ausgelesen werden.

Im Konfigurationsmodus sind diese Anzeigen nicht gültig und nur für interne Zwecke bestimmt.

Fehlernummer	Fehlerbeschreibung
0	Reserviert
1	Hardwarefehler
2	EEROM-Fehler
3	Interner Speicherfehler
4	Feldbus Hardwarefehler
5	Script-Fehler
6	Reserviert
7	RS Sende-Puffer-Überlauf
8	RS Empfangs-Puffer-Überlauf
9	RS Timeout
10	Allgemeiner Feldbusfehler
11	Parity- oder Frame-Check-Fehler
12	Reserviert
13	Feldbus Konfigurationsfehler
14	Feldbus Datenpuffer-Überlauf
15	Reserviert

Tabelle 1: Fehlerbehandlung beim UNIGATE® IC2

## 11 Firmware Update

### 11.1 Übersicht

Im Firmware-Update-Modus kann über die serielle Schnittstelle des UNIGATE® IC2 die Firmware ersetzt werden.

### 11.2 Einstellen des Firmware-Update-Modus

#### 11.2.1 Einstellung per Hardware

Das UNIGATE® IC2 kann per Hardware in den Firmware-Update-Modus gebracht werden. Hierzu muss während des Power-Up-Vorgangs das Signal  $\overline{BE}$  (-Boot enable) auf das Potential GND gezogen werden.

#### 11.2.2 Einstellung per Software

Befindet sich das UNIGATE® IC2 im Konfigurationsmodus (siehe Kapitel 7.2) kann es über das Kommando CTRL-F (0x06) interaktiv in den Firmware-Update-Mode gebracht werden. Nach dem Senden des Kommandos erfolgt eine Sicherheitsabfrage, die mit J oder N (Ja oder Nein) beantwortet werden muss. Nach positiver Bestätigung erfolgt der Neustart des IC2 im Firmware-Update-Mode.

### 11.3 Durchführung des Firmware-Updates

Der sicherste Weg für den Firmware-Update ist die Benutzung der Basisplatine in Kombination mit der Software "FDT.EXE" (Firmware-Download-Tool). Diese Hilfsmittel sind von Deutschmann zu erhalten (siehe Kapitel 13).

### 11.4 Sicherheitshinweis

Das Firmware-Update sollte nur dann durchgeführt werden, wenn keine andere Möglichkeit zur Verfügung steht. Ein begonnener Firmware-Update-Vorgang kann NICHT rückgängig gemacht werden. Die vorige Firmware ist damit endgültig unbrauchbar.

### 11.5 Betriebsmodi des IC2

#### Standard-Betriebsmodus

Dieser Modus wird für die normale Nutzung des IC2 benötigt. In diesem Modi wird das IC2 alle Script Befehle abarbeiten und normal die entsprechenden Benutzerdaten austauschen; auch der Bus wird in diesem Modus durch das IC2 bedient.

#### Konfigurationsmodus

Im Konfigurationsmodus wird das UNIGATE® IC2 nach dem Start (oder nach einem Reset) einen Selbsttest ausführen und nach erfolgreichem Selbsttest auf weitere Anweisungen warten. Hier ist es möglich ein übersetztes Script in das Gerät zu laden oder auch den Firmware Download Mode zu initiieren.

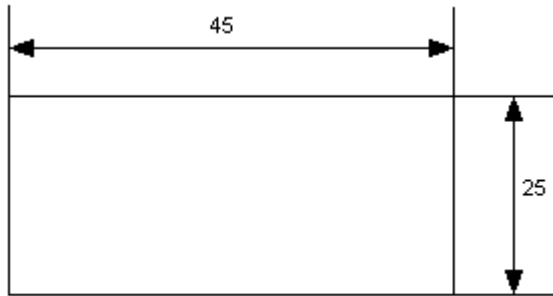


## 12 Technische Daten

Hier finden Sie alle nötigen technischen Daten zum UNIGATE<sup>®</sup> IC2.  
Alle Maßangaben in mm.

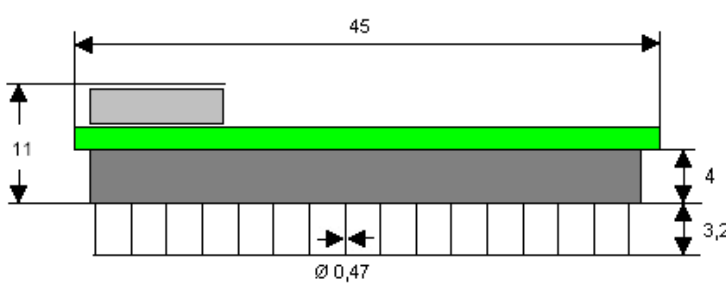
### 12.1 Mechanik des UNIGATE<sup>®</sup> IC2

#### 12.1.1 UNIGATE<sup>®</sup> IC2 allgemeine Abmessungen

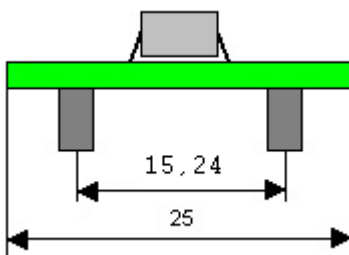


Max. Höhe = 20 mm; inkl. Stift

#### 12.1.2 Abmessungen UNIGATE<sup>®</sup> IC2 (nur Fast Ethernet)



Die Stifte des UNIGATE<sup>®</sup> IC2 - Ethernet sind im Rastermaß 2,54 mm angeordnet.



DIP-Spacing Code 6

Wenn Sie auch an den Einsatz von unseren anderen Feldbus IC2 denken, müssen Sie eine max. Bauhöhe (inkl. Stifte) von  $\leq 20$  mm berücksichtigen.

## 12.2 Technische Daten UNIGATE® IC2 - Fast Ethernet

<b>Merkmale</b>	<b>Erläuterungen</b>
Betriebsspannung	3,3 V $\pm$ 5 %, 200 mA (typ.), 210 mA (max.)
Schnittstelle	2 UART Schnittstellen, 1 synchron serielle Schnittstelle
Galvanische Trennung Ethernetseite	Standard
TCP/IP-Adresse	Einstellbar
Baudrate	10/100 MBaud
UART-Baudrate	Bis 7,5 MBaud (einstellbar über Script)
Protokolle	UDP, TCP, Modbus TCP, HTTP, FTP
Sonstiges	Extern können z. B. I/Os, analoge Signale, Schieberegister, LEDs, Schalter etc. angeschlossen werden
Abmessungen	45 x 25 x 11 mm (BxTxH)
Montage	32 DIL
Gewicht	ca. 9 g
Betriebstemperatur	-40°C..+85°C
Lager-/Transporttemperatur	-40°C..+100°C
Einbaulage	Beliebig
Technologie	Mikroprozessor

## 13 Zubehör

Von Deutschmann Automation sind folgende Hilfsmittel zu erhalten.

### 13.1 Adapter RS232

Der Adapter RS232 bietet die Möglichkeit in einer Applikation einen bestehenden Treiber MAX 232 (nur im DIL-16-Gehäuse) durch diesen Adapter zu ersetzen. Diese Platine ermöglicht den Einsatz des IC2 nach Kapitel 3.4, auf Seite 15. Zu beachten ist, dass der Ethernet hierdurch keinen normkonformen Anschluss bietet. Mit einem Zwischenstecker ist aber der Betrieb des Ethernet zumindest möglich.

Diese Hardware ist nur für Entwicklungszwecke gedacht. Sie bietet die Möglichkeit innerhalb kürzester Zeit eine bestehende Applikation busfähig zu machen und die Einsatzfähigkeit und Funktionalität des IC2 zu testen.

### 13.2 Adapter RS485

Der RS485 Adapter ist von der Funktionalität zu sehen wie der RS232 Adapter. Er bietet die Möglichkeit einen Baustein LS 176 (nur im DIL 8 Gehäuse) durch das IC2 zu ersetzen.

Es bestehen alle Einschränkungen, die auch bei dem RS232 Adapter bestehen.

### 13.3 FirmwareDownloadTool (FDT)

Das FirmwareDownloadTool steht auf der Homepage von Deutschmann Automation zum Download bereit. Es wird benötigt um ein Update der Firmware durchzuführen. Voraussetzung hierfür ist, dass die Software an die serielle Schnittstelle des IC2 herangeführt werden kann. Die Vorgehensweise für das Update selbst ist in der Software beschrieben.

### 13.4 Protocol Developer

Der Protocol Developer ist die Entwicklungsumgebung für Scripte, die auch den Debugger enthält. In diesem Software Paket ist auch die Dokumentation zu allen Scriptbefehlen enthalten. Auf der Homepage von Deutschmann Automation <http://www.deutschmann.de> liegt diese Software zum Download bereit. Das im PDF-Format verfügbare Handbuch Protocol Developer gibt weiterführende Hinweise zur Benutzung der Software.

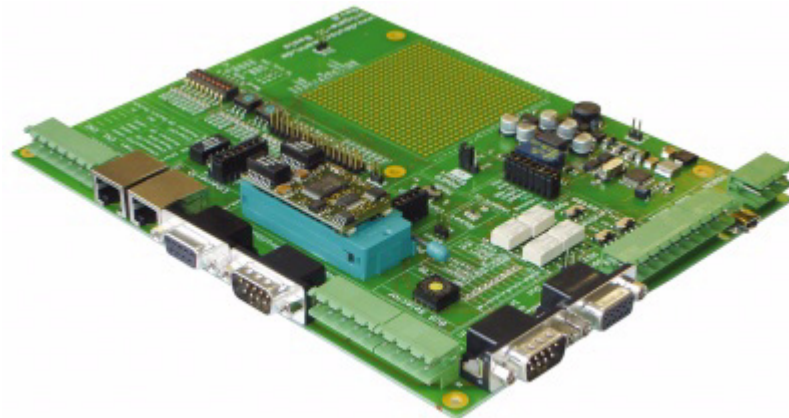
### 13.5 Developerkit UNIGATE® IC-AB

**Das Developerkit beinhaltet**

- ein Developerboard UNIGATE® IC-AB (siehe Kapitel 13.5.1)
- ein Steckernetzteil zur Versorgung des Developerboards
- sowie Anschlusskabel für Appl. RS232, Debug RS232 und Appl. RS422/485
- USB-Kabel
- Software und Dokumentation ergänzen das Paket.

#### 13.5.1 Developerboard UNIGATE® IC-AB

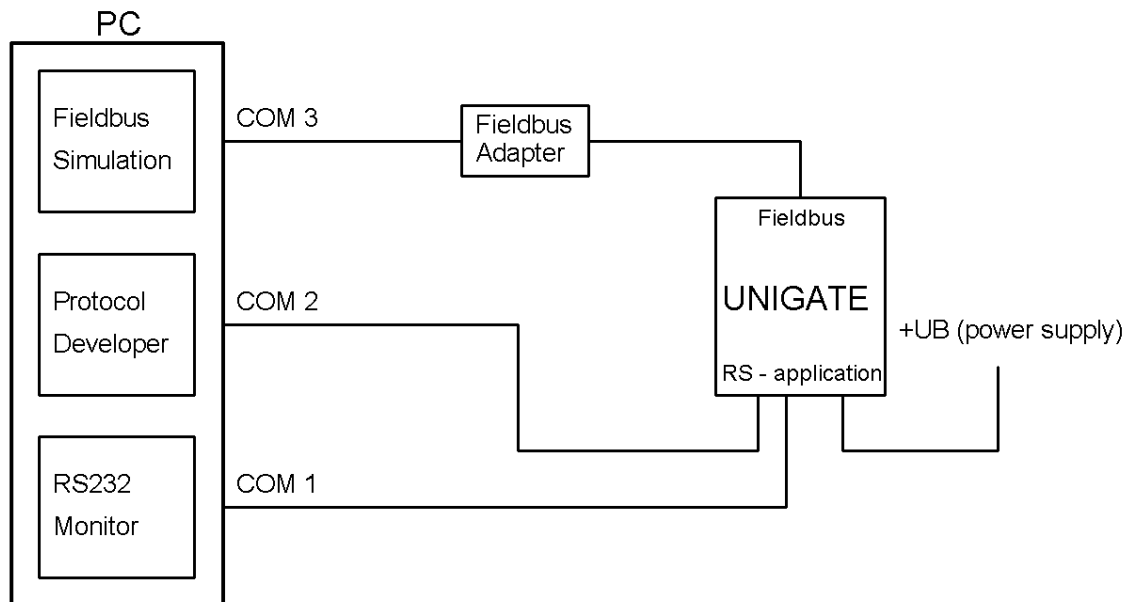
Das Developerboard wurde entwickelt, um die schnelle Implementierung des Deutschmann All-In-One Bus Node UNIGATE® IC und UNIGATE® IC2 in Ihre Elektronik zu gewährleisten. Das Board ist für alle von Deutschmann unterstützten Feldbusse und Industrial Ethernet Busse geeignet.



Das oder die benötigten UNIGATE® IC2s werden separat bestellt. Die je nach Ausführung benötigte Spannung (5 Volt oder 3,3 Volt) ist einstellbar. Für die Anbindung an den PC (DEBUG-Schnittstelle) steht sowohl eine RS232-Schnittstelle wie auch ein USB-Anschluss zur Auswahl.

Die Applikationsanbindung erfolgt wahlweise über USB, RS232, RS485 oder die RS422. Zum Testen der jeweiligen Busseite stehen die Busanschlüsse gemäß Norm oder Marktstandard zur Verfügung. Die Deutschmann Add-On-Pakete (Busmaster-Simulation) sind optional erhältlich. Das Board enthält 32 Bit Input und 16 Bit Output, jeweils mit einer LED versehen. Verschiedene Abgriffe erlauben eine einfache Ankopplung an Ihren Prozessor. Ein Lochrasterfeld mit den wichtigsten Signalen (Spannung, IOs) erlaubt eine kundenspezifische Hardwareerweiterung (z. B. um einen D/A Wandler) anzubinden.

**13.5.2 Schnellstart**



Für einen transparenten Datenaustausch finden Sie im Protocol Developer unter "File->New" Beispiel-Scripte für den jeweiligen Feldbus.

## 14 Anhang

### 14.1 Erläuterungen der Abkürzungen

#### Allgemein

CL	=	Produktgruppe CL (Compact Line)
CM	=	Produktgruppe CM (CANopen Line)
CX	=	Produktgruppe CX
EL	=	Produktgruppe EL (Ethernet Line)
FC	=	Produktgruppe FC (Fast Connect)
GT	=	Galvanische Trennung RS-Seite
GY	=	Gehäusefarbe grau
MB	=	Produktgruppe MB
RS	=	Produktgruppe RS
SC	=	Produktgruppe SC (Script)
232/485	=	Schnittstelle RS232 und RS485 umschaltbar
232/422	=	Schnittstelle RS232 und RS422 umschaltbar
DB	=	zusätzlich eine RS232 DEBUG-Schnittstelle
D9	=	Anschluss der RS über 9pol. D-SUB statt 5pol. Schraub-Steckverbinder
PL	=	Nur Platine ohne DIN-Schienenmodul und ohne Gehäusedeckel
PD	=	Nur Platine ohne DIN-Schienenmodul mit Gehäusedeckel
AG	=	Gateway montiert im Aludruckgussgehäuse
EG	=	Gateway montiert im Edelstahlgehäuse
IC2	=	Produktgruppe IC2 (IC2-Bauform DIL32)
IC	=	Produktgruppe IC (IC-Bauform DIL32)
IO8	=	Option I/O8
16	=	Scriptspeicher auf 16KB erweitert
5V	=	Betriebsspannung 5V
3,3V	=	Betriebsspannung 3,3V

#### Feldbus

CO	=	CANopen
C4	=	CANopen V4
C4X	=	CANopen V4-Variante X (siehe Vergleichstabelle UNIGATE® IC2 beim jeweiligen Produkt)
DN	=	DeviceNet
EC	=	EtherCAT
EI	=	EtherNet/IP
FE	=	Ethernet 10/100 MBit
FEX	=	Ethernet 10/100 MBit-Variante X (siehe Vergleichstabelle UNIGATE® IC2 beim jeweiligen Produkt)
IB	=	Interbus
IBL	=	Interbus
LN62	=	LONWorks62
LN512	=	LONWorks512
ModTCP	=	ModbusTCP
MPI	=	Siemens MPI®
PL	=	Powerlink
PN	=	Profinet-IO
PBDP	=	ProfibusDP
PBDPL	=	ProfibusDP-Variante L (siehe Vergleichstabelle UNIGATE® IC2 beim jeweiligen Produkt)

		Produkt)
PBDPX	=	ProfibusDP-Variante X (siehe Vergleichstabelle UNIGATE® IC2 beim jeweiligen Produkt)
PBDPV0	=	ProfibusDPV0
PBDPV1	=	ProfibusDPV1
RS	=	Serial RS232/485/422

## 15 Service

Sollten einmal Fragen auftreten, die in diesem Handbuch nicht beschrieben sind, finden Sie im

- FAQ/Wiki Bereich unserer Homepage [www.deutschmann.de](http://www.deutschmann.de) oder [www.wiki.deutschmann.de](http://www.wiki.deutschmann.de) weiterführende Informationen.

Falls dennoch Fragen unbeantwortet bleiben sollten wenden Sie sich direkt an uns.

Bitte halten Sie für Ihren Anruf folgende Angaben bereit:

- Gerätebezeichnung
- Seriennummer (S/N)
- Artikel-Nummer
- Fehlernummer und Fehlerbeschreibung

Ihre Anfragen werden im Support Center aufgenommen und schnellstmöglich von unserem Support Team bearbeitet. (In der Regel innerhalb 1 Arbeitstag, selten länger als 3 Arbeitstage.)

Der technische Support ist erreichbar von Montag bis Donnerstag von 8.00 bis 12.00 und von 13.00 bis 16.00, Freitag von 8.00 bis 12.00 (MEZ).

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

Zentrale und Verkauf           +49 6434 9433-0  
Technischer Support           +49 6434 9433-33

Fax Verkauf                       +49 6434 9433-40  
Fax Technischer Support       +49 6434 9433-44

Email Technischer Support   support@deutschmann.de

### 15.1 Einsendung eines Gerätes

Bei der Einsendung eines Gerätes benötigen wir eine möglichst umfassende Fehlerbeschreibung. Insbesondere benötigen wir die nachfolgenden Angaben:

- Welche Fehlernummer wurde angezeigt
- Wie groß ist die Versorgungsspannung ( $\pm 0,5V$ ) mit angeschlossenem Gateway
- Was waren die letzten Aktivitäten am Gerät (Programmierung, Fehler beim Einschalten, ...)

Je genauer Ihre Angaben und Fehlerbeschreibung, umso exakter können wir die möglichen Ursachen prüfen.

### 15.2 Download von PC-Software

Von unserem Internet-Server können Sie kostenlos aktuelle Informationen und Software laden.

<http://www.deutschmann.de>

