

# **EXDUL-593E**

**EDV-Nr.: A-372320**

# **EXDUL-593S**

**EDV-Nr.: A-372310**

1 Eingang bipolar über Optokoppler  
1 Ausgang über Optokoppler1A  
1 Zähler 32 Bit  
Programmierbare Logik  
TCP/IP- und Webserver  
LCD-Anzeige (nur EXDUL-593E)

**wasco<sup>®</sup>**

Handbuch

## Copyright® 2021 by Messcomp Datentechnik GmbH

Diese Dokumentation ist urheberrechtlich geschützt. Alle Rechte sind vorbehalten.

Messcomp Datentechnik GmbH behält sich das Recht vor, die in dieser Dokumentation beschriebenen Produkte jederzeit und ohne Vorankündigung zu verändern.

Ohne schriftliche Genehmigung der Firma Messcomp Datentechnik GmbH darf diese Dokumentation in keinerlei Form vervielfältigt werden.

## Geschützte Warenzeichen

Windows®, Visual Basic®, Visual C++®, Visual C#® sind eingetragene Warenzeichen von Microsoft.

**wasco®** ist ein eingetragenes Warenzeichen.

**EXDUL®** ist ein eingetragenes Warenzeichen.

LabVIEW® ist ein eingetragenes Warenzeichen.

Bei anderen genannten Produkt- und Firmennamen kann es sich um Warenzeichen der jeweiligen Inhaber handeln.

## Haftungsbeschränkung

Die Firma Messcomp Datentechnik GmbH haftet für keinerlei durch den Gebrauch des Multifunktionsmoduls EXDUL-593 und dieser Dokumentation direkt oder indirekt entstandenen Schäden.

## Wichtiger Hinweis:

Dieses Handbuch wurde für die Module EXDUL-593E und EXDUL-593S erstellt. Das EXDUL-593E bietet zusätzlich eine LCD-Anzeige, alle weiteren Funktionen der Module sind identisch. Für das EXDUL-593S sind die Befehle und Funktionen, die das Display betreffen, nicht zutreffend.

# Inhaltsverzeichnis

<b>1. Produktbeschreibung .....</b>	<b>5</b>
<b>2. Anschlussklemmen.....</b>	<b>6</b>
2.1 Klemmenbelegung von CN1.....	6
<b>3. Systemkomponenten .....</b>	<b>7</b>
3.1 Blockschaltbild EXDUL-593E .....	7
3.2 Blockschaltbild EXDUL-593S .....	8
3.3 Digitaler Eingang über Optokoppler .....	9
3.4 Digitaler Ausgang über Optokoppler.....	9
3.5 Zähler .....	9
3.6 6 Temperaturmesseinheiten .....	9
3.7 Programmierbare Logik .....	9
3.8 TCP/IP-Server .....	9
3.9 Webserver .....	10
3.10 Kommunikations-Watchdog-Timer .....	10
3.11 LCD Anzeige (nur EXDUL-593E).....	10
<b>4. Inbetriebnahme .....</b>	<b>11</b>
4.1 Anschluss an einen Ethernet-Port .....	11
4.2 Anschluss der Betriebsspannung .....	11
4.3 Integrierte Webpage des EXDUL-593E / EXDUL-593S .....	12
4.4 Passwortschutz - Zugangskennung .....	12
4.5 Grundeinstellung Netzwerk-Konfiguration.....	13
4.6 Zusammensetzung und Aufbau der IP-Adresse .....	13
4.7 Änderung der Netzwerk-Konfiguration .....	14
4.8 Konfiguration mit statischer IP-Adresse (DHCP deaktiviert) .....	15
4.9 Konfiguration mit dynamischer IP-Adresse (DHCP aktiviert) .....	17
4.10 LCD-Anzeige während des Bootvorgangs (nur EXDUL-593E) .....	18
4.11 LCD-Anzeige während des Betriebs (nur EXDUL-593E) .....	19
<b>5. Zugriff auf das EXDUL-593 .....</b>	<b>20</b>
5.1 Zugriff über die EXDUL Webpage .....	20
5.2 Zugriff über TCP/IP-Sockets .....	21
5.3 Host-Namen, IP-Adresse und MAC-Adresse feststellen .....	22
<b>6. Temperaturmesseinheiten .....</b>	<b>23</b>
6.1 Beschaltung.....	23
6.2 Messmöglichkeiten .....	24
6.3 Fehlererkennung .....	24
6.4 Sensorart konfigurieren .....	26
6.5 Abgleich bzw. Kalibrierung der Messeinheiten .....	27

<b>7. 1 Optokopplereingang</b> .....	28
7.1 Pinbelegung der Eingangsoptokoppler.....	28
7.2 Eingangsbeschaltung .....	29
7.3 Eingangsstrom.....	29
<b>8. 1 Optokopplerausgang</b> .....	30
8.1 Pinbelegung des Ausgangsoptokopplers .....	30
8.2 Ausgangsbeschaltung .....	30
8.3 Ausgangsdaten.....	30
8.4 Programmierung des Optokopplerausgangs.....	31
8.5 Optokopplerausgang rücklesen.....	31
<b>9. Zähler</b> .....	32
<b>10. Programmierbare Logik</b> .....	33
10.1 Logikzweig.....	34
10.2 Logikeingänge .....	35
10.3 Logikverknüpfung .....	36
10.4 Logikausgänge .....	36
10.5 Message an PC schicken .....	37
10.6 Timing der programmierbaren Logik.....	38
<b>11. Watchdog-Timer (WDT)</b> .....	39
<b>12. Error-Register</b> .....	40
<b>13. Informations-, LCD- und Userregister</b> .....	41
13.1 Register HW-Kennung und Seriennummer .....	41
13.2 Speicherbereiche UserA, UserB, UserLCD1m* und UserLCD2m* .....	42
13.3 Display-Register UserLCD-Zeile1*, UserLCD-Zeile2* und LCD-Kontrast* .....	42
<b>14. Installation des Treibers</b> .....	43
<b>15. Programmierung</b> .....	44
15.1 Einführung .....	44
15.2 Programmierarten.....	44
15.3 Programmierung unter Windows mit der .NET EXDUL.dll Library .....	44
15.4 Programmierung mit TCP-Libraries.....	52
15.5 Receivermodus.....	91
15.6 Modulzugriff über LabVIEW und EXDUL.dll .....	92
<b>16. FAQ - Problembehandlung</b> .....	93
<b>17. Technische Daten</b> .....	98
<b>18. Beschaltungsbeispiele</b> .....	100
18.1 Beschaltung des Optokoppler-Eingangs .....	100
18.2 Beschaltung des Optokoppler-Ausgangs .....	102
<b>19. ASCII-Tabelle</b> .....	103
<b>20. Produkthaftungsgesetz</b> .....	106
<b>21. EG-Konformitätserklärung</b> .....	108

## 1. Produktbeschreibung

Das EXDUL-593E verfügt über 6 Temperaturmesseinheiten für die Sensoren PT100 und PT1000 mit jeweils eigener Stromquelle und Messeingängen. Die Messung der einzelnen Sensoren erfolgt per Software-Befehl. Dabei kann sowohl die Temperatur als auch der Sensorwiderstand gemessen werden.

Zusätzlich hat das Modul einen digitalen Eingang und einen digitalen Ausgang mit galvanischer Trennung über hochwertige Optokoppler und zusätzlichen Schutzdioden. Der spezielle leistungsfähige Ausgangsoptokoppler bewältigen einen Schaltstrom von bis zu 150 mA.

Über eine externe Spannungsquelle wird das Modul mit der notwendigen Betriebsspannung versorgt.

Über die integrierte Webpage kann das Modul benutzerfreundlich konfiguriert werden. Zudem ist auch ein einfacher Funktionstest möglich.

Die Kommunikation zwischen PC und Modul erfolgt über das Senden und Empfangen von Byte-Arrays über eine stabile TCP/IP-Verbindung, welche durch ein Handshakeprotokoll abgesichert ist.

























Die programmierbare Logik des Moduls kann sowohl für autarke Aktionen am Ausgang, als auch für Meldungen an den PC genutzt werden. Hierdurch ist ein Polling der Eingänge oft nicht mehr nötig und sowohl der Datenverkehr als auch die Rechnerauslastung kann wesentlich verringert werden.

Die programmierbare LCD-Anzeige ermöglicht die Darstellung von digitalen und analogen I/O-Statusinformationen oder programmierbaren anwenderspezifischen Daten.

Die Anschlüsse für die Spannungsversorgung sind wie die Anschlüsse des Eingangs- und Ausgangsoptokopplers einer 24poligen Schraubklemmleiste zugeführt. Das kompakte Gehäuse erlaubt den Einsatz als mobiles Modul am Notebook sowie als Steuermodul im Steuerungs- und Maschinenbau mit einfacher Wandmontage oder unkomplizierter Montage auf DIN EN-Tragschienen.

## 2. Anschlussklemmen

### 2.1 Klemmenbelegung von CN1

RTDIN0+	2 	 1	FORCE0+
FORCE1+	4 	 3	FORCE0-
FORCE1-	6 	 5	RTDIN1+
RTDIN2+	8 	 7	FORCE2+
FORCE3+	10 	 9	FORCE2-
FORCE3-	12 	 11	RTDIN3+
RTDIN4+	14 	 13	FORCE4+
FORCE5+	16 	 15	FORCE4-
FORCE5-	18 	 17	RTDIN5+
DOUT0-	20 	 19	DOUT0+
DIN0-	22 	 21	DIN0+
GND_EXT	24 	 23	Vcc_EXT

**Vcc\_EXT:**

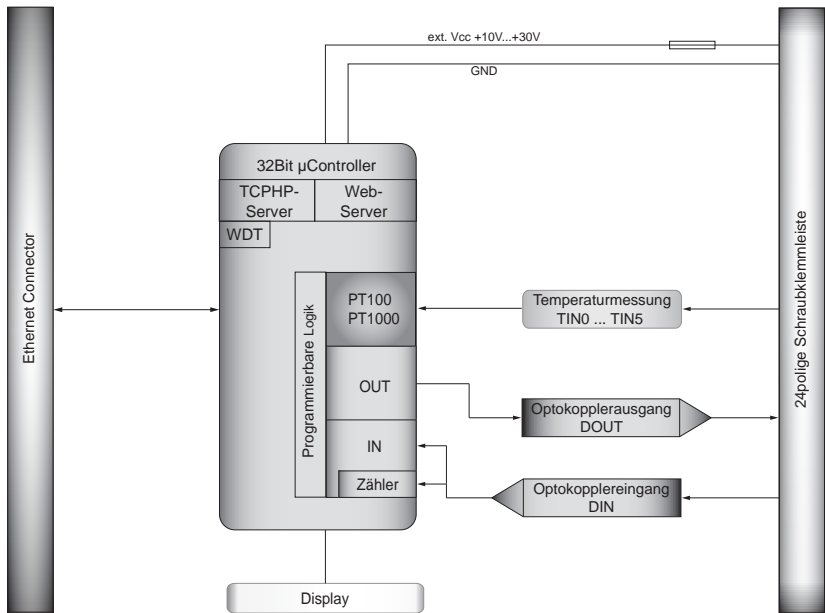
Anschlussklemme für externe Versorgungsspannung

**GND\_EXT:**

Masse-Anschluss

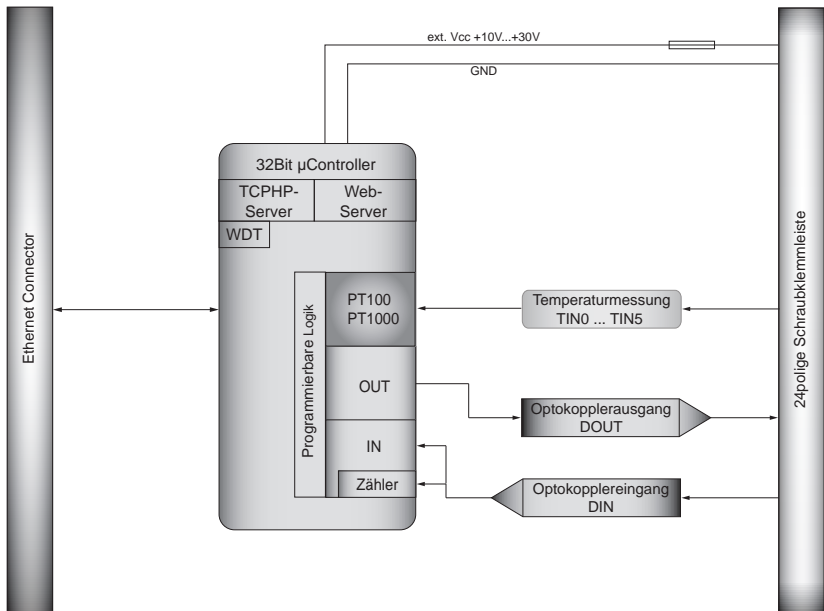
### 3. Systemkomponenten

#### 3.1 Blockschahtbild EXDUL-593E



Grafik 3.1 Blockschahtbild EXDUL-593E

### 3.2 Blockschahtbild EXDUL-593S



Grafik 3.2 Blockschahtbild EXDUL-593S



### **3.3 Digitaler Eingang über Optokoppler**

1 bipolarer Kanal mit galvanischer Trennung  
Überspannungsschutz-Dioden  
Eingangsspannungsbereich  
    high = 10..30 Volt  
    low = 0..3 Volt

### **3.4 Digitaler Ausgang über Optokoppler**

1 Kanal mit galvanischer Trennung über Leistungsoptokoppler  
Verpolungsschutz-Dioden  
Spannung-CE: max. 50 V  
Ausgangsstrom: max. 150mA

### **3.5 Zähler**

1 programmierbarer Zähler 32 Bit (belegt den Optokoppler-Eingang)  
Zählfrequenz: max. 5 kHz  
Automatische Sicherung der Zählerstände im 10kHz Takt

### **3.6 6 Temperaturmeseinheiten**

Sensortyp PT100 und PT1000 je Einheit individuell über Jumper wählbar  
3-Leiteranschluss

### **3.7 Programmierbare Logik**

Ausgänge über Verknüpfungen schalten  
Meldung an PC bei Zustandsänderung an den Eingängen senden

### **3.8 TCP/IP-Server**

Stabile und einfache Kommunikation mit einfachem Protokoll

### **3.9 Webserver**

Konfiguration

Funktionstest

Update der Firmware

### **3.10 Kommunikations-Watchdog-Timer**

Einstellbares Zeitfenster von 1 bis  $2^{32}$  - 1 ms

### **3.11 LCD Anzeige (nur EXDUL-593E)**

Matrixanzeige mit 2 Zeilen und 16 Spalten zur Darstellung von 16 Zeichen je Zeile

Programmierbar zur Darstellung anwendungsspezifischer Daten oder als I/O-Zustandsanzeige

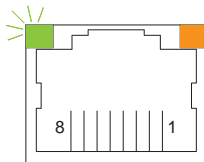
## 4. Inbetriebnahme

Der Netzwerk- bzw. PC-Anschluss erfolgt einfach und unkompliziert über eine Ethernet-Schnittstelle, die Konfiguration über einen Webbrowser. Für die Versorgung mit der notwendigen Betriebsspannung ist ein externes Netzteil notwendig.

### 4.1 Anschluss an einen Ethernet-Port

Das EXDUL-593E/EXDUL-593S verfügt über ein 10/100 Fast Ethernet Interface mit RJ45 Anschluss und wird über ein Netzwerkkabel direkt an einen PC, einen Ethernet-Hub oder an einen Ethernet-Switch angeschlossen.

Das Modul bootet nach Anlegen der Betriebsspannung, sobald eine stabile Ethernet-Verbindung besteht, leuchtet die linke LED an der RJ45-Buchse des EXDUL-593 kontinuierlich grün.



### 4.2 Anschluss der Betriebsspannung

Das EXDUL-593E / EXDUL-593S benötigt für den Betrieb eine Spannungsversorgung von +10 V ... +30 V DC an Klemme 23 (Vcc) und Klemme 24 (GND).

### 4.3 Integrierte Webpage des EXDUL-593E / EXDUL-593S

Der Zugriff auf die Webpage des EXDUL-593 ist über eine TCP/IP-Verbindung durch einen beliebigen Web-/Internetbrowser wie Mozilla Firefox<sup>®</sup>, Chrome<sup>®</sup>, Edge<sup>®</sup>, Safari<sup>®</sup> usw. möglich. Über die Webpage besteht die Möglichkeit, Verbindungsinformationen auszulesen und passwortgeschützt Konfigurationsdaten zu verändern. Vorgenommene Einstellungen werden im internen Speicher des EXDUL-593 gespeichert und beim Booten geladen. Die EXDUL-593-Webpage ermöglicht zudem das Beschreiben, Auslesen und Anzeigen der Anwender-Speicherbereiche UserA, UserB, UserLCD1m und UserLCD2m sowie das Starten und Stoppen des Zählers. Zudem ist das Testen der digitalen Ein- und Ausgänge möglich.

### 4.4 Passwortschutz - Zugangskennung

Wie bereits aufgeführt, ist über die EXDUL-Webpage die Netzwerk-Konfiguration, die Einstellung der LCD-Anzeige, das Beschreiben der User-Speicherbereiche sowie das Setzen der Ein- und Ausgänge und des Zählers möglich. Um unbefugten Zugriff zu vermeiden, sind diese Einstellungsbereiche durch ein Passwort geschützt.

Werksmäßig ist folgende Zugangskennung voreingestellt:

**Benutzerkennung:** admin  
**Kennwort:** 11111111

Groß-/Kleinschreibung beachten!

Falls über diese Zugangsdaten kein Zugriff möglich ist, wurde die Einstellung der Zugangskennung von Ihrem Systemadministrator geändert.

#### **4.5 Grundeinstellung Netzwerk-Konfiguration**

In der Grundeinstellung ist das EXDUL-593 auf DHCP (Dynamic Host Configuration Protocol), d.h. auf dynamische IP-Adresse eingestellt. Bei der Inbetriebnahme schickt das EXDUL-593 eine Nachfrage ins Netzwerk (LAN). Im Netzwerk, aktiver DHCP-Dienst vorausgesetzt, wird dem Modul daraufhin automatisch eine IP-Adresse zugewiesen. Diese Einstellung ermöglicht Ihnen einfach und unkompliziert den Anschluss des Moduls zum Anpassen der Konfigurationsdaten nach Ihren Bedürfnissen.

#### **4.6 Zusammensetzung und Aufbau der IP-Adresse**

IP4-Adressen bestehen aus 32 Bits = 4 Bytes (Oktetten). Jedes Byte kann einen Wert zwischen 0 und 255 annehmen. Die Darstellung erfolgt als vier Dezimalzahlen durch Punkte getrennt (z.B. 192.168.1.83).

Jede IP-Adresse enthält einen Netzwerk- und Geräteanteil (Hostanteil). Über die Subnetzmaske erfolgt die Trennung zwischen Netz- und Hostteil. Alle Geräte, die sich im gleichen Netzwerkteil befinden, können miteinander kommunizieren.

Beispiel:

Ist der IP-Adresse 192.168.1.83 die Subnetzmaske 255.255.0.0 zugeordnet, so befindet sich das Gerät im Netz 192.168.-.- als Gerät -.-.1.83.

## 4.7 Änderung der Netzwerk-Konfiguration

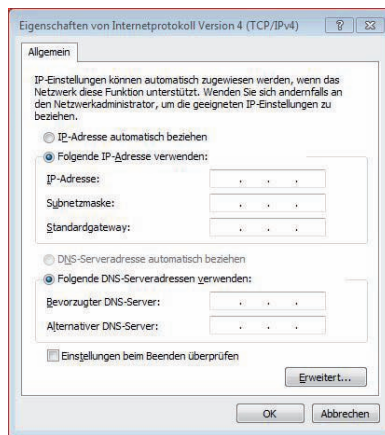
Zum Ändern der werksmäßigen Konfigurationseinstellungen schließen Sie das EXDUL-593 über die RJ45-Buchse und das beiliegende Standard-Netzwerkkabel an ein lokales Netzwerk an. Zudem ist eine Spannungsversorgung (+10 V ...+30 V) an Klemme 23 (Vcc) und Klemme 24 (GND) des Moduls nötig. Das EXDUL bootet jetzt, die linke LED an der RJ45-Buchse des EXDUL-593 leuchtet kontinuierlich grün, sobald eine stabile Ethernet-Verbindung besteht. Über einen beliebigen Web-/Internetbrowser ist durch Eingabe des Host-Namens **http://EXDUL-593** der Zugriff auf die EXDUL-593-Webpage möglich. Durch Anklicken des Buttons **TCP/IP Config** öffnet folgende Konfigurationsmaske:

The screenshot shows the EXDUL web interface for TCP/IP configuration. At the top, there is a navigation bar with 'Home', 'Status', 'Info', 'Config', and 'Peripherie'. The main heading is 'TCP/IP Konfiguration'. Below this, a note states: 'Diese Seite dient zur Einstellung und Änderung der Netzwerkparameter.' A red warning box contains the text: 'Achtung: Beachten Sie dringend die Hinweise im Handbuch, durch unkorrekte Einstellungen geht die Netzwerkverbindbarkeit verloren.' The configuration form includes the following fields: 'MAC Address' (44:04:1a:00:00:00), 'Host Name' (EXDUL-518), 'IP Address' (192.168.180.4), 'Gateway' (192.168.180.1), 'Subnet Mask' (255.255.255.0), 'Primary DNS' (192.168.100.1), and 'Secondary DNS' (0.0.0.0). There is a checked checkbox for 'Enable DHCP' and a 'Daten speichern' button at the bottom. The footer of the page reads 'EXDUL Webpage Copyright © 2021'.

Über das beiliegende Standard-Netzwerkkabel kann das EXDUL-593 auch direkt an einen PC mit Ethernet-Schnittstelle angeschlossen werden. Beim Direktanschluß steht im Normalfall kein DHCP-Dienst zur Verfügung, da weder der PC noch das EXDUL diesen bietet. Für diesen Fall ist das EXDUL-593 über die zuletzt eingestellte statische IP-Adresse ansprechbar. Für EXDUL-Neugeräte ist die IP-Adresse 169.254.1.1 gültig.

## Wichtiger Hinweis:

Zum Kommunizieren mit dem angeschlossenen EXDUL muss sich der verwendete PC im gleichen Netzwerk befinden. Dazu muss beim Rechner DHCP deaktiviert und z.B. von der EXDUL-IP-Adresse 192.168.1.83 (Subnetzmaske 255.255.0.0) die ersten zwei Zahlenblöcke (192.168) in die IP-Adresse des Rechners übernommen werden. Die zwei weiteren Blöcke können, jeweils Ihren Bedürfnissen angepasst, mit Werten zwischen 0 und 255 belegt werden.



Der Netzanteil der IP-Adresse ist von der Subnetzmaske abhängig. Von der IP-Adresse des EXDUL-593 müssen alle Oktette, die in der Subnetzmaske mit 255 belegt sind, in die IP-Adresse des Rechners übernommen werden (siehe auch Kapitel 4.6 Zusammensetzung und Aufbau der IP-Adresse).

## 4.8 Konfiguration mit statischer IP-Adresse (DHCP deaktiviert)

Für die Konfiguration des EXDUL-593 auf die statische IP-Adresse muss DHCP in der EXDUL-593-Konfigurationsmaske deaktiviert werden. Für die Konfigurationsänderung schließen Sie das EXDUL-593 wie unter Kapitel 4.7 (Änderung der Netzwerk-Konfiguration) an ein LAN oder einen PC an.

Sobald eine stabile Ethernet-Verbindung besteht, haben Sie über einen beliebigen Internetbrowser Zugriff auf die Webpage des EXDUL-593. Durch Eingabe des Host-Namens **http://EXDUL-593** sollte der Browser die EXDUL-593-Webpage öffnen, das Anklicken des Buttons TCP/IP Config öffnet folgende Konfigurationsmaske:

The screenshot shows the 'TCP/IP Konfiguration' web page for the EXDUL-593 device. At the top, there is a navigation bar with 'Home', 'Status', 'Info', 'Config', and 'Peripherie'. Below this, the title 'TCP/IP Konfiguration' is displayed. A message states: 'Diese Seite dient zur Einstellung und Änderung der Netzwerkparameter.' A red warning box contains the text: 'Achtung: Beachten Sie dringend die Hinweise im Handbuch, durch unkorrekte Einstellungen geht die Netzwerkkonnektivität verloren.' The configuration form includes the following fields: 'MAC Address' (04:34:3e:00:00:00), 'Host Name' (EXDUL-593), 'IP Address' (192.168.180.4), 'Gateway' (192.168.180.1), 'Subnet Mask' (255.255.255.0), 'Primary DNS' (192.168.180.1), and 'Secondary DNS' (0.0.0.0). There is an 'Enable DHCP' checkbox which is currently unchecked. A 'Daten speichern' button is located at the bottom of the form. The footer of the page reads 'EXDUL Webpage Copyright © 2021'.

Sobald Sie DHCP deaktivieren, können Sie Ihre gewünschte IP-Adresse, Subnetzmaske und den gewünschten Host-Namen eintragen. Durch Anklicken des Buttons **Konfiguration speichern** werden die aktuell eingetragenen Daten in den internen Speicher des EXDUL-593 übernommen. Das Modul ist ab diesem Zeitpunkt nur über die hier eingestellte IP-Adresse bzw. über den angegebenen Host-Namen ansprechbar. Der verwendete PC oder das LAN muss sich dazu im gleichen Netz befinden.

**Wichtiger Hinweis:** Jede IP-Adresse und jeder Host-Name darf nur einem Gerät bzw. Modul in einem Netzwerk zugeteilt werden, eine Doppelvergabe ist nicht zulässig! Der Host-Name darf beliebig gewählt werden, jedoch nur aus den ASCII-Zeichen 0-9 sowie A-Z (Groß/Kleinschreibung egal) und dem - (Bindestrich) bestehen. Bestimmte IP-Adressen sind reserviert bzw. haben eine Sonderfunktion wie z.B.: 127.0.0.1 (local Host), 192.168.1.0 (0 ist Adresse des Netzes) mit 255.255.255.0 (Subnetzmaske)



Bitte erkundigen Sie sich bei Ihrem Netzwerkadministrator, welche IP-Adresse Sie verwenden dürfen. Bei Verwendung unzulässiger IP-Adressen besteht die Möglichkeit, dass der Zugriff auf das Modul nicht mehr möglich ist. Die Einstellung der wichtigsten nicht zulässigen Adressen wird vom Modul blockiert.

#### **4.9 Konfiguration mit dynamischer IP-Adresse (DHCP aktiviert)**

Falls Sie das EXDUL-593 in ein bestehendes Netzwerk mit bereits aktiven DHCP-Server einbinden und über eine dynamische IP-Adresse ansprechen möchten, muss DHCP (Dynamic Host Configuration Protocol) in der Konfigurationsmaske der Webpage aktiviert sein. In der werksmäßigen Grundeinstellung ist DHCP bereits aktiviert und keine Umstellung notwendig.

Bei eingestellter statischer IP-Adresse wird DHCP wie folgt aktiviert: Schließen Sie dazu das EXDUL-593 über ein Netzkabel (bei älteren PCs über ein Crossover-Netzkabel) an einen Computer an. Der verwendete PC muss auf „**Folgende IP-Adresse verwenden: (DHCP-deaktiviert)**“ eingestellt sein. In die IP-Adresse des Rechners muss der Netzanteil (siehe dazu Kapitel 4.6, Zusammensetzung und Aufbau der IP-Adresse) aus der IP-Adresse des EXDUL-593 übernommen werden, denn der Computer und das EXDUL müssen sich im gleichen Netzwerk befinden. An den Klemmen 23 (Vcc) und 24 (GND) schließen Sie die Spannungsversorgung (+10 V ...+30 V) für das EXDUL-593 an. Das Modul bootet jetzt, die linke LED an der RJ45-Buchse des EXDUL-593 leuchtet kontinuierlich grün, sobald eine stabile Ethernet-Verbindung besteht. Den Zugriff auf die Webpage des EXDUL-593 ermöglicht Ihnen ein beliebiger Internetbrowser durch Eingabe des Host-Namens oder der IP-Adresse des EXDUL-593. Jetzt sollte der Browser die EXDUL-593-Webpage öffnen. Durch Anklicken des Buttons **TCP/IP Config** öffnet sich die Konfigurationsmaske, in der Sie wie in nachfolgender Abbildung **Enable DHCP** aktivieren:

**EXDUL**  
Home Status Info Config Peripherie

## TCP/IP Konfiguration

Diese Seite dient zur Einstellung und Änderung der Netzwerkparameter.

**Achtung:** Beachten Sie dringend die Hinweise im Handbuch, durch unkorrekte Einstellungen geht die Netzwerkkonnektivität verloren.

MAC Address: 04:34:3e:00:00:00  
 Host Name: EXDUL-593  
 Enable DHCP  
 IP Address: 192.168.180.4  
 Gateway: 192.168.180.1  
 Subnet Mask: 255.255.255.0  
 Primary DNS: 192.168.180.1  
 Secondary DNS: 0.0.0.0  
 Daten speichern

EXDUL Webpage Copyright © 2021

### Wichtiger Hinweis:

Bevor Sie DHCP aktivieren, ist eine Absprache mit Ihrem Netzwerkadministrator dringend notwendig. Sollten Sie mehrere gleiche Module aus der EXDUL-5xx-Serie in ein Netzwerk einbinden, müssen Sie den voreingestellten Host-Namen der einzelnen Module ändern. Jeder Host-Name darf innerhalb eines Netzwerkes nur einem Gerät bzw. Modul zugewiesen werden. Der Host-Name kann beliebig gewählt werden, jedoch nur aus den ASCII-Zeichen 0-9 sowie A-Z (Groß/Kleinschreibung) und dem - (Bindestrich) bestehen.

### 4.10 LCD-Anzeige während des Bootvorgangs (nur EXDUL-593E)

Während des Bootvorgangs des Moduls erscheint im Display eine Infoanzeige. In Zeile 1 wird der Modul-Name angezeigt, in Zeile 2 die Information, dass das Modul initialisiert wird. Sobald der Bootvorgang abgeschlossen ist, erscheint in beiden Fällen, je nach Einstellung, entweder die I/O-Statusanzeige oder die UserLCD-Anzeige.

#### **4.11 LCD-Anzeige während des Betriebs (nur EXDUL-593E)**

Nach dem Booten schaltet das Display, je nach Einstellung, von der Infoanzeige in die I/O-Statusanzeige oder UserLCD-Anzeige. Während der I/O-Statusanzeige werden in Zeile1 die aktuellen Zustände der Eingänge, in Zeile 2 die Zustände der Ausgänge angezeigt. Falls in der EXDUL-593 Webpage der UserLCD-Modus aktiviert ist, erscheint anstelle der I/O-Statusanzeige die UserLCD-Anzeige mit den Werten aus den Speicherbereichen UserLCD1m und UserLCD2m. Die Daten aus UserLCD1m und UserLCD2m werden solange auf dem LCD-Display angezeigt, bis neue Benutzerdaten über UserLCD-Zeile1 und UserLCD-Zeile2 auf die LCD-Anzeige geschrieben werden. Um einen „Screen-Burn“ zu vermeiden, wechselt die Anzeige im laufenden Betrieb etwa jede Minute für ca. fünf Sekunden von der I/O-Statusanzeige oder UserLCD-Anzeige in die Infoanzeige mit der aktuellen IP-Adresse.

## 5. Zugriff auf das EXDUL-593

Der Zugriff auf die Konfigurationseinstellungen und auf den Ein- und Ausgang des EXDUL-593 ist wie bereits erwähnt über die EXDUL-593 Webpage sowie über TCP/IP-Sockets möglich. Dazu wird die IP-Adresse, der Host-Name oder die MAC-Adresse benötigt.

### 5.1 Zugriff über die EXDUL Webpage

Die Webpage des EXDUL-593 ermöglicht es den Eingang zu lesen, den Ausgang zu setzen, die Anwender-Speicherbereiche UserA, UserB und UserLCD sowie die Verbindungs- oder Modulinformationen auszulesen und die Konfigurationsdaten zu verändern. Der Zugriff auf die Webpage ist von jedem mit dem Modul verbundenen Computer über einen beliebigen Internetbrowser möglich. Der verwendete PC muss auf **IP-Adresse automatisch beziehen (DHCP-aktiviert)** eingestellt sein, soweit sich das EXDUL-593 noch im Auslieferungszustand (DHCP aktiviert) befindet und in ein Netzwerk mit aktiven DHCP-Dienst integriert ist. Durch Eingabe des Host-Namens (Im Auslieferungszustand **http://EXDUL-593**, ansonsten den von Ihnen eingestellten Host-Namen, evtl. über ExdulUtility\_v2\_xx oder spätere Version feststellen) oder der IP-Adresse können Sie die Webpage öffnen. Falls ein Öffnen der EXDUL-593-Webpage nicht möglich ist, überprüfen Sie die Netzwerkverbindung oder den eingegebenen Host-Namen. Nähere Informationen finden Sie im Kapitel FAQ - Problembehandlung.

[Home](#) [Status](#) [Info](#) [Config](#) [Peripherie](#)

## EXDUL-593

Die EXDUL-593E und EXDUL-593S sind netzwerkfähige, analoge, digitale I/O-Module mit Ethernet-Interface.

Jedes Modul verfügt zur Temperaturmessung über sechs Messeinheiten mit jeweil eigener Stromquelle und Messeingängen. Zusätzlich verfügen sie über einen digitalen Eingang und einen digitalen Ausgang mit galvanischer Trennung über hochwertige Optokoppler und zusätzlichen Schutzdioden. Der Optokoppler-Eingang kann bei Bedarf auch mit einem 32Bit Zähler-Eingang programmiert und genutzt werden.

Das EXDUL-593E bietet zusätzlich eine LCD-Anzeige zur Darstellung von I/O-Statusinformationen oder anwenderspezifischen Daten. Die Anschlüsse für die notwendige externe Spannungsversorgung sind wie die Anschlüsse des Eingangs- und Ausgangsoptokoppler der 24poligen Schraubklemmleiste zugeführt.

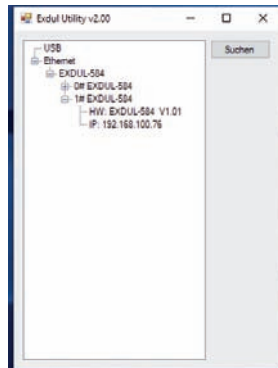
EXDUL Webpage Copyright © 2021

## 5.2 Zugriff über TCP/IP-Sockets

Mit der Verwendung des TCP-Protokolls wird eine zuverlässige Verbindung zwischen PC und dem EXDUL-593 erreicht. Das Protokoll ergreift selbstständig Maßnahmen bei Datenverlust. Die Adressierung des Moduls findet über eine 4 Byte IP-Adresse (IPv4) bzw. über den vergebenen Host-Namen und einer Portnummer 9760 statt. Der PC versendet über die Verbindung für jeden Befehl ein Byte-Array. Das Modul verarbeitet den Befehl und sendet immer eine Rückantwort. In Kombination mit einer Hochsprache ist über die TCP/IP-Verbindung das Lesen der Eingänge, das Setzen der Ausgänge, das Starten, Stoppen und Auslesen des Zählers, das Beschreiben der User-Speicherbereiche, das Auslesen der Verbindungs- und Modulinformationen sowie das Verändern der Konfigurationsdaten möglich. Insgesamt können 3 TCP/IP-Verbindungen gleichzeitig mit dem Modul geöffnet sein.

### 5.3 Host-Namen, IP-Adresse und MAC-Adresse feststellen

Falls Sie für EXDUL-5xx-Module weder den Host-Namen noch die IP-Adresse oder MAC-Adresse kennen, ermöglicht Ihnen das Suchprogramm ExdulUtility\_v2\_xx (oder höher) diese festzustellen. Falls Ihre Firewall die Kommunikation des Suchprogrammes mit den EXDUL-5xx verhindert, ist eine Freigabe für das Programm in der Firewall erforderlich.



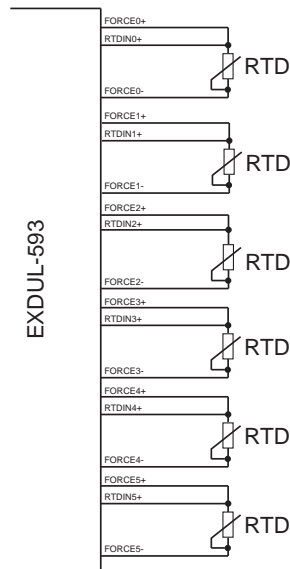
Das Suchprogramm **ExdulUtility\_v2\_xx** (oder höher) befindet sich auf der EXDUL-Software-CD oder steht auf [www.messcomp.com](http://www.messcomp.com) zum Download bereit.

## 6. Temperaturmesseinheiten

Das Modul EXDUL-593 besitzt zur Temperaturmessung mit PT100- und PT1000-Sensoren (IEC 751  $\alpha = 0.00385$ ) 6 Messeinheiten, welche zur Bestimmung der Temperatur eine 3-Leiterschaltung verwendet. Die 3-Leiterschaltung sorgt für eine automatische Messfehlerkompensation der Sensorleitungen und führt zu einer genaueren Temperaturmessung. Dabei ist darauf zu achten, dass die Leitungsadern zum Sensor den selben Widerstand besitzen müssen (identische Länge, Querschnitt und Material). Jede Messeinheit liefert während der Temperaturmessung den nötigen Messstrom und übergibt dem Anwender nach abgeschlossener Messung den Temperaturwert und bei Bedarf den Widerstandswert.

Um für die Temperaturmessung den idealen Messstrom zu gewährleisten, besitzt jede Messeinheit einen eigenen Jumper, mit welchem die Sensorart (PT100 oder PT1000) konfiguriert werden kann.

### 6.1 Beschaltung



## 6.2 Messmöglichkeiten

Bei der Durchführung einer Messung gibt es mehrere Modi, welche bestimmen, wie das Messergebnis verarbeitet oder als Befehlsantwort an den Anwender geschickt werden soll (z.B. mΩ oder °C)

### 6.2.1 Widerstandsmessung

In diesem Modus wird der angeschlossene Widerstand gemessen und in mΩ an den PC übergeben. Der zu messende Widerstand darf im Bereich von 0 bis 370Ω liegen. Die Messart ist nur in der PT100-Konfiguration verfügbar.

### 6.2.2 Temperaturmessung PT100/PT100 nach IEC 751

Wird dieser Modus verwendet, so misst das Modul den Sensorwiderstand und berechnet die daraus resultierende Temperatur mit Hilfe der aus IEC 751 ( $\alpha=0.00385$ ) vorgegebenen Kennlinie des Sensors. Die Temperatur wird mit dem Faktor 100 an den PC zurück geschickt.

Verwendete Callendar-Van Dusen Koeffizienten:

$$a = 3.908030 \times 10^{-3}$$

$$b = -5.7750 \times 10^{-7}$$

$$c = -4.18301 \times 10^{-12}$$

## 6.3 Fehlererkennung

Um Fehler bei der Temperaturmessung erkennen zu können, gibt es die Möglichkeit, einen Fehlertest durchzuführen. Hier können Fehler wie Leiterbruch, Kurzschlüsse und Über-/Unterspannung erkannt werden.

Wird mit Aufruf des entsprechenden Befehls ein Fehlertest durchgeführt, wird nach einigen wenigen ms ein Fehlerbyte an den Anwender gesendet, welches der Fehleranalyse dient.

Während des Fehlertests kann keine Temperaturmessung durchgeführt werden.



## Codes bei Fehlermeldung

Fehler-Bit	mögliche Fehlerursache	Fehlerbeschreibung
D7	reserviert	
D6	reserviert	
D5	Fehler bei der Verdrahtung	
D4	Fehler bei der Verdrahtung	
D3	Fehler bei der Verdrahtung	
D2	Overvoltage oder Under-voltage	evtl. externe Spannung eingespeist

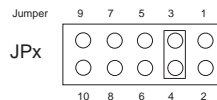
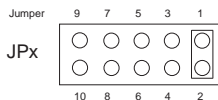
Steht in einem Bit eine 1, so liegt entsprechend dessen Bedeutung ein Fehler vor.

## 6.4 Sensorart konfigurieren

Jede Messeinheit des Moduls ist mit den Sensoren PT100 und PT1000 kompatibel. Um einen idealen Messstrom generieren zu können, muss der gewünschte Sensortyp zum einen innerhalb der Software (Softwarebefehl oder Webpage) eingestellt werden und zum anderen der passende Jumper im Modul gesetzt werden.

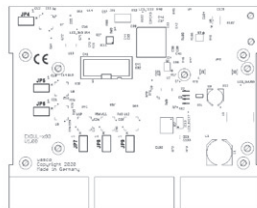
Um Werkzustand sind alle Messeinheiten für den PT100 konfiguriert. Soll mit einer oder mehreren Einheiten ein PT1000 Sensor gemessen werden, so gehen Sie wie folgt vor:

1. Entfernen Sie den Gehäusedeckel. Dafür müssen Sie mit einem Schraubenzieher das Gehäuse an den 4 Laschen auf der Bodenrückseite etwas nach außen drücken und den Deckel herunterziehen.
2. Setzen Sie die Jumper wie gewünscht. Für eine PT100-Messung setzen sie den Jumper zwischen Pin 1 und 2, für eine PT1000-Messung zwischen Pin 3 und 4 (siehe Grafiken).



Folgende Jumperblöcke sind den jeweiligen Messeinheiten zugeordnet.

Jumperblock-Nummer	Messeinheit
JP4	Messeinheit 0
JP5	Messeinheit 1
JP6	Messeinheit 2
JP7	Messeinheit 3
JP8	Messeinheit 4
JP9	Messeinheit 5



3. Anschließend montieren Sie den Deckel wieder auf den Boden. Achten Sie dabei darauf, dass die Kontaktklemmen richtig in die Führungen des Deckels passen.

## **6.5 Abgleich bzw. Kalibrierung der Messeinheiten**

Jede Messeinheit des Moduls wird vor der Auslieferung bereits für die Sensoren PT100 und PT1000 mit Präzisionswiderständen abgeglichen. Sollte ein erneuter Abgleich aufgrund besonderer Einflüsse (hohe oder niedrige Außentemperatur) nötig sein, so kann dieser entweder mit dem Kalibrierungsbefehl im Anwenderprogramm oder über die Webpage abgeglichen werden. Vor der Kalibrierung muss sowohl an die abzugleichende Messeinheit entweder ein genauer Messwiderstand (PT100 = 100R, PT1000 = 1K) oder ein auf 0°C gekühlter Sensor angeschlossen werden als auch die verwendete Sensorart konfiguriert werden. Beachten Sie, dass die Leitungslängen/Leitungswiderstände identisch sind.

## 7. 1 Optokopplereingang

Das EXDUL-593 verfügt über einen Eingangskanal, dessen galvanische Trennung mittels Optokoppler erreicht wird. Die Isolationsspannung des Optokopplers zwischen Masse des Moduls und Eingang beträgt 500 Volt.

### 7.1 Pinbelegung der Eingangsoptokoppler

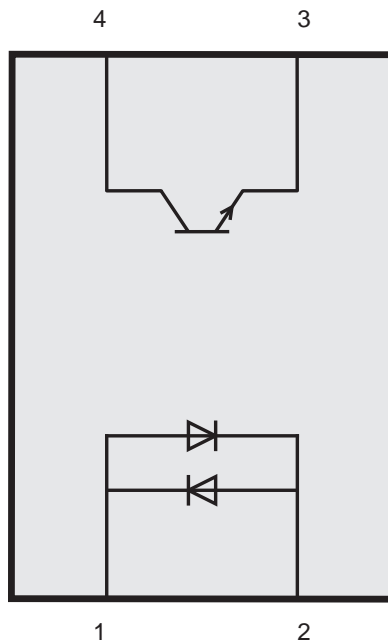


Abb. 7.1 Pinning Eingangsoptokoppler

## 7.2 Eingangsbeschaltung

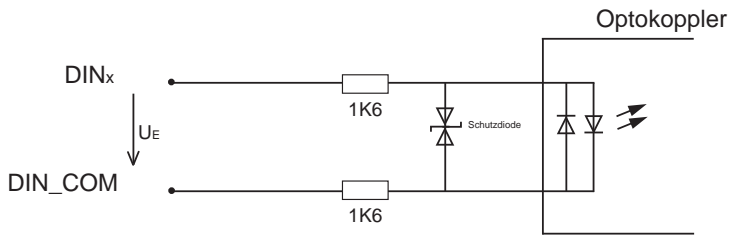


Abb. 7.2 Eingangsbeschaltung

Die Eingänge des Optokopplers sind bipolar ausgeführt. Im Normalfall wird der DIN<sub>COM</sub>/DIN<sub>0</sub>- Anschluß auf Minus gelegt und am DIN<sub>0</sub>+ -Anschluss eine Spannung angelegt. Sie können jedoch auch, falls es schaltungstechnisch sinnvoller ist, am DIN<sub>COM</sub>/DIN<sub>0</sub>- Anschluß die Plus-Spannung und am DIN<sub>0</sub>+ Anschluß die Minus-Spannung anlegen.

## 7.3 Eingangsstrom

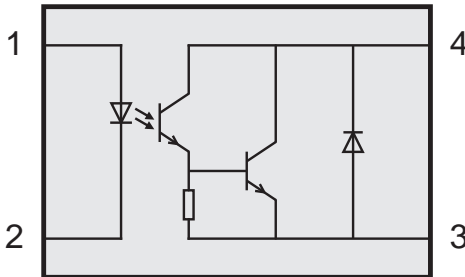
$$I_E \approx \frac{U_E - 1,1V}{3200\Omega}$$

Bei einer Eingangsspannung zwischen DIN<sub>0</sub>+ und DIN<sub>0</sub>- von 24 Volt ergibt sich ein Eingangsstrom von ca. 7mA, bei 12V von ca. 3,4 mA.

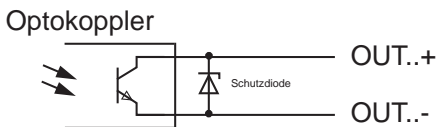
## 8. 1 Optokopplerausgang

Das EXDUL-Modul verfügt über 1 Ausgangskanal, dessen galvanische Trennung ebenfalls mittels Optokoppler erreicht wird. Die Isolationsspannung zwischen Masse des Moduls und Ausgang beträgt 500V.

### 8.1 Pinbelegung des Ausgangsoptokopplers



### 8.2 Ausgangsbeschaltung



### 8.3 Ausgangsdaten

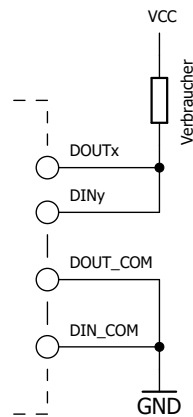
Spannung-CE:	max. 50V
Spannung-EC	0.1V
Strom-CE:	150mA

## 8.4 Programmierung des Optokopplerausgangs

Für die Programmierung des Optokopplerausgangs stehen mehrere Zugriffsfunktionen zur Verfügung. So kann der Ausgangskanal einfach beschrieben werden. Sollte der aktuelle Zustand des Ausgangs im Anwenderprogramm nicht speicherbar sein, so kann dieser über einen Lese-Befehl rückgelesen werden. Hier ist zu beachten, dass der gelesene Schaltzustand nur dem primären Zustand (auf Prozessorseite) entspricht. Soll der tatsächliche Schaltzustand bzw. der Pegel des Ausgangs rückgelesen werden, muss dafür ein Optokopplereingang verwendet werden (siehe Kapitel Optokopplerausgang rücklesen). Eine detaillierte Beschreibung der Programmierung ist im Kapitel Programmierung zu finden.

## 8.5 Optokopplerausgang rücklesen

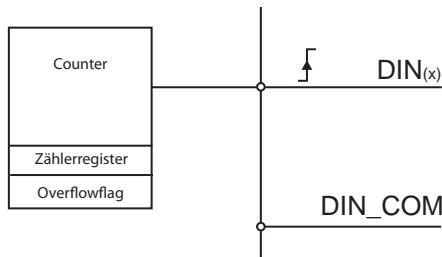
Gelegentlich ist es bei Applikationen nötig, den tatsächlichen Ausgangszustand in dem Programm zu wissen. Ein Beispiel kann z. B. eine Fehlererkennung im Programm sein. Dies kann durch die Rückführung des Optokopplerausgangs zu dem Optokopplereingang erfolgen. Im folgenden Beschaltungsbeispiel wird der Ausgang x mit dem Eingang y verbunden. Beachten Sie bitte, daß in dieser Schaltungsversion das Ergebnis des Eingangs negiert ist. Wenn der Ausgang geschaltet ist, liegt am Transistor keine Spannung an und somit zeigt der Eingang eine „0“ an. Zudem fließt bei nicht geschaltetem Ausgang über den Optokoppler-Eingang und damit auch über den Verbraucher ein geringer Strom (bei 24V ca. 7mA).



## 9. Zähler

Das Modul stellt für den Optokopplereingang (DIN0) einen eigenständigen, hardwareunterstützten 32bit Zähler zur Verfügung. Bei Bedarf kann dieser aktiviert werden und reagiert bei jeder steigenden Flanke durch die Inkrementierung des Zählerstandes. Ein abrufbares Flag signalisiert einen Overflow.

Damit bei einer außerplanmäßigen Unterbrechung der Spannungsversorgung der Zählerstand nicht verloren geht, wird dieser ungefähr alle 100µs gesichert. Liegt dann am Modul wieder eine Spannungsversorgung an, wird der gesicherte Wert automatisch in das Zählerregister geladen.



Funktionen	Beschreibung
Start	Startet den Zähler bzw. gibt den Eingang frei
Stop	Stoppt den Zähler, Signal an dem Eingang wird ignoriert
Reset	Setzt den Zählerstand auf 0
Zählerstand lesen	Liest den aktuellen Zählerstand
Overflowflag lesen	Liest das Overflowflag
Clear Overflowflag	Setzt das Overflowflag zurück

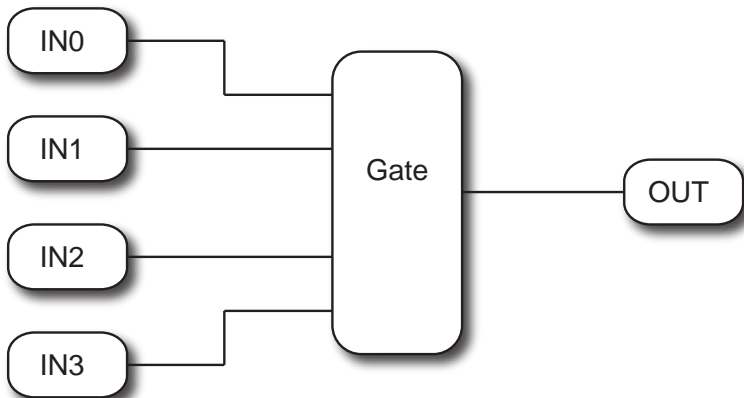


## 10. Programmierbare Logik

In verschiedenen Anwendungen kann es von Vorteil sein, wenn das Modul selbständig auf Eingangssignale bzw. Änderungen an den Eingängen reagiert. Ohne diese Möglichkeit kann eine Änderung am Eingang nur durch regelmäßige Abfrage der Eingänge mittels Polling erkannt werden. Dieses Polling verursacht eine erhöhte Auslastung des Netzwerkes und des Rechners.

Um diese eigenständige Reaktion des Moduls zu erzeugen, werden vier Logikzweige mit jeweils vier Logikeingängen, einer Verknüpfung sowie einem Logikausgang zur Verfügung gestellt. In Abbildung 10.1 ist einer dieser Zweige zu sehen.

Den Logikeingängen bzw. Ausgängen können einzelne Funktionen wie später näher erläutert zugewiesen werden.



## 10.1 Logikzweig

Im Modul können vier verschiedene Logikzweige verwendet werden. Jeder Zweig hat vier Eingänge, eine Logikverknüpfung (UND, ODER) und einen Ausgang. Den Logikeingängen sowie Logikausgängen können verschiedene Funktionen zugewiesen werden. Dabei kann die Logikverknüpfung eventgesteuert sein (z.B. Flanke an den Eingängen invertiert Optokopplerausgang oder sendet Nachricht an PC) als auch zustandsgesteuert (z.B. wenn Eingang DIN0 gleich HIGH dann ist DOUT0 ebenfalls HIGH). Bei der Auswahl der Funktionen ist darauf zu achten, dass bei einem Eventausgang (z.B. Message an PC oder Optokopplerausgang DOUT0 setzen) immer ein Eventeingang im Logikzweig vorhanden ist. Ist einem Logikausgang die Funktion NONE zugewiesen, so ist dieser Zweig automatisch deaktiviert und schont damit die Prozessorressourcen.

## 10.2 Logikeingänge

Jedem Logikzweig stehen vier Logikeingänge zur Verfügung. Diesen können folgende Funktionen zugewiesen werden.

Eingangsfunktion	Beschreibung	zustand/event
TRUE	Eingang liefert immer eine 1	zustand
FALSE	Eingang liefert immer eine 0	zustand
DINxx	Logikeingang wird mit dem jeweiligen Optokopplereingang verknüpft	zustand
DINxx_EDGE	Logikeingang erkennt eine steigende Flanke und liefert für einen Zweigtakt (~10ms) eine 1	event
TEMPx_OVER_STATE	Temperatur am Temperatureingang x hat die eingestellte Schwelle (HighThreshold) überschritten	zustand
TEMPx_UNDER_STATE	Temperatur am Temperatureingang x hat die eingestellte Schwelle (LowThreshold) unterschritten	zustand
TEMPx_OVER_EVENT	Temperatur am Temperatureingang x hat die eingestellte Schwelle (HighThreshold) überschritten und liefert für einen Zweigtakt eine 1	event
TEMPx_UNDER_EVENT	Temperatur am Temperatureingang x hat die eingestellte Schwelle (LowThreshold) unterschritten und liefert für einen Zweigtakt eine 1	event

Als Ausgangsbasis für die Logikeingänge werden die Optokopplereingänge in regelmäßigen Zeitabständen (ca. 1ms) abgetastet. Hierbei werden Logikpegel oder Pegeländerungen erfaßt und an die programmierbare Logik übergeben.

### 10.3 Logikverknüpfung

Jedem Logikzweig steht eine Logikverknüpfung zur Verfügung. Dieser können folgende Funktionen zugewiesen werden.

Ausgangsfunktion	Beschreibung	zustand/event
AND	UND-Verknüpfung	zustand/event
OR	ODER-Verknüpfung	zustand/event

### 10.4 Logikausgänge

Jedem Logikzweig steht ein Logikausgang zur Verfügung. Diesem können folgende Funktionen zugewiesen werden.

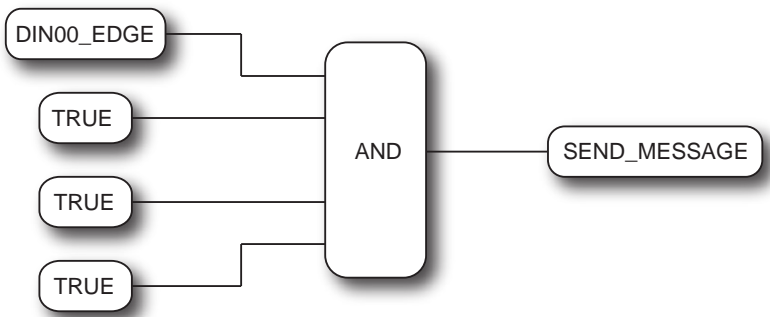
Ausgangsfunktion	Beschreibung	zustand/event
NONE	Logikzweig ist deaktiviert	zustand
SEND_MESSAGE <sub>x</sub>	Modul sendet eine Message(x) an den PC	event
WRITE_DOUT <sub>xx</sub>	Optokopplerausgang wird mit dem Ergebnis der Logikverknüpfung beschrieben	zustand
,SET_DOUT <sub>xx</sub>	Bei einer 1 der Verknüpfung wird der jeweilige Optokopplerausgang durchgeschaltet	event
CLEAR_DOUT <sub>xx</sub>	Bei einer 1 der Verknüpfung wird der jeweilige Optokopplerausgang gesperrt	event
TOGGLE_DOUT <sub>xx</sub>	Bei einer 1 der Verknüpfung wird der jeweilige Optokopplerausgang invertiert	event

### 10.5 Message an PC schicken

Möchten Sie bei einem Event eine Benachrichtigung an den PC bzw. an die Anwendung senden, geht dies über die Message-Ausgänge. Es gibt insgesamt 4 Message-Funktionsausgänge, von welcher jeder eine eigene individuelle Nachricht an den PC schickt.

Für diese Funktion muss in dem Programm eine weitere TCP/IP-Verbindung mit dem Modul im Receiver-Modus aufgebaut werden. Liegt nun z.B. eine steigende Flanke an einem der zu überwachenden Eingänge der programmierbaren Logik an, wird automatisch eine Message an den Receiver gesendet. Diese Message muss nicht mit einer Rückantwort bestätigt werden.

Folgendes Beispiel veranschaulicht die Konfiguration.



In diesem Beispiel soll bei jeder steigenden Flanke an DIN00 eine Nachricht an den PC gesendet werden. Dafür wird für den Eingangsfunktionsblock IN0 die Funktion DIN00\_EDGE ausgewählt und für alle anderen TRUE. Als Gater wird eine UND-Verknüpfung (AND) und als Ausgang SEND\_MESSAGE1 gewählt. Liegt nun eine steigende Flanke an DIN00 an, so wird an den TCP/IP Client mit der Receiver-Verbindung ein 12 Byte langes Array gesendet.

Damit der PC unterscheiden kann, von welchem Logikblock die Meldung erfolgt, stehen 4 Message-Arrays zur Verfügung. In folgender Tabelle werden die 4 möglichen Byte-Arrays dargestellt. Die ersten 4 Byte stellen den Befehl, in diesem Fall immer 0x0E, sowie das Längenbyte (immer 2) dar. Byte 4 bis Byte 6 sind reserviert und haben den Wert 0x00. In Byte 7 steht der Message-Index. Wurde als Ausgang die Message1 ausgewählt steht in diesem Byte der Wert 1. Zur Kontrolle der Reihenfolge der gesendeten Meldungen, bzw. um auch das Fehlen einer Meldung erkennen zu können, beinhaltet des Byte-Array zusätzlich einen Zähler. Byte 8 bis Byte 11 entsprechen dem Wert des Receivernachrichten-Zählers. Bei jeder Nachricht an den PC wird dieser Wert incrementiert.

Receivernachrichtenzähler:

Wert = dd\*0x100000 + cc\*0x10000 + bb\*0x100 + aa

Byte-Index	0	1	2	3	4	5	6	7	8	9	10	11
Message1	0x0E	0x00	0x00	0x02	0x00	0x00	0x00	0x01	aa	bb	cc	dd
Message2	0x0E	0x00	0x00	0x02	0x00	0x00	0x00	0x02	aa	bb	cc	dd
Message3	0x0E	0x00	0x00	0x02	0x00	0x00	0x00	0x03	aa	bb	cc	dd
Message4	0x0E	0x00	0x00	0x02	0x00	0x00	0x00	0x04	aa	bb	cc	dd

## 10.6 Timing der programmierbaren Logik

	typ. Bearbeitungsfrequenz
Logikzweig-Update	100 Hz
Input-Abtastung	1 kHz
Temperaturmessung	1/6 Hz

## 11. Watchdog-Timer (WDT)

Das EXDUL-Modul besitzt einige Sicherheitsmechanismen für eine stabile Kommunikation. Sollte es jedoch z.B. Aufgrund von Routingproblemen zu einer Unterbrechung der Kommunikation kommen und ein Wiederaufbau der Verbindung nicht mehr möglich sein, kann der integrierte Watchdog-Timer des Moduls zu Problembeseitigung verwendet werden.

Zu Beginn der Applikation muss der WDT mit einer ausgewählten Periodendauer initialisiert und gestartet werden. Wird der Timer nicht innerhalb der eingestellten Periodendauer durch einen PC-Befehl zurückgesetzt, so wird ein vollständiger Reset des EXDUL-Moduls durchgeführt. Beim Auslösung des Watchdog-Resets wird das entsprechende Flag im Error-Register gesetzt. Hierdurch kann später durch das Überprüfen dieses Bits der Reset erkannt werden.

## 12. Error-Register

Kommt es zu außerplanmäßigen Fehlern (z.B. einem Watchdog-Reset), so werden diese in den beiden Error-Registern angezeigt. Tritt ein Fehler auf, so wird das dem Fehler zugewiesene Bit gesetzt. Die Fehlerregister bleiben auch nach einem Reset oder dem Entfernen der Spannungsversorgung erhalten. Die Register können mit dem entsprechenden Befehl zurückgesetzt werden.

<b>Error-Register 0</b>	<b>Bit31 .. Bit2</b>	<b>Bit1</b>	<b>Bit0</b>
<b>Bedeutung</b>	reserviert	WDT_SW	reserviert
<b>Error-Register 1</b>	<b>Bit31 .. Bit0</b>		
<b>Bedeutung</b>	reserviert		

Beschreibung:

WDT\_SW: 1 = Der Watchdog-Reset wurde ausgeführt.



## 13. Informations-, LCD- und Userregister

### 13.1 Register HW-Kennung und Seriennummer

Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
HW-Kennung	E	X	D	U	L	-	5	9	3			V	1	.	0	1
	45 <sub>hex</sub>	58 <sub>hex</sub>	44 <sub>hex</sub>	55 <sub>hex</sub>	4C <sub>hex</sub>	2D <sub>hex</sub>	35 <sub>hex</sub>	39 <sub>hex</sub>	33 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	56 <sub>hex</sub>	31 <sub>hex</sub>	3E <sub>hex</sub>	30 <sub>hex</sub>	31 <sub>hex</sub>
S/N	1	0	4	4	0	2	6									
	31 <sub>hex</sub>	30 <sub>hex</sub>	34 <sub>hex</sub>	34 <sub>hex</sub>	30 <sub>hex</sub>	32 <sub>hex</sub>	36 <sub>hex</sub>									

Tabelle 12.1 Register HW-Kennung und Seriennummer

Im Register HW-Kennung ist der Modulname sowie die Version der Firmware abgelegt und kann zur Feststellung der Produkt-Identität vom User gelesen werden. In der o. a. Tabelle sind als Beispiel in der Zeile HW-Kennung jeweils der Hex-Wert und das dazugehörige ASCII-Zeichen für das Modul EXDUL-593 mit Firmware-Version 1.01 dargestellt.

Das Register Serien-Nummer kann vom Anwender lediglich gelesen werden. Die Serien-Nummer in der o. a. Tabelle dient als Formatbeispiel. In der Zeile S/N ist jeweils der Hex-Wert und darüber das dazugehörige ASCII-Zeichen für die Serien-Nummer 1044026 dargestellt.

### 13.2 Speicherbereiche UserA, UserB, UserLCD1m\* und UserLCD2m\*

Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
UserA																
	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>
UserB																
	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>
UserLCD1m*																
	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>
UserLCD2m*																
	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>	20 <sub>hex</sub>

In den Registern UserA, UserB, UserLCD1m\* und UserLCD2m\* können jeweils 16 Stellen (16 Byte) zur eigenen Verwendung genutzt werden. Die Daten bleiben beim Ausschalten erhalten, ein Default-Reset setzt diese Register in die Werkseinstellung (Auslieferungszustand) zurück. Im Auslieferungszustand steht in allen vier User-Speicherbereichen an jeder Stelle der Hex-Wert 20, der im ASCII-Code einem Leer-Zeichen entspricht. In der o. a. Tabelle sind jeweils der Hex-Wert und darüber das dazugehörige ASCII-Zeichen dargestellt.

### 13.3 Display-Register UserLCD-Zeile1\*, UserLCD-Zeile2\* und LCD-Kontrast\*

Die Register UserLCD-Zeile1 und UserLCD-Zeile2 dienen bei aktivierten UserLCD-Modus zum Beschreiben der beiden LCD-Zeilen mit jeweils 16 beliebigen Zeichen. Mit Übernahme der Daten erfolgt die Anzeige im Display anstelle der Daten aus UserLCD1m\* und UserLCD2m\*. Die Daten in den Registern UserLCD-Zeile1 und UserLCD-Zeile2 bleiben beim Ausschalten **nicht** erhalten. Über das Register LCD-Kontrast ist der Display-Kontrast einstellbar, der auch beim Ausschalten erhalten bleibt.

\*: Nur für EXDUL-593E zutreffend, bei EXDUL-593S ohne Funktion!

## **14. Installation des Treibers**

Für das Ethernetmodul EXDUL-593 ist kein Treiber notwendig. Voraussetzung ist eine bereitgestellte Netzwerkverbindung von PC (Netzwerkkarte mit Treiber) oder mobilen Device. Für den direkten Zugriff auf das Modul werden die bei vielen Hochsprachen wie C, C++, C#, Visual Basic oder Java vorhandenen TCP/IP-Libraries benötigt. Beispiele in mehreren Programmiersprachen für den Zugriff befinden sich auf der mitgelieferten CD und auf unserer Website [www.messcomp.com](http://www.messcomp.com).

## 15. Programmierung

### 15.1 Einführung

Die Programmierung erfolgt mit Hilfe des Standard TCP/IP-Protokolls und somit über sogenannte TCP/IP-Sockets, für die in vielen gängigen Programmiersprachen Standardbibliotheken vorhanden sind. Eine einfache und schnelle Implementierung ermöglicht die Verwendung des .Net Frameworks von Microsoft. Durch das Verwenden des Standard-Protokolls kann das Modul neben Windows auch mit einer Vielzahl von anderen Betriebssystemen wie Ubuntu (Linux-basierend) oder Android verbunden werden. Verschiedene Programmierbeispiele sind auf der beiliegenden CD und auf unserer Website bereitgestellt.

### 15.2 Programmierarten

Für den Zugriff auf das EXDUL-Modul gibt es mehrere Möglichkeiten. So kann für die Programmierung unter Windows und .NET die Library EXDUL.dll verwendet werden. Diese ermöglicht einen leichten und schnellen Einstieg, um den Zugriff auf das Modul zu programmieren. Des Weiteren können auch TCP und Socket Libraries verwendet werden, welche bei vielen Programmiersprachen wie C# oder Java vorhanden sind. Sie ermöglichen oft eine breite Einstellmöglichkeit der Schnittstelle. LabVIEW-Anwender können ebenfalls mit Hilfe der EXDUL.dll oder den TCP-Funktionsblöcken leicht auf das Modul zugreifen.

### 15.3 Programmierung unter Windows mit der .NET EXDUL.dll Library

Wird für den Modul-Zugriff eine .NET-Programmiersprache verwendet (C#, C++.NET oder VB.NET), so kann die Library EXDUL.dll verwendet werden. Sie besitzt einen objektorientierten Aufbau, in welchem jedes EXDUL-Modul durch ein Objekt mit ihren Methoden dargestellt wird.

Bei der Entwicklung der Library wurde auf eine möglichst einheitliche API zwischen den unterschiedlichen EXDUL-Modulen geachtet.

Dies ermöglicht es dem Anwender, bei Bedarf ohne großen Programmieraufwand von z.B. einem USB-EXDUL-Modul auf ein Ethernet-EXDUL-Modul zu wechseln.

## Befehlsübersicht für EXDUL.dll-Library

### Open:

[bool](#) Open()

Rückgabewerte: true wenn erfolgreich / false bei Fehler

Zusammenfassung: Verbindung zu Modul aufbauen

---

### Close

void Close()

Zusammenfassung: Verbindung zu Modul schließen

---

### Schreiben in Inforegister:

void SetModullInfo ([byte](#) type, [string](#) info)

Parameter: type: Info-Typ (siehe Handbuch)

info: Bis zu 16 Zeichen langer Info-String

Zusammenfassung: Beschreibt die Modul-Informationsregister

Infobereich	Info-Byte
UserA	0
UserB	1

---

### Lesen aus Inforegister:

[string](#) GetModullInfo([byte](#) type)

Parameter: type: Info-Typ (siehe Handbuch)

Rückgabewerte: Gibt das Register "type" als string zurück

Zusammenfassung: Liest die Modul-Information-Register aus

Infobereich	Info-Byte
UserA	0
UserB	1
Hardwarekennung	3
Seriennummer	4

## Schreiben in LCD-Register UserLCD:

void SetUserLCD([byte](#) *line*, [string](#) *text*)

Parameter: *line*: 0 = 1. Zeile / 1 = 2. Zeile

*text*: Bis zu 16 Zeichen langer LCD-Text

Zusammenfassung: Beschreibt die UserLCD-Register. Der Parameter *line* legt die Zeile (0 oder 1) fest und *text* den Text aus 16 Zeichen.

---

## Schreiben in LCD-Register UserLCDm:

void SetUserLCDm([byte](#) *line*, [string](#) *text*)

Parameter: *line*: 0 = 1. Zeile / 1 = 2. Zeile

*text*: Bis zu 16 Zeichen langer LCD-Text

Zusammenfassung: Beschreibt die UserLCDm-Register. Der Parameter *line* legt die Zeile (0 oder 1) fest und *text* den Text aus 16 Zeichen

---

## Schreiben des LCD-Modes:

void SetLCDMode([byte](#) *mode*)

Parameter: *mode*: LCD-Modus

Zusammenfassung: Setzt den LCD-Modus fest

LCD-Modus	LCD-Modus-Byte
IO-Mode	0
User-Mode	1

**Lesen des LCD-Modes:**[byte](#) GetLCDMode()

Rückgabewerte: LCD-Modus

Zusammenfassung: Liest den LCD-Modus aus

LCD-Modus	LCD-Modus-Byte
IO-Mode	0
User-Mode	1

---

**Schreiben LCD-Kontrastwert:**void SetLCDContrast([ushort](#) contrast)Parameter: *contrast*: Wert zwischen 0 und 4095 (empfohlen  
800 bis 1800)

Zusammenfassung: Legt den LCD-Kontrast fest

---

**Lesen LCD-Kontrastwert:**[ushort](#) GetLCDContrast()

Rückgabewerte: LCD-Kontrast

Zusammenfassung: Liest den LCD-Kontrast aus

## Optokopplerausgänge lesen:

[uint](#) GetOptoOut()

Rückgabewerte: Zustand der Optokopplerausgänge

Zusammenfassung: Liest den Zustand der Optokopplerausgänge

---

## Optokopplerausgänge schreiben:

void SetOptoOut([uint](#) value)

Parameter: *value*: Zustand der Ausgänge

Zusammenfassung: Setzt die Optokopplerausgänge

---

## Optokopplerausgang schreiben:

void WriteOptoOut(byte channel, [uint](#) value)

Parameter: *channel*: Index des Ausgangskanals

*value*: Zustand der Ausgänge

Zusammenfassung: Setzt die Optokopplerausgänge

---

## Optokopplereingänge lesen:

[uint](#) GetOptoIn()

Rückgabewerte: Aktueller Zustand der Optokopplereingänge

Zusammenfassung: Liest den aktuellen Zustand an den Optokopplereingängen

---



## Zähler starten:

void StartCounter([byte](#) index)

Parameter: *index*: Counter-Index

Zusammenfassung: Startet den Zähler mit der Nummer index

---

## Zähler stoppen:

void StopCounter([byte](#) index)

Parameter: *index*: Counter-Index

Zusammenfassung: Stoppt den Zähler mit der Nummer index

---

## Zähler resettten:

void ResetCounter([byte](#) index)

Parameter: *index*: Counter-Index

Zusammenfassung: Setzt den Zählerstand des Zählers mit der Nummer index zurück auf 0

---

## Zählerstand lesen:

[uint](#) ReadCounter([byte](#) index)

Parameter: *index*: Counter-Index

Rückgabewerte: Zählerstand

Zusammenfassung: Liest den Zählerstand des Zählers mit der Nummer index aus

---

## Overflow-Flag lesen:

[bool](#) ReadOverflowFlagCounter([byte](#) *index*)

Parameter: *index*: Counter-Index

Rückgabewerte: Overflowflag false = kein Overflow  
true = Overflow

Zusammenfassung: Liest das Overflowflag des Zählers mit der Nummer *index* aus

---

## Overflow-Flag rücksetzen:

void ResetOverflowFlagCounter([byte](#) *index*)

Parameter: *index*: Counter-Index

Zusammenfassung: Setzt das Overflowflag des Zählers mit der Nummer *index* zurück

---

## Temperatur messen:

[int](#) GetTemperature\_PT([byte](#) *channel*)

Parameter: *channel*: Kanalauswahl

Rückgabewerte: Temperatur in °C \* 100

Zusammenfassung: Misst die Temperatur durch den angeschlossenen PT-Sensor

---

## Temperatursensor-Typ einstellen:

void SetSensortypeTempUnit([byte](#) *channel*, [byte](#) *sensor*)

Parameter: *channel*: Kanalauswahl

*sensor*: Sensortyp (0=PT100, 1=PT1000)

Zusammenfassung: Stellt den Sensortyp ein und speichert ihn ab

## Temperatur-Messeinheit kalibrieren:

void CalibrateTempUnit\_PT([byte](#) channel)

Parameter: *channel*: Kanalauswahl

Zusammenfassung: Mit der Funktion kann die Messeinheit zur Temperaturmessung kalibriert werden. Dazu muss ein im Handbuch festgelegter Widerstand angelegt werden

---

## Fehlererkennung an der Temperaturmesseinheit durchführen:

[int](#) FaultDetectionTempUnit([byte](#) channel)

Parameter: *channel*: Kanalauswahl

Rückgabewerte: Fehlerregister

Zusammenfassung: Führt eine Fehlererkennung an der gewünschten Temperaturmesseinheit channel durch

---

## Werksreset:

void DefaultReset()

Zusammenfassung: Setzt das Modul auf die Werkseinstellung zurück. Nach dem Befehl muss das Modul geschlossen und wieder neu geöffnet werden

## 15.4 Programmierung mit TCP-Libraries

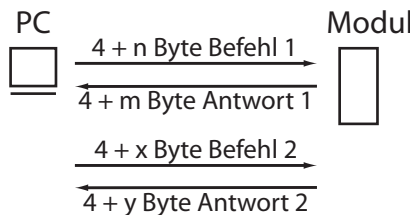
Durch die Möglichkeit mit Standard TCP/Socket-Libraries auf das Modul zugreifen zu können, kann der Anwender mit einer Vielzahl an Sprachen seine Anwendung auf verschiedenen Plattformen programmieren. So kann unter Windows neben dem .NET-Framework auch Delphi oder Java verwendet werden. Auch können Anwendungen auf vielen Linux basierten Betriebssystemen mit z.B. C oder Java entworfen werden. Dabei ist zu beachten, dass das Modul immer als Server fungiert.

### 15.4.1 Kommunikation mit dem EXDUL-593

Der Datenaustausch erfolgt durch Senden bzw. Empfangen von Byte-Arrays mit unterschiedlicher Länge über die TCP/IP-Schnittstelle.

Jeder erlaubte Sendestring wird mit einem definierten Ergebnis- bzw. Bestätigungsstring beantwortet.

Vor dem Senden eines Strings muss der letzte Ergebnis- bzw. Bestätigungsstring gelesen werden.



Grafik 14.4 Kommunikationsmodell

### 15.4.2 Befehls- und Datenformat

Der Datenaustausch erfolgt durch Senden und Empfangen von Byte-Arrays. Jedes zu sendende bzw. zu empfangende Byte-Array besteht aus mindestens 4 Bytes. Dabei stellen die ersten drei Bytes den Befehl und das vierte die Anzahl der noch folgenden 4 Byte-Blöcke dar.

Befehl Byte 0	Befehl Byte 1	Befehl Byte 2	Längenbyte
------------------	------------------	------------------	------------

Die Anzahl der 4-Byte-Blöcke variiert von Befehl zu Befehl und ist zum Teil von der zu sendenden Datenmenge abhängig. Genauere Informationen befinden sich bei den einzelnen Befehlsbeschreibungen.

### 15.4.3 Passwortschutz

Um das Modul vor unbefugten Zugriffen zu schützen, kann für den Datenaustausch ein einfacher Passwortschutz verwendet werden. Ist dieser aktiviert (siehe Befehl Securitykonfiguration), so muss an jedes gesendete Byte-Array das richtige Passwort (+8 Bytes) angehängt werden. Ist das Passwort falsch oder wurde es nicht dem Befehlsstring hinzugefügt, so wird eine Error-Antwort zurückgeschickt. Die Rückantwort vom Modul bleibt unverändert

Beispiel: Optokopplerausgänge schreiben mit aktiviertem Passwortschutz und dem Default-Passwort „11111111“

Byte	Senden	Empfangen	Beschreibung
0	08	08	Befehlscode 1. Byte
1	00	00	Befehlscode 2. Byte
2	00	00	Befehlscode 3. Byte
3	03 (→ 12Byte)	00	Längenbyte
4	00		r/w Byte
5	0w 00 (gesperrt) 01 (durchgeschaltet)		Optokopplerzustand
6	00		reserviert
7	00		reserviert
8	31 <sub>hex</sub>		Passwort 1. Zeichen 1 <sub>ascii</sub>
9	31 <sub>hex</sub>		Passwort 2. Zeichen 1 <sub>ascii</sub>
10	31 <sub>hex</sub>		Passwort 3. Zeichen 1 <sub>ascii</sub>
11	31 <sub>hex</sub>		Passwort 4. Zeichen 1 <sub>ascii</sub>
12	31 <sub>hex</sub>		Passwort 5. Zeichen 1 <sub>ascii</sub>
13	31 <sub>hex</sub>		Passwort 6. Zeichen 1 <sub>ascii</sub>
14	31 <sub>hex</sub>		Passwort 7. Zeichen 1 <sub>ascii</sub>
15	31 <sub>hex</sub>		Passwort 8. Zeichen 1 <sub>ascii</sub>

Der normale Befehl ohne Passwortschutz beinhaltet nur 8 Bytes und im Längenbyte steht der Wert 1. Wie in der Tabelle zu sehen, wurden nun 8 weitere Bytes, in welchen das Passwort steht, angehängt. Zusätzlich muss das Längenbyte um den Wert 2 (+8Byte) erhöht werden. Die Rückantwort beinhaltet kein Passwort und entspricht dem normalen Antwortarray ohne Passwort.

## 15.4.4 Befehlsübersicht

Hexcode	Beschreibung
0C 00 00	Inforegister lesen und schreiben
0C 00 03	LCD-Register lesen und schreiben
0C 00 08	Netzwerkkonfigurationen lesen und schreiben
0C 00 0C	Securitykonfigurationen lesen und schreiben
0C 00 0D	Passwort ändern
08 00 00	Optokopplerausgänge lesen und schreiben
08 00 01	Optokopplereingänge bearbeiten
09 00 00	Zähler0
09 00 01	Zähler1
09 00 02	Zähler2
09 00 03	Zähler3
09 00 04	Zähler4
0A 04 00	Temperaturmesseinheit Temperatur-/Widerstandsmessung
0A 04 01	Temperaturmesseinheit Fehlererkennung durchführen
0A 04 08	Temperaturmesseinheit Sensortyp einstellen
0A 04 09	Temperaturmesseinheit oberer Schwellenwert definieren
0A 04 0A	Temperaturmesseinheit unterer Schwellenwert definieren
0A FF F7	Temperaturmesseinheit kalibrieren
0C 01 01	Software Watch Dog Timer
0C 02 10	Initialisierung eines programmierbaren Logikzweiges
0C 03 00	Receivermodus aktivieren/deaktivieren + Messagezähler lesen
FF 00 00	Fehlerregister auslesen/rücksetzen

## 15.4.5 Befehlszusammensetzung

### Schreiben in Inforegister

Das EXDUL-Modul stellt mehrere beschreibbare Inforegister zur Verfügung. UserA/B sind zwei 16-Byte-Bereiche für den Anwender, um Informationen in einem nicht-flüchtigen Speicher (FLASH) zu sichern. Die Register sind nur als ganzer 16-Byte-Block beschreibbar.

Infobereich	Info-Byte
UserA	0
UserB	1

Beispiel: Schreiben der Zeichenfolge EXDUL-593 in Register UserA und UserB

Byte	Senden	Empfangen	Beschreibung
0	0C	0C	Befehlscode 1. Byte
1	00	00	Befehlscode 2. Byte
2	00	00	Befehlscode 3. Byte
3	05	00	Längenbyte → 20 Byte
4	00 (UserA) 01 (UserB)		Info-Byte
5	00		reserviert
6	00		reserviert
7	00		Schreibfunktion Infobereich
8	45		Daten 1. Zeichen E <sub>asci</sub>
9	58		Daten 2. Zeichen X <sub>asci</sub>
10	44		Daten 3. Zeichen D <sub>asci</sub>
11	55		Daten 4. Zeichen U <sub>asci</sub>
12	4C		Daten 5. Zeichen L <sub>asci</sub>
13	2D		Daten 6. Zeichen _ <sub>asci</sub>
14	35		Daten 7. Zeichen 5 <sub>asci</sub>
15	39		Daten 8. Zeichen 9 <sub>asci</sub>
16	33		Daten 9. Zeichen 3 <sub>asci</sub>
17	20		Daten 10. Zeichen [Leer] <sub>asci</sub>
18	20		Daten 11. Zeichen [Leer] <sub>asci</sub>
19	20		Daten 12. Zeichen [Leer] <sub>asci</sub>
20	20		Daten 13. Zeichen [Leer] <sub>asci</sub>
21	20		Daten 14. Zeichen [Leer] <sub>asci</sub>
22	20		Daten 15. Zeichen [Leer] <sub>asci</sub>
23	20		Daten 16. Zeichen [Leer] <sub>asci</sub>



## Lesen aus Inforegister

Das EXDUL-Modul besitzt mehrere 16-Byte breite Infobereiche, in welchen Modulinformationen wie die Seriennummer oder die Hardwarekennung stehen. Des Weiteren kann der Anwender auch die beschreibbaren User-Register auslesen.

Infobereich	Info-Byte
UserA	0
UserB	1
Hardwarekennung	3
Seriennummer	4

Info: Alle Infobereiche lassen sich nur als ganzer 16-Byte-Block auslesen.

Beispiel: Infobereich UserA auslesen (User-String = „EXDUL-593“)

Gesendet wird ein 8Byte langer Block und empfangen ein 20Byte langer Block mit Inhalt von UserA bzw. UserB

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	0C	Befehlscode 1. Byte	0C	Befehlscode 1. Byte
1	00	Befehlscode 2. Byte	00	Befehlscode 2. Byte
2	00	Befehlscode 3. Byte	00	Befehlscode 3. Byte
3	01	Längenbyte → 4Byte	04	Längenbyte → 16Byte
4	00 (UserA) 01 (UserB)	Info-Byte	45	Daten 1. Zeichen E <sub>ASCII</sub>
5	00	reserviert	58	Daten 2. Zeichen X <sub>ASCII</sub>
6	00	reserviert	44	Daten 3. Zeichen D <sub>ASCII</sub>
7	01	Lesefunktion Infobereich	55	Daten 4. Zeichen U <sub>ASCII</sub>
8			4C	Daten 5. Zeichen L <sub>ASCII</sub>
9			2D	Daten 6. Zeichen r <sub>ASCII</sub>
10			35	Daten 7. Zeichen 5 <sub>ASCII</sub>
11			39	Daten 8. Zeichen 9 <sub>ASCII</sub>
12			33	Daten 9. Zeichen 3 <sub>ASCII</sub>
13			20	Daten 10. Zeichen [Leer] <sub>ASCII</sub>
14			20	Daten 11. Zeichen [Leer] <sub>ASCII</sub>
15			20	Daten 12. Zeichen [Leer] <sub>ASCII</sub>
16			20	Daten 13. Zeichen [Leer] <sub>ASCII</sub>
17			20	Daten 14. Zeichen [Leer] <sub>ASCII</sub>
18			20	Daten 15. Zeichen [Leer] <sub>ASCII</sub>
19			20	Daten 16. Zeichen [Leer] <sub>ASCII</sub>

Beispiel: Infobereich Hardwarekennung auslesen

Gesendet wird ein 8Byte langer Block und empfangen ein 20Byte langer Block mit der Hardwarekennung

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	0C	Befehlscode 1. Byte	0C	Befehlscode 1. Byte
1	00	Befehlscode 2. Byte	00	Befehlscode 2. Byte
2	00	Befehlscode 3. Byte	00	Befehlscode 3. Byte
3	01	Längenbyte → 4Byte	04	Längenbyte → 16Byte
4	04	Info-Byte	45	Daten 1. Zeichen E <sub>ascii</sub>
5	00	reserviert	58	Daten 2. Zeichen X <sub>ascii</sub>
6	00	reserviert	44	Daten 3. Zeichen D <sub>ascii</sub>
7	01	Lesefunktion Infobereich	55	Daten 4. Zeichen U <sub>ascii</sub>
8			4C	Daten 5. Zeichen L <sub>ascii</sub>
9			2D	Daten 6. Zeichen * <sub>ascii</sub>
10			35	Daten 7. Zeichen 5 <sub>ascii</sub>
11			39	Daten 8. Zeichen 9 <sub>ascii</sub>
12			33	Daten 9. Zeichen 3 <sub>ascii</sub>
13			20	Daten 10. Zeichen [Leer] <sub>ascii</sub>
14			20	Daten 11. Zeichen [Leer] <sub>ascii</sub>
15			20	Daten 12. Zeichen [Leer] <sub>ascii</sub>
16			20	Daten 13. Zeichen [Leer] <sub>ascii</sub>
17			20	Daten 14. Zeichen [Leer] <sub>ascii</sub>
18			20	Daten 15. Zeichen [Leer] <sub>ascii</sub>
19			20	Daten 16. Zeichen [Leer] <sub>ascii</sub>

Beispiel: Infobereich Seriennummer auslesen

Gesendet wird ein 8Byte langer Block und empfangen ein 20Byte langer Block mit der Seriennummer

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	0C	Befehlscode 1. Byte	0C	Befehlscode 1. Byte
1	00	Befehlscode 2. Byte	00	Befehlscode 2. Byte
2	00	Befehlscode 3. Byte	00	Befehlscode 3. Byte
3	01	Längenbyte → 4Byte	03	Längenbyte → 16Byte
4	04	Info-Byte	31	Daten 1. Zeichen 1 <sub>dez</sub>
5	00	reserviert	30	Daten 2. Zeichen 0 <sub>dez</sub>
6	00	reserviert	34	Daten 3. Zeichen 4 <sub>dez</sub>
7	01	Lesefunktion Infobereich	34	Daten 4. Zeichen 4 <sub>dez</sub>
8			30	Daten 5. Zeichen 0 <sub>dez</sub>
9			32	Daten 6. Zeichen 2 <sub>dez</sub>
10			36	Daten 7. Zeichen 6 <sub>dez</sub>
11				reserviert
12				reserviert
13				reserviert
14				reserviert
15				reserviert
16				reserviert
17				reserviert
18				reserviert
19				reserviert

## Schreiben in LCD-Register

Das EXDUL-Modul stellt mehrere beschreibbare LCD-Register zur Verfügung. UserLCD1 und UserLCD2 entsprechen den beiden Zeilen während der UserMode-LCD-Anzeige. UserLCD1m und UserLCD2m sind zwei 16-Byte-Bereiche, welche direkt in einen nicht-flüchtigen Speicher (FLASH) abgelegt werden und beim Modulstart in die Register UserLCD1m bzw. UserLCD2m geladen werden. Alle Register sind nur als ganze 16-Byte-Blöcke beschreibbar.

LCD-Befehl	LCD-Befehl-Byte
UserLCD1	0
UserLCD2	1
UserLCD1m	2
UserLCD2m	3

Beispiel: Schreiben der Zeichenfolge EXDUL-593 in Register

Byte	Senden	Empfangen	Beschreibung
0	0C	0C	Befehlscode 1. Byte
1	00	00	Befehlscode 2. Byte
2	03	03	Befehlscode 3. Byte
3	05	00	Längenbyte → 20 Byte
4	00 (UserLCD1) 01 (UserLCD2) 02 (UserLCD1m) 03 (UserLCD2m)		LCD-Befehl
5	00		reserviert
6	00		reserviert
7	00		Schreibfunktion
8	45		Daten 1. Zeichen E <sub>asci</sub>
9	58		Daten 2. Zeichen X <sub>asci</sub>
10	44		Daten 3. Zeichen D <sub>asci</sub>
11	55		Daten 4. Zeichen U <sub>asci</sub>
12	4C		Daten 5. Zeichen L <sub>asci</sub>
13	2D		Daten 6. Zeichen _ <sub>asci</sub>
14	35		Daten 7. Zeichen 5 <sub>asci</sub>
15	39		Daten 8. Zeichen 9 <sub>asci</sub>
16	33		Daten 9. Zeichen 3 <sub>asci</sub>
17	20		Daten 10. Zeichen [Leer] <sub>asci</sub>
18	20		Daten 11. Zeichen [Leer] <sub>asci</sub>
19	20		Daten 12. Zeichen [Leer] <sub>asci</sub>
20	20		Daten 13. Zeichen [Leer] <sub>asci</sub>
21	20		Daten 14. Zeichen [Leer] <sub>asci</sub>
22	20		Daten 15. Zeichen [Leer] <sub>asci</sub>
23	20		Daten 16. Zeichen [Leer] <sub>asci</sub>

### Lesen von LCD-Register

Das EXDUL-Modul stellt mehrere beschreib- bzw. lesbare LCD-Register zur Verfügung. UserLCD1 und UserLCD2 entsprechen den beiden Zeilen während der UserMode-LCD-Anzeige. UserLCD1m und UserLCD2m sind zwei 16-Byte-Bereiche, welche direkt in einen nicht-flüchtigen Speicher (FLASH) abgelegt werden und beim Modulstart in die Register UserLCD1m bzw. UserLCD2m geladen werden. Alle Register sind nur als ganze 16-Byte-Blöcke lesbar.

LCD-Befehl	LCD-Befehl-Byte
UserLCD1 & UserLCD2	0
UserLCD1m & UserLCD2m	2

Beispiel: Lesen der Zeichenfolge EXDUL-593 aus Register

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	0C	Befehlscode 1. Byte	0C	Befehlscode 1. Byte
1	00	Befehlscode 2. Byte	00	Befehlscode 2. Byte
2	03	Befehlscode 3. Byte	03	Befehlscode 3. Byte
3	01	Längenbyte → 4 Byte	08	Längenbyte → 32 Byte
4	00 (UserLCD1&2) 02 (UserLCD1m&2m)	LCD-Befehl	45	Daten Zeile1 1. Zeichen E <sub>asci</sub>
5	00	reserviert	58	Daten Zeile1 2. Zeichen X <sub>asci</sub>
6	00	reserviert	44	Daten Zeile1 3. Zeichen D <sub>asci</sub>
7	01	Lesefunktion von LCD-Registern	55	Daten Zeile1 4. Zeichen U <sub>asci</sub>
8			4C	Daten Zeile1 5. Zeichen L <sub>asci</sub>
9			2D	Daten Zeile1 6. Zeichen ^ <sub>asci</sub>
10			35	Daten Zeile1 7. Zeichen 5 <sub>asci</sub>
11			39	Daten Zeile1 8. Zeichen 9 <sub>asci</sub>
12			33	Daten Zeile1 9. Zeichen 3 <sub>asci</sub>
13			20	Daten Zeile1 10. Zeichen [Leer] <sub>asci</sub>
14			20	Daten Zeile1 11. Zeichen [Leer] <sub>asci</sub>
15			20	Daten Zeile1 12. Zeichen [Leer] <sub>asci</sub>
16			20	Daten Zeile1 13. Zeichen [Leer] <sub>asci</sub>
17			20	Daten Zeile1 14. Zeichen [Leer] <sub>asci</sub>
18			20	Daten Zeile1 15. Zeichen [Leer] <sub>asci</sub>
19			20	Daten Zeile1 16. Zeichen [Leer] <sub>asci</sub>
20			45	Daten Zeile2 1. Zeichen E <sub>asci</sub>
21			58	Daten Zeile2 2. Zeichen X <sub>asci</sub>
22			44	Daten Zeile2 3. Zeichen D <sub>asci</sub>
23			55	Daten Zeile2 4. Zeichen U <sub>asci</sub>
24			4C	Daten Zeile2 5. Zeichen L <sub>asci</sub>
25			2D	Daten Zeile2 6. Zeichen ^ <sub>asci</sub>
26			35	Daten Zeile2 7. Zeichen 5 <sub>asci</sub>
27			39	Daten Zeile2 8. Zeichen 9 <sub>asci</sub>
28			33	Daten Zeile2 9. Zeichen 3 <sub>asci</sub>
29			20	Daten Zeile2 10. Zeichen [Leer] <sub>asci</sub>
30			20	Daten Zeile2 11. Zeichen [Leer] <sub>asci</sub>
31			20	Daten Zeile2 12. Zeichen [Leer] <sub>asci</sub>
32			20	Daten Zeile2 13. Zeichen [Leer] <sub>asci</sub>
33			20	Daten Zeile2 14. Zeichen [Leer] <sub>asci</sub>
34			20	Daten Zeile2 15. Zeichen [Leer] <sub>asci</sub>
35			20	Daten Zeile2 16. Zeichen [Leer] <sub>asci</sub>

## Schreiben des LCD-Modes

Die LCD-Anzeige des EXDUL-Moduls stellt mehrere Anzeige-Modi bereit. Diese können mit folgendem Befehl eingestellt werden. Der LCD-Modus wird in einem nicht-flüchtigen Speicher abgelegt und wird auch nach einem Neustart des Moduls verwendet

LCD-Modus	LCD-Modus-Byte
IO-Mode	0
User-Mode	1

Beispiel: Schreiben des LCD-Modes

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	0C	Befehlscode 1. Byte	0C	Befehlscode 1. Byte
1	00	Befehlscode 2. Byte	00	Befehlscode 2. Byte
2	03	Befehlscode 3. Byte	03	Befehlscode 3. Byte
3	02	Längenbyte → 8 Byte	00	Längenbyte → 0 Byte
4	04	LCD-Befehl LCD-Mode		
5	00	reserviert		
6	00	reserviert		
7	00	Schreibfunktion		
8	00 (IO-Mode) 01 (User-Mode)	LCD-Modus		
9	00	reserviert		
10	00	reserviert		
11	00	reserviert		

## Lesen des LCD-Modes

Die LCD-Anzeige des EXDUL-Moduls stellt mehrere Anzeige-Modi bereit. Der eingestellte LCD-Modus kann mit folgendem Befehl ausgelesen werden.

LCD-Modus	LCD-Modus-Byte
IO-Mode	0
User-Mode	1

Beispiel: Lesen des LCD-Modes

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	0C	Befehlscode 1. Byte	0C	Befehlscode 1. Byte
1	00	Befehlscode 2. Byte	00	Befehlscode 2. Byte
2	03	Befehlscode 3. Byte	03	Befehlscode 3. Byte
3	01	Längenbyte → 4 Byte	01	Längenbyte → 4 Byte
4	04	LCD-Befehl LCD-Mode	00 (IO-Mode) 01 (User-Mode)	LCD-Modus
5	00	reserviert	00	reserviert
6	00	reserviert	00	reserviert
7	01	Lesefunktion	00	reserviert



## Schreiben LCD-Kontrastwert

Über diesen Befehl ist der Display-Kontrast einstellbar. Werte zwischen 0 bis 4095 werden akzeptiert. Der Display-Kontrast verringert sich mit ansteigendem Wert. Eine angenehme Darstellung wird im Bereich 800 bis 1800 erreicht.

Beispiel: Schreiben Display-Kontrast-Wert 800

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	0C	Befehlscode 1. Byte	0C	Befehlscode 1. Byte
1	00	Befehlscode 2. Byte	00	Befehlscode 2. Byte
2	03	Befehlscode 3. Byte	03	Befehlscode 3. Byte
3	02	Längenbyte → 8 Byte	00	Längenbyte → 0 Byte
4	0B	LCD-Befehl LCD-Kontrast		
5	00	reserviert		
6	00	reserviert		
7	00	Schreibfunktion		
8	50	Kontrastwert (Lowbyte - 00...FF)		
9	03	Kontrastwert (Highbyte - 00...0F)		
10	00	reserviert		
11	00	reserviert		

## Lesen LCD-Kontrastwert

Über diesen Befehl ist der Display-Kontrast auslesbar. Der Wert kann zwischen 0 bis 4095 liegen. Der Display-Kontrast verringert sich mit ansteigendem Wert. Eine angenehme Darstellung wird im Bereich 800 bis 1800 erreicht.

Beispiel: Lesen Display-Kontrast-Wert 800

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	0C	Befehlscode 1. Byte	0C	Befehlscode 1. Byte
1	00	Befehlscode 2. Byte	00	Befehlscode 2. Byte
2	03	Befehlscode 3. Byte	03	Befehlscode 3. Byte
3	01	Längenbyte → 4 Byte	01	Längenbyte → 4 Byte
4	0B	LCD-Befehl LCD-Kontrast	50	Kontrastwert (Lowbyte - 00...FF)
5	00	reserviert	03	Kontrastwert (Highbyte - 00...0F)
6	00	reserviert	00	reserviert
7	01	Lesefunktion	00	reserviert

## Schreiben Netzwerkkonfigurationen

Dieser Befehl ermöglicht das Abändern aller Netzwerkkonfigurationen wie IP-Adresse, Subnetzmaske, Hostname, Gateway, DNS-Adressen sowie die Einstellung der DHCP-Client-Funktion.

Beispiel: Schreiben der Netzwerkkonfigurationen

Hostname = „EXDUL-593“, IP = 192.168.0.63, Subnetzmaske = 255.255.255.0,

Gateway = 192.168.0.1, Primary DNS = 192.168.0.1, Secondary DNS = 217.237.151.115

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	0C	Befehlscode 1. Byte	0C	Befehlscode 1. Byte
1	00	Befehlscode 2. Byte	00	Befehlscode 2. Byte
2	08	Befehlscode 3. Byte	08	Befehlscode 3. Byte
3	0B	Längenbyte → 44 Byte	00	Längenbyte → 0 Byte
4	00	reserviert		
5	00	reserviert		
6	00	reserviert		
7	00	Schreibfunktion		
8	45	Hostname 1. Zeichen E <sub>ascii</sub>		
9	58	Hostname 2. Zeichen X <sub>ascii</sub>		
10	44	Hostname 3. Zeichen D <sub>ascii</sub>		
11	55	Hostname 4. Zeichen U <sub>ascii</sub>		
12	4C	Hostname 5. Zeichen L <sub>ascii</sub>		
13	2D	Hostname 6. Zeichen _ <sub>ascii</sub>		
14	35	Hostname 7. Zeichen 5 <sub>ascii</sub>		
15	39	Hostname 8. Zeichen 9 <sub>ascii</sub>		
16	33	Hostname 9. Zeichen 3 <sub>ascii</sub>		
17	20	Hostname 10. Zeichen [Leer] <sub>ascii</sub>		
18	20	Hostname 11. Zeichen [Leer] <sub>ascii</sub>		
19	20	Hostname 12. Zeichen [Leer] <sub>ascii</sub>		
20	20	Hostname 13. Zeichen [Leer] <sub>ascii</sub>		
21	20	Hostname 14. Zeichen [Leer] <sub>ascii</sub>		
22	20	Hostname 15. Zeichen [Leer] <sub>ascii</sub>		
23	20	Hostname 16. Zeichen [Leer] <sub>ascii</sub>		
24	3F	IP-Adresse 4. Byte Dezimalwert 63		
25	0	IP-Adresse 3. Byte Dezimalwert 0		
26	A8	IP-Adresse 2. Byte Dezimalwert 168		
27	C0	IP-Adresse 1. Byte Dezimalwert 192		

28	00	Subnetmaske 4.Byte Dezimalwert 0		
29	FF	Subnetmaske 3.Byte Dezimalwert 255		
30	FF	Subnetmaske 2.Byte Dezimalwert 255		
31	FF	Subnetmaske 1.Byte Dezimalwert 255		
32	01	Gateway 4.Byte Dezimalwert 1		
33	00	Gateway 3.Byte Dezimalwert 0		
34	A8	Gateway 2.Byte Dezimalwert 168		
35	C0	Gateway 1.Byte Dezimalwert 192		
36	01	Primary DNS 4.Byte Dezimalwert 1		
37	00	Primary DNS 3.Byte Dezimalwert 0		
38	A8	Primary DNS 2.Byte Dezimalwert 168		
39	C0	Primary DNS 1.Byte Dezimalwert 192		
40	73	Secondary DNS 4.Byte Dezimalwert 115		
41	97	Secondary DNS 3.Byte Dezimalwert 151		
42	ED	Secondary DNS 2.Byte Dezimalwert 237		
43	D9	Secondary DNS 1.Byte Dezimalwert 217		
44	0w 00 (DHCP disable) 01 (DHCP enable)	DCHP-Client Konfiguration		
45	00	reserviert		
46	00	reserviert		
47	00	reserviert		

## Lesen Netzwerkkonfigurationen

Dieser Befehl ermöglicht das Lesen aller Netzwerkkonfigurationen wie IP-Adresse, Subnetzmaske, Hostname, Gateway, DNS-Adressen, die Einstellung der DHCP-Client-Funktion sowie der MAC-Adresse.

Beispiel: Lesen der Netzwerkkonfigurationen

Hostname = „EXDUL-593“, IP = 192.168.0.63, Subnetzmaske = 255.255.255.0,

Gateway = 192.168.0.1, Primary DNS = 192.168.0.1, Secondary DNS = 217.237.151.115

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	0C	Befehlscode 1. Byte	0C	Befehlscode 1. Byte
1	00	Befehlscode 2. Byte	00	Befehlscode 2. Byte
2	08	Befehlscode 3. Byte	08	Befehlscode 3. Byte
3	01	Längenbyte → 4 Byte	0C	Längenbyte → 48 Byte
4	00	reserviert	45	Hostname 1. Zeichen E <sub>ASCII</sub>
5	00	reserviert	58	Hostname 2. Zeichen X <sub>ASCII</sub>
6	00	reserviert	44	Hostname 3. Zeichen D <sub>ASCII</sub>
7	01	Lesefunktion	55	Hostname 4. Zeichen U <sub>ASCII</sub>
8			4C	Hostname 5. Zeichen L <sub>ASCII</sub>
9			2D	Hostname 6. Zeichen - <sub>ASCII</sub>
10			35	Hostname 7. Zeichen 5 <sub>ASCII</sub>
11			39	Hostname 8. Zeichen 9 <sub>ASCII</sub>
12			33	Hostname 9. Zeichen 3 <sub>ASCII</sub>
13			20	Hostname 10. Zeichen [Leer] <sub>ASCII</sub>
14			20	Hostname 11. Zeichen [Leer] <sub>ASCII</sub>
15			20	Hostname 12. Zeichen [Leer] <sub>ASCII</sub>
16			20	Hostname 13. Zeichen [Leer] <sub>ASCII</sub>
17			20	Hostname 14. Zeichen [Leer] <sub>ASCII</sub>
18			20	Hostname 15. Zeichen [Leer] <sub>ASCII</sub>
19			20	Hostname 16. Zeichen [Leer] <sub>ASCII</sub>
20			3F	IP-Adresse 4. Byte Dezimalwert 63
21			0	IP-Adresse 3. Byte Dezimalwert 0
22			A8	IP-Adresse 2. Byte Dezimalwert 168
23			C0	IP-Adresse 1. Byte Dezimalwert 192
24			00	Subnetzmaske 4. Byte Dezimalwert 0
25			FF	Subnetzmaske 3. Byte Dezimalwert 255
26			FF	Subnetzmaske 2. Byte Dezimalwert 255
27			FF	Subnetzmaske 1. Byte Dezimalwert 255

28			01	Gateway 4.Byte Dezimalwert 1
29			00	Gateway 3.Byte Dezimalwert 0
30			A8	Gateway 2.Byte Dezimalwert 168
31			C0	Gateway 1.Byte Dezimalwert 192
32			01	Primary DNS 4.Byte Dezimalwert 1
33			00	Primary DNS 3.Byte Dezimalwert 0
34			A8	Primary DNS 2.Byte Dezimalwert 168
35			C0	Primary DNS 1.Byte Dezimalwert 192
36			73	Secondary DNS 4.Byte Dezimalwert 115
37			97	Secondary DNS 3.Byte Dezimalwert 151
38			ED	Secondary DNS 2.Byte Dezimalwert 237
39			D9	Secondary DNS 1.Byte Dezimalwert 217
40			0w 00 (DHCP disable) 01 (DHCP enable)	DCHP-Client Konfiguration
41			00	reserviert
42			00	reserviert
43			00	reserviert
44				reserviert
45				reserviert
46			00	MAC-Adresse 6.Zeichen
47			00	MAC-Adresse 5.Zeichen
48			00	MAC-Adresse 4.Zeichen
49			3E	MAC-Adresse 3.Zeichen
50			B4	MAC-Adresse 2.Zeichen
51			D4	MAC-Adresse 1.Zeichen

## Schreiben der Securitykonfiguration

Das EXDUL-Modul stellt für die sichere Kommunikation einen Passwortschutz zur Verfügung. Ist dieser aktiviert, so muss bei jedem Transfer zum Modul das richtige Passwort gesendet werden. Im Default-Zustand ist dieser deaktiviert

Securitykonfiguration	Security-Byte
Passwort deaktiviert	0
Passwort aktiviert	1

Beispiel: Schreiben der Securitykonfiguration

Byte	Senden	Empfangen	Beschreibung
0	0C	0C	Befehlscode 1. Byte
1	00	00	Befehlscode 2. Byte
2	0C	0C	Befehlscode 3. Byte
3	01	01	Längenbyte → 20 Byte
4	00 (Passwort deaktiviert) 01 (Passwort aktiviert)		Security-Byte
5	00		reserviert
6	00		reserviert
7	00		Schreibfunktion Securitykonfiguration

## Lesen der Securitykonfiguration

Das EXDUL-Modul stellt für die sichere Kommunikation einen Passwortschutz zur Verfügung. Ist dieser aktiviert, so muss bei jedem Transfer zum Modul das richtige Passwort gesendet werden. Im Default-Zustand ist dieser deaktiviert.

Securitykonfiguration	Security-Byte
Passwort deaktiviert	0
Passwort aktiviert	1

Beispiel: Lesen der Securitykonfiguration

Byte	Senden	Empfangen	Beschreibung
0	0C	0C	Befehlscode 1. Byte
1	00	00	Befehlscode 2. Byte
2	0C	0C	Befehlscode 3. Byte
3	01	01	Längenbyte → 20 Byte
4	00	00 (Passwort deaktiviert) 01 (Passwort aktiviert)	Security-Byte
5	00	00	reserviert
6	00	00	reserviert
7	01	00	Lesefunktion Securitykonfiguration

## Ändern des Passwortes

Das EXDUL-Modul stellt für die sichere Kommunikation einen Passwortschutz zur Verfügung. Ist dieser aktiviert, so muss bei jedem Transfer zum Modul das richtige Passwort gesendet werden. Im Default-Zustand lautet das Passwort „11111111“ in ASCII und entspricht dem der Webpage. Mit dieser Funktion kann das Passwort abgeändert werden.

Beispiel: Ändern des Passwortes in „EXDUL593“

Byte	Senden	Empfangen	Beschreibung
0	0C	0C	Befehlscode 1. Byte
1	00	00	Befehlscode 2. Byte
2	0D	0D	Befehlscode 3. Byte
3	02	00	Längenbyte → 20 Byte
4	45 <sub>hex</sub>		Passwort 1. Zeichen E <sub>ascii</sub>
5	58 <sub>hex</sub>		Passwort 2. Zeichen X <sub>ascii</sub>
6	44 <sub>hex</sub>		Passwort 3. Zeichen D <sub>ascii</sub>
7	55 <sub>hex</sub>		Passwort 4. Zeichen U <sub>ascii</sub>
8	4C <sub>hex</sub>		Passwort 5. Zeichen L <sub>ascii</sub>
9	35 <sub>hex</sub>		Passwort 6. Zeichen 5 <sub>ascii</sub>
10	39 <sub>hex</sub>		Passwort 7. Zeichen 9 <sub>ascii</sub>
11	33 <sub>hex</sub>		Passwort 8. Zeichen 3 <sub>ascii</sub>



## Optokopplerausgang lesen

Dieser Befehl ermöglicht das Einlesen des aktuellen Zustands des Optokopplerausgangs.

Optokoppler durchgeschaltet = 1, Optokoppler nicht durchgeschaltet = 0

Beispiel: Auslesen des Optokopplerausgangszustands

Gesendet wird ein 8Byte langer Block und empfangen ein 8Byte langer Block mit dem Zustand des Optokopplerausgangs

Ausgangskanal	DOUT0
Schraubklemme	1
Schaltzustand	1

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	08	Befehlscode 1. Byte	08	Befehlscode 1. Byte
1	00	Befehlscode 2. Byte	00	Befehlscode 2. Byte
2	00	Befehlscode 3. Byte	00	Befehlscode 3. Byte
3	01 (→ 4Byte)	Längenbyte	01 (→ 4Byte)	Längenbyte
4	01	r/w Byte (1→ lesen)	01	Read-Funktion
5	00	reserviert	01	Zustand Optokopplerausgang
6	00	reserviert	00	reserviert
7	00	reserviert	00	reserviert

## Optokopplerausgang schreiben

Dieser Befehl ermöglicht dem Anwender, den Ausgangsoptokoppler zu sperren oder durchzuschalten.

Optokoppler durchgeschaltet = 1, Optokoppler nicht durchgeschaltet = 0

Beispiel: Ausgabe des Zustands (0x01<sub>hex</sub>) an dem Optokopplerausgang (= DOUT0 durchgeschaltet)

Gesendet wird ein 8Byte langer Block und empfangen ein 4Byte Block als Bestätigung

Ausgangskanal	DOUT0
Schraubklemme	19
Schaltzustand	0

Byte	Senden	Empfangen	Beschreibung
0	08	08	Befehlscode 1. Byte
1	00	00	Befehlscode 2. Byte
2	00	00	Befehlscode 3. Byte
3	01 (→ 4Byte)	00	Längenbyte
4	00		r/w Byte (0→ schreiben)
5	01		Optokopplerzustand
6	00		reserviert
7	00		reserviert

## Optokopplerausgänge einzeln schreiben

Dieser Befehl ermöglicht dem Anwender, einzelne Ausgangsoptokoppler zu sperren oder durchzuschalten. Der Befehl wurde aus Kompatibilitätsgründen hinzugefügt, hat in diesem Fall jedoch die selbe Wirkung wie der Befehl Optokopplerausgang schreiben.

Optokoppler durchgeschaltet = 1, Optokoppler nicht durchgeschaltet = 0

Beispiel: DOUT0 soll unabhängig der anderen Ausgänge geschaltet werden  
Gesendet wird ein 8Byte langer Block und empfangen ein 4Byte Block als Bestätigung

Ausgangskanal	DOUT0
Schraubklemme	19
Kanal-Index	0

Byte	Senden	Empfangen	Beschreibung
0	08	08	Befehlscode 1. Byte
1	00	00	Befehlscode 2. Byte
2	00	00	Befehlscode 3. Byte
3	01 (→ 4Byte)	00	Längenbyte
4	02		r/w (2→ einzeln schreiben)
5	00		Kanal-Index = 0
6	01		Optokopplerzustand
7	00		reserviert

## Optokopplereingang lesen

Dieser Befehl ermöglicht das Einlesen des aktuellen Zustands an dem Optokopplereingang.

Beispiel: Einlesen des Zustands an dem Optokopplereingang

Gesendet wird ein 4Byte langer Block und empfangen ein 8Byte langer Block mit den Zuständen.

Voraussetzung für diese Beispiele ist das Anlegen des Eingangspegels HIGH (hier 0x01) am Eingang:

(0 = LOW = 0...3V, 1 = HIGH = 10..30V)

Ausgangskanal	DIN0
Schraubklemme	21
Schaltzustand	1

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	08	Befehlscode 1. Byte	08	Befehlscode 1. Byte
1	00	Befehlscode 2. Byte	00	Befehlscode 2. Byte
2	01	Befehlscode 3. Byte	01	Befehlscode 3. Byte
3	00	Längenbyte	01 (→ 4Byte)	Längenbyte
4			01	Zustand Optokopplereingänge LB (DIN0)
5			00	Zustand Optokopplereingänge HB reserviert
6			00	reserviert
7			00	reserviert

## Zähler

Dieser Befehl ermöglicht den Zugriff auf die Zähler. So kann der Zähler gestartet, gestoppt, resettet und gelesen werden. Zudem besteht die Möglichkeit, das Overflow-Flag einzulesen und rückzusetzen.

Code	Zähler-Befehlscode
00	Zähler starten
01	Zähler stoppen
02	Zähler resettet
03	Zählerstand lesen
04	reserviert
05	Overflow-Flag lesen
06	Overflow-Flag rücksetzen

Beispiel für Zähler0:

### Zähler0 Start / Stop / Reset

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	09	Befehlscode 1. Byte	09	Befehlscode 1. Byte
1	00	Befehlscode 2. Byte	00	Befehlscode 2. Byte
2	00	Befehlscode 3. Byte	00	Befehlscode 3. Byte
3	01	Längenbyte	01	Längenbyte
4	bb 00 01 02	Zähler Befehlscode Zähler0 starten Zähler0 stoppen Zähler0 resettet	bb	Zähler Befehlscode
5	00	reserviert	00	reserviert
6	00	reserviert	00	reserviert
7	00	reserviert	00	reserviert

## Zähler0 lesen

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	09	Befehlscode 1. Byte	09	Befehlscode 1. Byte
1	00	Befehlscode 2. Byte	00	Befehlscode 2. Byte
2	00	Befehlscode 3. Byte	00	Befehlscode 3. Byte
3	01	Längenbyte	02 (→ 8Byte)	Längenbyte
4	03	Zähler Befehlscode	03	Zähler Befehlscode
5	00	reserviert	00	reserviert
6	00	reserviert	00	reserviert
7	00	reserviert	00	reserviert
8			ww	Zählerstand Byte0
9			ww	Zählerstand Byte1
10			ww	Zählerstand Byte2
11			ww	Zählerstand Byte3

Zählerstand = Zählerstand Byte3 \* 0x1000000 + Zählerstand Byte2 \* 0x10000 + Zählerstand Byte1 \* 0x100 + Zählerstand Byte0

## Overflow-Flag Zähler0 lesen

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	09	Befehlscode 1. Byte	09	Befehlscode 1. Byte
1	00	Befehlscode 2. Byte	00	Befehlscode 2. Byte
2	00	Befehlscode 3. Byte	00	Befehlscode 3. Byte
3	01	Längenbyte	02 (→ 8Byte)	Längenbyte
4	05	Zähler Befehlscode Overflow-Flag lesen	05	Zähler Befehlscode Overflow-Flag lesen
5	00	reserviert	00	reserviert
6	00	reserviert	00	reserviert
7	00	reserviert	0f	Overflow-Flag

## Overflow-Flag Zähler0 rücksetzen

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	09	Befehlscode 1. Byte	09	Befehlscode 1. Byte
1	00	Befehlscode 2. Byte	00	Befehlscode 2. Byte
2	00	Befehlscode 3. Byte	00	Befehlscode 3. Byte
3	01	Längenbyte	01 (→ 4Byte)	Längenbyte
4	06	Zähler Befehlscode Overflow-Flag rücksetzen	06	Zähler Befehlscode Overflow-Flag rücksetzen
5	00	reserviert	00	reserviert
6	00	reserviert	00	reserviert
7	00	reserviert	00	reserviert

## Temperaturmesseinheit Temperatur-/Widerstandsmessung

Dieser Befehl ermöglicht das Messen des Temperatursensors mit der gewünschten Temperaturmesseinheit. Dabei kann der gemessene Wert als Temperatur in °C \* 100 oder als Widerstand (nur bei PT100) übergeben werden.

Beispiel: Temperaturmessung an Messeinheit TIN1

Gesendet wird ein 8Byte langer Block und empfangen ein 12Byte langer Block mit der Temperatur.

Messart	Messart-Byte
Temperaturmessung	0
Widerstandsmessung	1

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	0A	Befehlscode 1. Byte	0A	Befehlscode 1. Byte
1	04	Befehlscode 2. Byte	04	Befehlscode 2. Byte
2	00	Befehlscode 3. Byte	00	Befehlscode 3. Byte
3	01	Längenbyte	02 (→ 8Byte)	Längenbyte
4	01	Messeinheitsindex	00	Messeinheitsindex
5	00	Messart-Byte (=Temperaturmessung)	00	Messart-Byte (=Temperaturmessung)
6	00	reserviert	00	reserviert
7	00	reserviert	00	reserviert
8			temp0	Temperatur Byte0
9			temp1	Temperatur Byte1
10			temp2	Temperatur Byte2
11			temp3	Temperatur Byte3

Temperatur = (integer)(temp3\*0x100000 + temp2\*0x10000 + temp1\*0x100 + temp0) [°C \*100]

## Temperaturmesseinheit Fehlerüberprüfung

Dieser Befehl führt einen Fehlertest bei der gewünschten Messeinheit durch.

Fehler-Bit	mögliche Fehlerursache wenn gesetzt	Fehlerbeschreibung
D7	reserviert	
D6	reserviert	
D5	Fehler bei der Verdrahtung	
D4	Fehler bei der Verdrahtung	
D3	Fehler bei der Verdrahtung	
D2	Overvoltage oder Undervoltage	evtl. externe Spannung eingespeist
D1	reserviert	
D0	reserviert	

Beispiel: Fehlerüberprüfung an Messeinheit TIN1

Gesendet wird ein 8Byte langer Block und empfangen ein 12Byte langer Block mit dem Fehlerbyte.

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	0A	Befehlscode 1. Byte	0A	Befehlscode 1. Byte
1	04	Befehlscode 2. Byte	04	Befehlscode 2. Byte
2	01	Befehlscode 3. Byte	01	Befehlscode 3. Byte
3	01	Längenbyte	02 (→ 8Byte)	Längenbyte
4	01	Messeinheitsindex	00	Messeinheitsindex
5	00	reserviert	00	reserviert
6	00	reserviert	00	reserviert
7	00	reserviert	00	reserviert
8			ee	Error-Byte
9			00	reserviert
10			00	reserviert
11			00	reserviert

## Temperaturmesseinheit Sensortyp einstellen

Mit diesem Befehl kann der Sensortyp einer jeden Messeinheit bestimmt werden. Zusätzlich muss der entsprechende Jumper gesetzt werden. Im Werkzustand ist der PT100 ausgewählt.

Typbyte	mögliche Fehlerursache wenn gesetzt
0x00 (default)	PT100
0x01	PT1000

Beispiel: Sensortyp PT1000 an Messeinheit TIN1 einstellen

Gesendet wird ein 8Byte langer Block und empfangen ein 8Byte langer Block mit der Temperatur.

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	0A	Befehlscode 1. Byte	0A	Befehlscode 1. Byte
1	04	Befehlscode 2. Byte	04	Befehlscode 2. Byte
2	08	Befehlscode 3. Byte	08	Befehlscode 3. Byte
3	01	Längenbyte	01 (→ 4Byte)	Längenbyte
4	01	Messeinheitsindex	00	Messeinheitsindex
5	00	reserviert	00	reserviert
6	01	Typbyte	00	reserviert
7	00	reserviert	00	reserviert



## Temperaturmesseinheit obere Schwellentemperatur einstellen

Mit diesem Befehl kann die obere Schwelle der Temperaturmesseinheit eingestellt werden. Die Schwelle kann in der programmierbaren Logik verwendet werden

Beispiel: Obere Schwellentemperatur in Messeinheit TIN1 einstellen

Gesendet wird ein 12Byte langer Block und empfangen ein 8Byte langer Block mit der Temperatur.

Temperatur = (integer)(temp3\*0x1000000 + temp2\*0x10000 + temp1\*0x100 + temp0) [°C \*100]

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	0A	Befehlscode 1. Byte	0A	Befehlscode 1. Byte
1	04	Befehlscode 2. Byte	04	Befehlscode 2. Byte
2	09	Befehlscode 3. Byte	09	Befehlscode 3. Byte
3	02	Längenbyte	01 (→ 4Byte)	Längenbyte
4	01	Messeinheitsindex	01	Messeinheitsindex
5	00	reserviert	00	reserviert
6	00	reserviert	00	reserviert
7	00	reserviert	00	reserviert
8	temp0	Temperatur Byte0		
9	temp1	Temperatur Byte1		
10	temp2	Temperatur Byte2		
11	temp3	Temperatur Byte3		

## Temperaturmesseinheit untere Schwellentemperatur einstellen

Mit diesem Befehl kann die untere Schwelle der Temperaturmesseinheit eingestellt werden. Die Schwelle kann in der programmierbaren Logik verwendet werden

Beispiel: Untere Schwellentemperatur in Messeinheit TIN1 einstellen

Gesendet wird ein 12Byte langer Block und empfangen ein 8Byte langer Block mit der Temperatur.

Temperatur = (integer)(temp3\*0x1000000 + temp2\*0x10000 + temp1\*0x100 + temp0) [°C \*100]

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	0A	Befehlscode 1. Byte	0A	Befehlscode 1. Byte
1	04	Befehlscode 2. Byte	04	Befehlscode 2. Byte
2	0A	Befehlscode 3. Byte	0A	Befehlscode 3. Byte
3	02	Längenbyte	01 (→ 4Byte)	Längenbyte
4	01	Messeinheitsindex	01	Messeinheitsindex
5	00	reserviert	00	reserviert
6	00	reserviert	00	reserviert
7	00	reserviert	00	reserviert
8	temp0	Temperatur Byte0		
9	temp1	Temperatur Byte1		
10	temp2	Temperatur Byte2		
11	temp3	Temperatur Byte3		

## Temperaturmesseinheit kalibrieren/abgleichen

Mit diesem Befehl kann die Kalibrierung bzw. der Abgleich der Messeinheit gestartet werden. Zuvor muss sowohl in der Software als auch an der Hardware der gewünschte Sensortyp ausgewählt werden.

Beispiel: Kalibrierung der Messeinheit TIN1

Gesendet wird ein 8Byte langer Block und empfangen ein 8Byte langer Block mit der Temperatur.

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	0A	Befehlscode 1. Byte	0A	Befehlscode 1. Byte
1	FF	Befehlscode 2. Byte	FF	Befehlscode 2. Byte
2	F7	Befehlscode 3. Byte	F7	Befehlscode 3. Byte
3	01	Längenbyte	01 (→ 4Byte)	Längenbyte
4	01	Messeinheitsindex	01	Messeinheitsindex
5	00	reserviert	00	reserviert
6	00	reserviert	00	reserviert
7	00	reserviert	00	reserviert

## Watchdog-Timer

Dieser Befehl ermöglicht den Zugriff auf die Zähler. So kann der Zähler gestartet, gestoppt, resettet und gelesen werden. Zudem besteht die Möglichkeit, das Overflow-Flag einzulesen und rückzusetzen.

Code	Zähler-Befehlscode
00	Watchdog-Timer starten
01	Watchdog-Timer stoppen
02	Watchdog-Timer resettet
03	WDT Periodendauer setzen

## Watchdog-Timer Start / Stop / Reset

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	0C	Befehlscode 1. Byte	0C	Befehlscode 1. Byte
1	00	Befehlscode 2. Byte	01	Befehlscode 2. Byte
2	01	Befehlscode 3. Byte	01	Befehlscode 3. Byte
3	01	Längenbyte	01	Längenbyte
4	bb 00 01 02 03	WDT Befehlscode WDT starten WDT stoppen WDT resettet WDT Periodendauer setzen	bb	Zähler Befehlscode
5	00	reserviert	00	reserviert
6	00	reserviert	00	reserviert
7	00	reserviert	00	reserviert

## Watchdog-Timer Periodendauer in ms einstellen

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	0C	Befehlscode 1. Byte	0C	Befehlscode 1. Byte
1	00	Befehlscode 2. Byte	01	Befehlscode 2. Byte
2	01	Befehlscode 3. Byte	01	Befehlscode 3. Byte
3	02	Längenbyte	01	Längenbyte
4	03	WDT Periodendauer setzen	bb	WDT Befehlscode
5	00	reserviert	00	reserviert
6	00	reserviert	00	reserviert
7	00	reserviert	00	reserviert
8	pp <sub>0</sub>	Periodendauer Byte0		
9	pp <sub>1</sub>	Periodendauer Byte1		
10	pp <sub>2</sub>	Periodendauer Byte2		
11	pp <sub>3</sub>	Periodendauer Byte3		

Periodendauer = Byte3 \* 0x1000000 + Byte2 \* 0x10000 + Byte1 \* 0x100 + Byte0 [ms]

## Initialisierung eines programmierbaren Logikzweiges

Dieser Befehl ermöglicht das Initialisieren eines der programmierbaren Logikzweige.

Code	Eingangsfunktion
00	NONE
01	TRUE (logische 1)
02	FALSE (logische 0)
03 .. 15	reserviert
16	Zustand DIN0
17 .. 31	reserviert
32	Steigende Flanke DIN0 (event)
33 .. 63	reserviert
64	Obere Temperaturschwelle überschritten TIN0
65	Obere Temperaturschwelle überschritten TIN1
66	Obere Temperaturschwelle überschritten TIN2
67	Obere Temperaturschwelle überschritten TIN3
68	Obere Temperaturschwelle überschritten TIN4
69	Obere Temperaturschwelle überschritten TIN5
70 .. 71	reserviert
72	Obere Temperaturschwelle überschritten TIN0 (event)
73	Obere Temperaturschwelle überschritten TIN1 (event)
74	Obere Temperaturschwelle überschritten TIN2 (event)
75	Obere Temperaturschwelle überschritten TIN3 (event)
76	Obere Temperaturschwelle überschritten TIN4 (event)
77	Obere Temperaturschwelle überschritten TIN5 (event)
78 .. 79	reserviert
80	Untere Temperaturschwelle unterschritten TIN0
81	Untere Temperaturschwelle unterschritten TIN1
82	Untere Temperaturschwelle unterschritten TIN2
83	Untere Temperaturschwelle unterschritten TIN3
84	Untere Temperaturschwelle unterschritten TIN4
85	Untere Temperaturschwelle unterschritten TIN5
86 .. 87	reserviert
88	Untere Temperaturschwelle unterschritten TIN0 (event)
89	Untere Temperaturschwelle unterschritten TIN1 (event)
90	Untere Temperaturschwelle unterschritten TIN2 (event)
91	Untere Temperaturschwelle unterschritten TIN3 (event)
92	Untere Temperaturschwelle unterschritten TIN4 (event)
93	Untere Temperaturschwelle unterschritten TIN5 (event)

Code	Gatterfunktion
00	UND-Verknüpfung
01	ODER-Verknüpfung

Code	Ausgangsfunktion
00	NONE (Zweig deaktiviert)
04	Send Message1 (event)
05	Send Message2 (event)
06	Send Message3 (event)
07	Send Message4 (event)
16	Write DOUT0
32	Set DOUT0 (event)
48	Clear DOUT0 (event)
64	Toggle DOUT0 (event)

Beispiel: Initialisierung des ersten Logikzweiges

Dabei soll bei jeder steigenden Flanke am Optokopplereingang DIN0 (event) die Message 1 an den PC geschickt werden. (IN0 = 32, IN1 = TRUE, IN2 = TRUE, IN3 = TRUE, Gate = 0, OUT = 4)

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	0C	Befehlscode 1. Byte	0C	Befehlscode 1. Byte
1	02	Befehlscode 2. Byte	02	Befehlscode 2. Byte
2	10	Befehlscode 3. Byte	10	Befehlscode 3. Byte
3	01	Längenbyte → 4 Byte	01	Längenbyte → 4 Byte
4	00	Lesen-/Schreibbyte	00	Lesen-/Schreibbyte
5	00	reserviert	00	reserviert
6	00	reserviert	00	reserviert
7	01	Zweigindex	00	reserviert
8	20	Funktion IN0		
9	00	reserviert		
10	00	reserviert		
11	00	reserviert		
12	01	Funktion IN1		
13	00	reserviert		
14	00	reserviert		
15	00	reserviert		
16	01	Funktion IN2		
17	00	reserviert		
18	00	reserviert		
19	00	reserviert		
20	01	Funktion IN3		
21	00	reserviert		
22	00	reserviert		
23	00	reserviert		
24	00	Gatterfunktion		
25	00	reserviert		
26	00	reserviert		
27	00	reserviert		
28	04	Ausgangsfunktion OUT0		
29	00	reserviert		
30	00	reserviert		
31	00	reserviert		

## Error-Register auslesen

Dieser Befehl ermöglicht das Auslesen der beiden Error-Register.

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	FF	Befehlscode 1. Byte	FF	Befehlscode 1. Byte
1	00	Befehlscode 2. Byte	00	Befehlscode 2. Byte
2	00	Befehlscode 3. Byte	00	Befehlscode 3. Byte
3	01	Längenbyte	03	Längenbyte
4	00	Error-Register auslesen	00	Error-Register auslesen
5	00	reserviert	00	reserviert
6	00	reserviert	00	reserviert
7	00	reserviert	00	reserviert
8			aa0	Register0 Byte0
9			aa1	Register0 Byte1
10			aa2	Register0 Byte2
11			aa3	Register0 Byte3
12			bb0	Register1 Byte0
13			bb1	Register1 Byte1
14			bb2	Register1 Byte2
15			bb3	Register1 Byte3

Error-Register0 = aa3 \* 0x1000000 + aa2 \* 0x10000 + aa1 \* 0x100 + aa0

Error-Register1 = bb3 \* 0x1000000 + bb2 \* 0x10000 + bb1 \* 0x100 + bb0

## Error-Register rücksetzen

Dieser Befehl ermöglicht das Rücksetzen der beiden Error-Register.

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	FF	Befehlscode 1. Byte	FF	Befehlscode 1. Byte
1	00	Befehlscode 2. Byte	00	Befehlscode 2. Byte
2	00	Befehlscode 3. Byte	00	Befehlscode 3. Byte
3	01	Längenbyte	01	Längenbyte
4	01	Error-Register rücksetzen	01	Error-Register rücksetzen
5	00	reserviert	00	reserviert
6	00	reserviert	00	reserviert
7	00	reserviert	00	reserviert



## Receivermodus aktivieren

Mit diesem Befehl kann der Receivermodus für die programmierbare Logik aktiviert werden. Nachdem eine zweite TCP/IP-Verbindung mit dem Modul aufgebaut wurde wird für diese Verbindung der Receivermodus aktiviert. Dieser Befehl bekommt keine Rückantwort. Ist dieser Modus eingeschaltet, funktionieren nur noch die Receiverbefehle über dieser TCP/IP-Verbindung. Sämtliche anderen Befehle, wie z. B. Eingänge lesen, müssen über die erste Verbindung erfolgen. Um den Receivermodus zu deaktivieren, muss der Befehl „Receivermodus deaktivieren“ auf der Receiver-Verbindung verwendet oder die Verbindung beendet werden.

Byte	Senden	Beschreibung
0	0C	Befehlscode 1. Byte
1	03	Befehlscode 2. Byte
2	00	Befehlscode 3. Byte
3	01	Längenbyte → 4 Byte
4	00	Receivermodus aktivieren
5	00	reserviert
6	00	reserviert
7	00	reserviert

## Receivermodus Zähler lesen (Receivermodusbefehl)

Mit diesem Befehl kann der aktuelle Stand des Receivermodus-Zählers gelesen werden. Er kann nur bei der Verbindung in Receivermodus benutzt werden. Mit ihm kann überprüft werden, ob auch alle Messages angekommen sind. Der Wert sollte um 1 größer als der Indexwert der letzten Message sein.

$$\text{Zählerindex} = dd * 0x1000000 + cc * 0x10000 + bb * 0x100 + aa$$

Byte	Senden	Beschreibung	Empfangen	Beschreibung
0	0C	Befehlscode 1. Byte	0C	Befehlscode 1. Byte
1	01	Befehlscode 2. Byte	01	Befehlscode 2. Byte
2	02	Befehlscode 3. Byte	02	Befehlscode 3. Byte
3	01	Längenbyte	02 (→ 8Byte)	Längenbyte
4	02	Receivermodus-Zähler lesen	02	Receivermodus-Zähler lesen
5	00	reserviert	00	reserviert
6	00	reserviert	00	reserviert
7	00	reserviert	00	reserviert
8			aa	Receivermodus-Index Byte0
9			bb	Receivermodus-Index Byte1
10			cc	Receivermodus-Index Byte2
11			dd	Receivermodus-Index Byte3

**Receivermodus deaktivieren (Receivermodusbefehl)**

Mit diesem Befehl kann der Receivermodus für die programmierbare Logik deaktiviert werden. Dieser Befehl bekommt keine Rückantwort und ist nur auf einer aktiven Receiver-Verbindung benutzbar.

Byte	Senden	Beschreibung
0	0C	Befehlscode 1. Byte
1	03	Befehlscode 2. Byte
2	00	Befehlscode 3. Byte
3	01	Längenbyte → 4 Byte
4	01	Receivermodus deaktivieren
5	00	reserviert
6	00	reserviert
7	00	reserviert

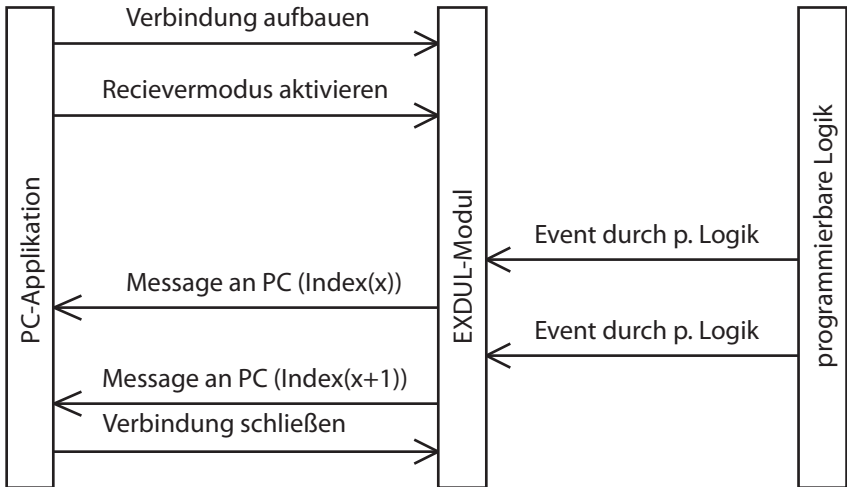
## 15.5 Receivermodus

Oft kommt es vor, dass in regelmäßigen Abständen eine Änderung z.B. an den Optokopplereingängen überprüft werden muss. Dies erfolgt normalerweise durch das Polling der entsprechenden Eingänge. Werden nun die Eingänge sehr oft abgefragt und sind auch noch mehrere Module gleichzeitig im Einsatz, so kann dies schnell zu einer erheblichen Auslastung des Netzwerkes führen. Um dies zu verhindern steht eine programmierbare Logik zu Verfügung, welche z. B. bei einer Änderung eines Optokopplereingangs eine Nachricht an den PC schicken kann. Damit entfällt die Notwendigkeit eines regelmäßigen Pollings des Optokopplereingangs.

Um diese Nachrichten der programmierbaren Logik empfangen zu können, muss eine extra TCP/IC-Verbindung (NetworkStream) mit dem Modul gestartet werden und diese mittels Befehl in den Receivermodus versetzt werden. Ist dies geschehen, kann von dem Anwenderprogramm auf Nachrichten der programmierbaren Logik gewartet werden. Dies kann je nach Bedarf und Möglichkeit synchron bzw. asynchron geschehen oder in einem externen Task ausgelagert werden.

Die Nachricht muss nicht mit einem Acknowledge im Programm bestätigt werden.

Während mit dem Modul mehrere TCP/IP-Verbindungen möglich sind, kann immer nur eine Verbindung im Receivermodus sein. Wird der Receivermodus nicht mehr benötigt, so kann diese Verbindung entweder durch Befehl in eine normale TCP/IP-Verbindung zurückgesetzt werden als auch durch einfaches Schließen der Verbindung beendet werden.



Für eine genauere Veranschaulichung des Receivermodus verweisen wir auf unsere Programmierbeispiele (z.B. Windows-C#-Beispiele)

### 15.6 Modulzugriff über LabVIEW und EXDUL.dll

Dank der EXDUL.dll kann das Modul ohne großen Aufwand in ein LabVIEW-Projekt eingebunden werden. Neben LabVIEW und der EXDUL.dll-Datei wird zudem auf dem Rechner das .NET-Framework benötigt.

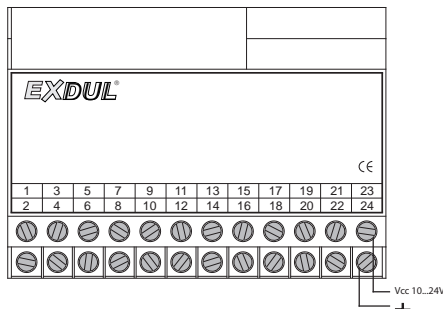
Genauere Informationen finden sie im EXDUL-LabVIEW-Tutorial.

## 16. FAQ - Problembehandlung

Nachfolgend finden Sie eine kurze Zusammenstellung der häufigsten bekannten Fehlerursachen, die während der Inbetriebnahme oder beim Zugriff auf das EXDUL-593 bzw. auf die EXDUL-Webpage auftreten können. Prüfen Sie bitte zunächst folgende Punkte, bevor Sie mit Ihrem Händler Kontakt aufnehmen.

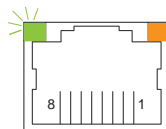
### Ist die Versorgungsspannung für das EXDUL-593 korrekt angeschlossen?

Für den Betrieb ist eine Spannungsversorgung von +10V ... +24 V DC an Klemme 23 (Vcc) und Klemme 24 (GND) nötig. Überprüfen Sie die Schraubklemmverbindungen am Modul sowie das Netzteil und die Anschlüsse am Netzteil.



### Leuchtet die Connect-LED an der RJ45-Buchse kontinuierlich grün?

Das EXDUL-593 bootet nach Anlegen der Betriebsspannung. Sobald eine stabile Ethernet-Verbindung besteht, leuchtet die Connect-LED an der 8P8C-Modularbuchse (RJ45-Buchse) des EXDUL-593 kontinuierlich grün. Sollte das nicht der Fall sein, überprüfen Sie die direkte Kabelverbindung zwischen EXDUL-593 und Computer (evtl. Crossover-Kabel notwendig), beim Netzwerkbetrieb prüfen Sie die Netzwerkleitungen zwischen EXDUL-593 und der Wand-Netzwerkbuchse, dem aktiven Ethernet-Switch oder Ethernet-Hub.



## **Besteht eine stabile Ethernet-Verbindung zwischen PC und Netzwerk?**

Prüfen Sie das Netzkabel zwischen dem Computer und der Netzwerkbuchse (RJ45-Wandsteckdose), dem aktiven Ethernet-Switch oder Ethernet-Hub. Das Ethernetkabel muss für die Ethernet-Verbindung geeignet, nicht beschädigt und beidseitig korrekt eingesteckt sein. Bei den aktuellen PCs befinden sich an der Netzwerkbuchse des Netzwerkadapters meistens zwei LEDs . Bei bestehender Verbindung mit dem Netzwerk leuchtet die grüne LED kontinuierlich. Befindet sich an der Netzwerkbuchse des PCs nur eine LED, blinkt bzw. flackert diese bei funktionierender Netzwerkverbindung.

## **Ist die verwendete Netzwerkleitung für den Anschluss geeignet?**

Beim Anschluss des EXDUL-593 an einen Switch, Hub oder PC mit Ethernet-Schnittstelle, die Auto-MDI(X) beherrscht, kann ein Standard Netzkabel (Cat. 5 oder höher) verwendet werden. Für ältere Rechner, deren Ethernet-Schnittstelle die Sende- und Empfangsleitungen nicht automatisch kreuzen, ist ein gekreuztes Netzkabel (Crossoverkabel) oder ein Crossover-Adapter notwendig.

## **Ist die Wandnetzwerkbuchse aktiv?**

Falls Sie das EXDUL-593 über eine Wandsteckdose in ein fest installiertes Netzwerk integrieren, prüfen Sie zusammen mit Ihrem Netzwerkadministrator, ob die Wandnetzwerkdose aktiv und mit einem aktiven Ethernet-Switch oder Ethernet-Hub verbunden ist.

## **Ist die Ethernet-Schnittstelle des Computers aktiviert?**

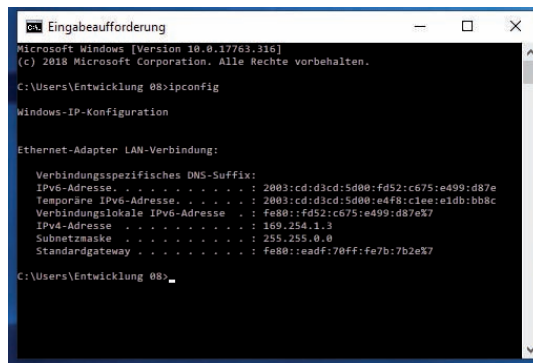
Im BIOS des PCs muss der Ethernetadapter aktiviert sein. Kontrollieren Sie im Windows Geräte Manager, ob dieser unter Netzwerkadapter gelistet ist. Der Eintrag darf nicht mit einem Ausrufezeichen versehen sein!

## **Sind die Netzwerkeinstellungen des Computers korrekt?**

Jedes aktive Gerät in einem TCP/IP-basierten Netzwerk benötigt eine eindeutige IP-Adresse, die in die Netzwerk-ID und Geräte-ID gegliedert ist. Die Subnetzmaske trennt die IP-Adresse in den Netzwerkeil (Netzpräfix, Netzwerk-ID) und den Geräteteil (Geräte-ID). In der Grundeinstellung ist das EXDUL-593 auf DHCP-Client eingestellt.

## Wie kann die Netzwerkeinstellung des PCs überprüft werden?

Die TCP/IP-Einstellungen Ihres Rechners können Sie über das Fenster Eigenschaften von Internetprotokoll Version 4 (TCP/IPv4) bzw. Status von LAN-Verbindung (siehe „Wie kann die IP-Adresse des PCs überprüft und geändert werden?“) oder über den einfachen Kommandozeilenbefehl IPCONFIG anzeigen. Wechseln Sie dazu in die MS-DOS-Eingabeaufforderung (siehe „Wie wechsele ich in die MS-DOS-Eingabeaufforderung?“), geben **ipconfig** ein und bestätigen Sie mit der **Eingabetaste** (Enter). Die Rückmeldung sollte ähnlich der folgenden Abbildung aussehen:



```
cmd Eingabeaufforderung
Microsoft Windows [Version 10.0.17763.316]
(c) 2018 Microsoft Corporation. Alle Rechte vorbehalten.

C:\Users\Entwicklung_00>ipconfig

Windows-IP-Konfiguration

Ethernet-Adapter LAN-Verbindung:

    Verbindungsspezifisches DNS-Suffix:
    IPv6-Adresse . . . . . : 2003:cd:d3cd:5d00:fd52:c675:e499:d87e
    Temporäre IPv6-Adresse . . . . . : 2003:cd:d3cd:5d00:e4f8:c1ee:e1db:bb8c
    Verbindungslokale IPv6-Adresse . . . . . : fe80::fd52:c675:e499:d87e%7
    IPv4-Adresse . . . . . : 169.254.1.3
    Subnetzmaske . . . . . : 255.255.0.0
    Standardgateway . . . . . : fe80::eadf:70ff:fe7b:7b2e%7

C:\Users\Entwicklung_00>
```

## Wie kann die IP-Adresse des PCs überprüft und geändert werden?

Windows10:

Start -> Einstellungen -> Netzwerk und Internet -> Status -> Adapteroptionen ändern -> im Fenster Netzwerkverbindungen die gewünschte LAN-Verbindung auswählen (Doppelklick oder rechte Maustaste) -> Eigenschaften -> Internetprotokoll Version 4 (TCP/IPV4) -> Eigenschaften

Hinweis: Zum Ändern der TCP/IP-Einstellungen sind Administratorrechte notwendig!

The screenshot shows the EXDUL TCP/IP Configuration web interface. At the top, there is a navigation bar with links for Home, Status, Info, Config, and Peripherie. The main heading is "TCP/IP Konfiguration". Below the heading, a note states: "Diese Seite dient zur Einstellung und Änderung der Netzwerkparameter." A warning box contains the text: "Achtung: Beachten Sie dringend die Hinweise im Handbuch, durch unkorrekte Einstellungen geht die Netzwerkkonnektivität verloren." The configuration form includes the following fields: MAC Address (04:b4:3a:00:00:00), Host Name (EXDUL-593), IP Address (192.168.180.4), Gateway (192.168.180.1), Subnet Mask (255.255.255.0), Primary DNS (192.168.180.1), and Secondary DNS (0.0.0.0). There is a checkbox for "Enable DHCP" which is checked, and a "Daten speichern" button at the bottom of the form. The footer of the page reads "EXDUL Webpage Copyright © 2021".



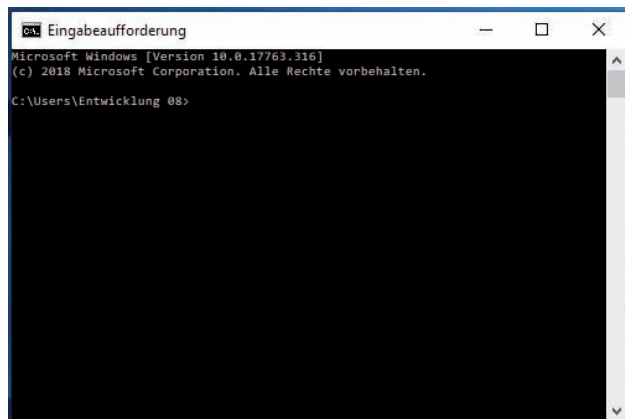
Wie wechsle ich in die MS-DOS-Eingabeaufforderung?

Windows10:

Start (Rechtsklick) -> Suchen -> **cmd** eingeben -> Bestätigung über die Eingabetaste (Enter)

oder

Start -> Windows-System -> Eingabeaufforderung



```
Eingabeaufforderung
Microsoft Windows [Version 10.0.17763.316]
(c) 2018 Microsoft Corporation. Alle Rechte vorbehalten.

C:\Users\Entwicklung 08>
```

## Ist es möglich EXDUL-5xx-Module in einem bestehenden Netzwerk zu orten und die Netzwerkdaten festzustellen?

Alle EXDUL-5xx-Module versenden in gewissen zeitlichen Abständen Erkennungssignale. Das Programm **ExdulUtility\_v2\_xx** (oder höher) wertet die Kennungsdaten aus und erstellt eine Liste mit Host-Namen, IP-Adresse und MAC-Adresse. Es eignet sich für ein einzelnes direkt am PC angeschlossenes EXDUL-5xx, wie für ein Netzwerk über Hub oder Switch mit mehreren Modulen. Für das Programm ist eine Freigabe in Ihrer Firewall erforderlich, falls diese die Kommunikation des Suchprogrammes mit den EXDUL-5xx verhindert.

## 17. Technische Daten

### **Digitaler Eingang über Optokoppler**

1 bipolarer Kanal mit galvanischer Trennung  
Überspannungsschutz-Dioden  
Eingangsspannungsbereich  
    high = 10..30 Volt  
    low = 0..3 Volt

### **Digitaler Ausgang über Optokoppler**

1 Kanal mit galvanischer Trennung über Leistungsoptokoppler  
Verpolungsschutz-Dioden  
Spannung-CE: max. 50 V  
Ausgangsstrom: max. 150mA

### **Zähler**

1 programmierbarer Zähler 32 Bit (belegt den Optokopplereingang)  
Zählfrequenz: max. 5 kHz  
Automatische Sicherung des Zählerstandes im 10kHz Takt

### **6 Temperaturmesseinheiten**

Sensortyp PT100 und PT1000 je Einheit individuell über Jumper wählbar  
3-Leiteranschluss

### **Programmierbare Logik**

Ausgänge über Verknüpfungen schalten  
Meldung an PC bei Zustandsänderung an den Eingängen senden

### **TCP/IP-Server**

Stabile und einfache Kommunikation mit einfachem Protokoll

### **Webserver**

Konfiguration  
Funktionstest  
Update der Firmware

## **Kommunikations-Watchdog-Timer**

Einstellbares Zeitfenster von 1 bis  $2^{32}$  - 1ms

## **LCD Anzeige (nur EXDUL-593E)**

Matrixanzeige mit 2 Zeilen und 16 Spalten zur Darstellung von 16 Zeichen je Zeile

Programmierbar zur Darstellung anwendungsspezifischer Daten oder als I/O-Zustandsanzeige

## **Modul-Anschlüsse**

1 \* 24polige Schraubklemmleiste

Ethernet RJ45-Buchse

## **Stromversorgung**

extern

Spannung: 10 .. 30V

Strom: typ. 100mA bei 24V

## **Ethernet-Anschlussleitung**

RJ45 Netzkabel Cat5 oder höher

## **Abmessungen**

105 mm x 89 mm x 59 mm (l x b x h)

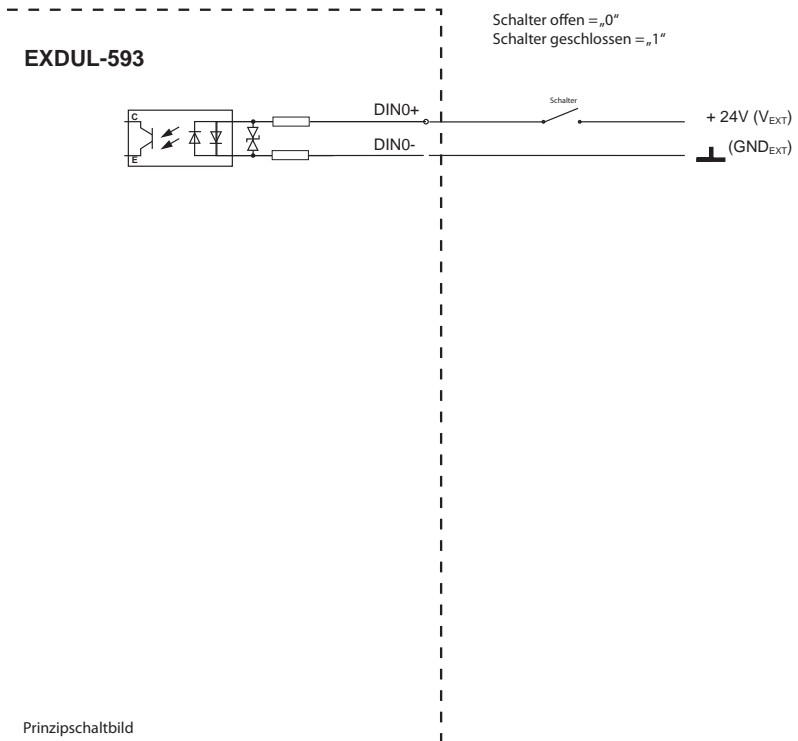
## **Gehäuse**

Isolierstoffgehäuse mit integrierter Schnapptechnik zur DIN EN-Hutschienenmontage

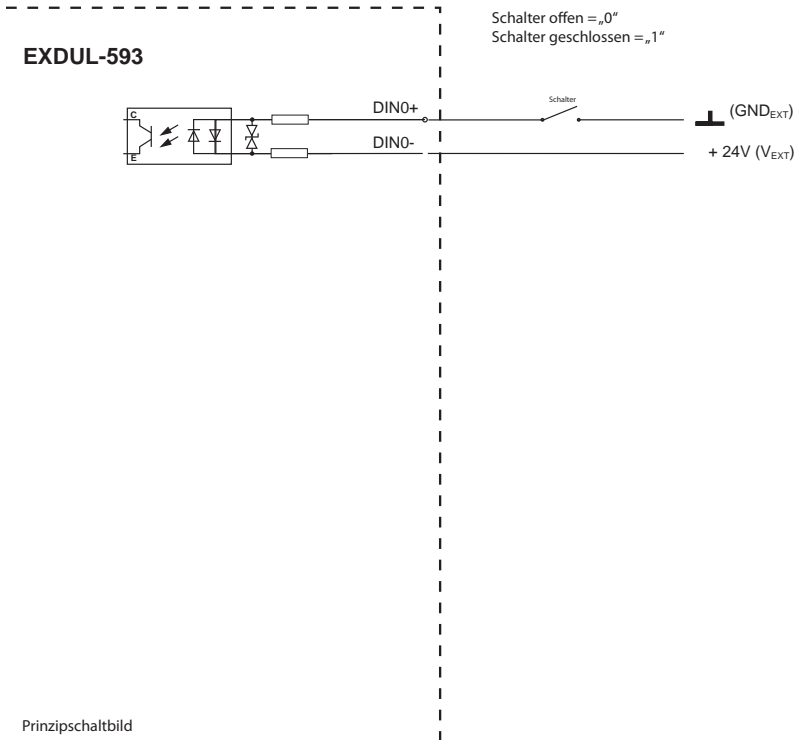
Geeignet für Aufbaumontagen, Schaltschrank- und Verteilereinbau sowie für mobile Tischeinsätze

## 18. Beschaltungsbeispiele

### 18.1 Beschaltung des Optokoppler-Eingangs

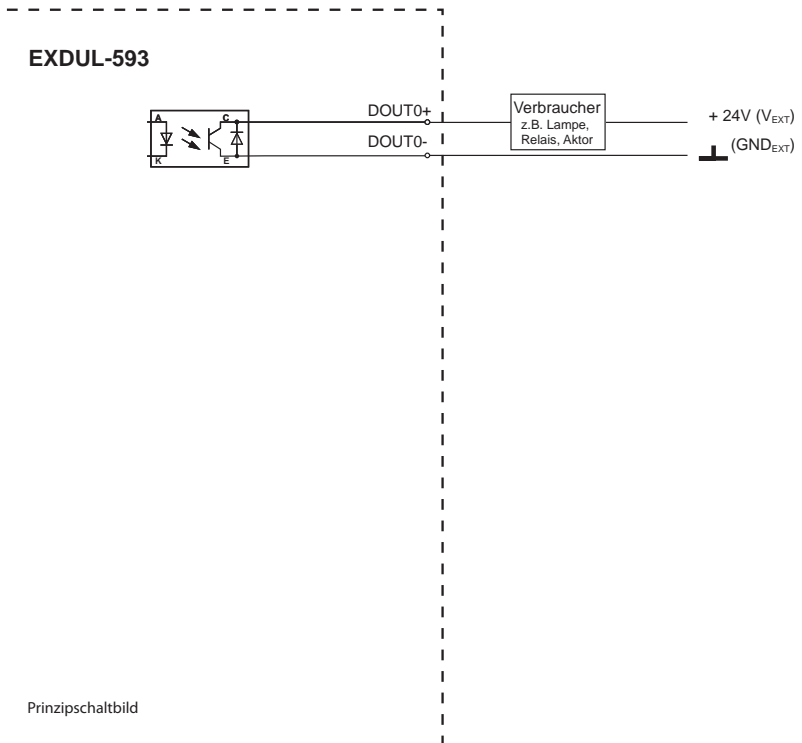


Grafik 18.1.1 Beschaltung des Optokopplereingangs



Grafik 18.1.2 Beschaltung der Optokopplereingänge revers

## 18.2 Beschaltung des Optokoppler-Ausgangs



Gratik 18.2 Beschaltung des Optokoppler-Ausgangs

## 19. ASCII-Tabelle

Hex	Dez	Binär	Zeichen
00	0	00000000	
01	1	00000001	
02	2	00000010	
03	3	00000011	
04	4	00000100	
05	5	00000101	
06	6	00000110	
07	7	00000111	
08	8	00001000	
09	9	00001001	
0A	10	00001010	
0B	11	00001011	
0C	12	00001100	
0D	13	00001101	
0E	14	00001110	
0F	15	00001111	
10	16	00010000	
11	17	00010001	
12	18	00010010	
13	19	00010011	
14	20	00010100	
15	21	00010101	
16	22	00010110	
17	23	00010111	
18	24	00011000	
19	25	00011001	
1A	26	00011010	
1B	27	00011011	
1C	28	00011100	
1D	29	00011101	
1E	30	00011110	
1F	31	00011111	
20	32	00100000	[Leer]
21	33	00100001	!
22	34	00100010	"
23	35	00100011	#
24	36	00100100	\$
25	37	00100101	%
26	38	00100110	&
27	39	00100111	'

Hex	Dez	Binär	Zeichen
28	40	00101000	(
29	41	00101001	)
2A	42	00101010	*
2B	43	00101011	+
2C	44	00101100	,
2D	45	00101101	-
2E	46	00101110	.
2F	47	00101111	/
30	48	00110000	0
31	49	00110001	1
32	50	00110010	2
33	51	00110011	3
34	52	00110100	4
35	53	00110101	5
36	54	00110110	6
37	55	00110111	7
38	56	00111000	8
39	57	00111001	9
3A	58	00111010	:
3B	59	00111011	;
3C	60	00111100	<
3D	61	00111101	=
3E	62	00111110	>
3F	63	00111111	?
40	64	01000000	@
41	65	01000001	A
42	66	01000010	B
43	67	01000011	C
44	68	01000100	D
45	69	01000101	E
46	70	01000110	F
47	71	01000111	G
48	72	01001000	H
49	73	01001001	I
4A	74	01001010	J
4B	75	01001011	K
4C	76	01001100	L
4D	77	01001101	M
4E	78	01001110	N
4F	79	01001111	O

Hex	Dez	Binär	Zeichen
50	80	01010000	P
51	81	01010001	Q
52	82	01010010	R
53	83	01010011	S
54	84	01010100	T
55	85	01010101	U
56	86	01010110	V
57	87	01010111	W
58	88	01011000	X
59	89	01011001	Y
5A	90	01011010	Z
5B	91	01011011	[
5C	92	01011100	
5D	93	01011101	]
5E	94	01011110	^
5F	95	01011111	_
60	96	01100000	`
61	97	01100001	a
62	98	01100010	b
63	99	01100011	c
64	100	01100100	d
65	101	01100101	e
66	102	01100110	f
67	103	01100111	g
68	104	01101000	h
69	105	01101001	i
6A	106	01101010	j
6B	107	01101011	k
6C	108	01101100	l
6D	109	01101101	m
6E	110	01101110	n
6F	111	01101111	o
70	112	01110000	p
71	113	01110001	q
72	114	01110010	r
73	115	01110011	s
74	116	01110100	t
75	117	01110101	u
76	118	01110110	v
77	119	01110111	w
78	120	01111000	x
79	121	01111001	y
7A	122	01111010	z
7B	123	01111011	{

Hex	Dez	Binär	Zeichen
7C	124	01111100	
7D	125	01111101	}
7E	126	01111110	
7F	127	01111111	
80	128	10000000	
81	129	10000001	
82	130	10000010	
83	131	10000011	
84	132	10000100	
85	133	10000101	
86	134	10000110	
87	135	10000111	
88	136	10001000	
89	137	10001001	
8A	138	10001010	
8B	139	10001011	
8C	140	10001100	
8D	141	10001101	
8E	142	10001110	
8F	143	10001111	
90	144	10010000	
91	145	10010001	
92	146	10010010	
93	147	10010011	
94	148	10010100	
95	149	10010101	
96	150	10010110	
97	151	10010111	
98	152	10011000	
99	153	10011001	
9A	154	10011010	
9B	155	10011011	
9C	156	10011100	
9D	157	10011101	
9E	158	10011110	
9F	159	10011111	
A0	160	10100000	
A1	161	10100001	
A2	162	10100010	
A3	163	10100011	
A4	164	10100100	
A5	165	10100101	
A6	166	10100110	
A7	167	10100111	



Hex	Dez	Binär	Zeichen
A8	168	10101000	
A9	169	10101001	
AA	170	10101010	
AB	171	10101011	
AC	172	10101100	
AD	173	10101101	
AE	174	10101110	
AF	175	10101111	
B0	176	10110000	
B1	177	10110001	
B2	178	10110010	
B3	179	10110011	
B4	180	10110100	
B5	181	10110101	
B6	182	10110110	
B7	183	10110111	
B8	184	10111000	
B9	185	10111001	
BA	186	10111010	
BB	187	10111011	
BC	188	10111100	
BD	189	10111101	
BE	190	10111110	
BF	191	10111111	
C0	192	11000000	
C1	193	11000001	
C2	194	11000010	
C3	195	11000011	
C4	196	11000100	
C5	197	11000101	
C6	198	11000110	
C7	199	11000111	
C8	200	11001000	
C9	201	11001001	
CA	202	11001010	
CB	203	11001011	
CC	204	11001100	
CD	205	11001101	
CE	206	11001110	
CF	207	11001111	
D0	208	11010000	
D1	209	11010001	
D2	210	11010010	
D3	211	11010011	

Hex	Dez	Binär	Zeichen
D4	212	11010100	
D5	213	11010101	
D6	214	11010110	
D7	215	11010111	
D8	216	11011000	
D9	217	11011001	
DA	218	11011010	
DB	219	11011011	
DC	220	11011100	
DD	221	11011101	
DE	222	11011110	
DF	223	11011111	
E0	224	11100000	
E1	225	11100001	
E2	226	11100010	
E3	227	11100011	
E4	228	11100100	
E5	229	11100101	
E6	230	11100110	
E7	231	11100111	
E8	232	11101000	
E9	233	11101001	
EA	234	11101010	
EB	235	11101011	
EC	236	11101100	
ED	237	11101101	
EE	238	11101110	
EF	239	11101111	
F0	240	11110000	
F1	241	11110001	
F2	242	11110010	
F3	243	11110011	
F4	244	11110100	
F5	245	11110101	
F6	246	11110110	
F7	247	11110111	
F8	248	11111000	
F9	249	11111001	
FA	250	11111010	
FB	251	11111011	
FC	252	11111100	
FD	253	11111101	
FE	254	11111110	
FF	255	11111111	

## 20. Produkthaftungsgesetz

### Hinweise zur Produkthaftung

Das Produkthaftungsgesetz (ProdHaftG) regelt die Haftung des Herstellers für Schäden, die durch Fehler eines Produktes verursacht werden.

Die Verpflichtung zu Schadenersatz kann schon gegeben sein, wenn ein Produkt aufgrund der Form der Darbietung bei einem nichtgewerblichen Endverbraucher eine tatsächlich nicht vorhandene Vorstellung über die Sicherheit des Produktes erweckt, aber auch wenn damit zu rechnen ist, dass der Endverbraucher nicht die erforderlichen Vorschriften über die Sicherheit beachtet, die beim Umgang mit diesem Produkt einzuhalten wären.

Es muss daher stets nachweisbar sein, dass der nichtgewerbliche Endverbraucher mit den Sicherheitsregeln vertraut gemacht wurde.

Bitte weisen Sie daher im Interesse der Sicherheit Ihre nichtgewerblichen Abnehmer stets auf Folgendes hin:

### Sicherheitsvorschriften

Beim Umgang mit Produkten, die mit elektrischer Spannung in Berührung kommen, müssen die gültigen VDE-Vorschriften beachtet werden.

Besonders sei auf folgende Vorschriften hingewiesen:

VDE0100; VDE0550/0551; VDE0700; VDE0711; VDE0860.

Sie erhalten VDE-Vorschriften beim vde-Verlag GmbH, Bismarckstraße 33, 10625 Berlin.

- \* Vor Öffnen eines Gerätes den Netzstecker ziehen oder sicherstellen, dass das Gerät stromlos ist.
- \* Bauteile, Baugruppen oder Geräte dürfen nur in Betrieb genommen werden, wenn sie vorher in ein berührungssicheres Gehäuse eingebaut wurden. Während des Einbaus müssen sie stromlos sein.
- \* Werkzeuge dürfen an Geräten, Bauteilen oder Baugruppen nur benutzt werden, wenn sichergestellt ist, dass die Geräte von der Versorgungsspannung getrennt sind und elektrische Ladungen, die in im Gerät befindlichen Bauteilen gespeichert sind, vorher entladen wurden.
- \* Spannungsführende Kabel oder Leitungen, mit denen das Gerät, das Bauteil oder die Baugruppe verbunden sind, müssen stets auf Isolationsfehler oder Bruchstellen untersucht werden. Bei Feststellen eines Fehlers in der Zuleitung muss das Gerät unverzüglich aus dem Betrieb genommen werden, bis die defekte Leitung ausgewechselt worden ist.
- \* Bei Einsatz von Bauelementen oder Baugruppen muss stets auf die strikte Einhaltung der in der zugehörigen Beschreibung genannten Kenndaten für elektrische Größen hingewiesen werden.
- \* Wenn aus den vorgelegten Beschreibungen für den nichtgewerblichen Endverbraucher nicht eindeutig hervorgeht, welche elektrischen Kennwerte für ein Bauteil gelten, so muss stets ein Fachmann um Auskunft ersucht werden.

Im Übrigen unterliegt die Einhaltung von Bau- und Sicherheitsvorschriften aller Art (VDE, TÜV, Berufsgenossenschaften usw.) dem Anwender/Käufer.

## 21. EG-Konformitätserklärung

Für die Erzeugnisse

**EXDUL-593E EDV-Nummer A-372320**  
**EXDUL-593S EDV-Nummer A-372310**

wird hiermit bestätigt, dass sie den Anforderungen der betreffenden EG-Richtlinien entsprechen. Bei Nichteinhaltung der im Handbuch angegebenen Vorschriften zum bestimmungsgemäßen Betrieb der Produkte verliert diese Erklärung Ihre Gültigkeit.

EN 5502 Klasse B  
IEC 801-2  
IEC 801-3  
IEC 801-4  
EN 50082-1  
EN 60555-2  
EN 60555-3

Diese Erklärung wird verantwortlich für den Hersteller

Messcomp Datentechnik GmbH  
Neudecker Str. 11  
83512 Wasserburg

abgegeben durch

Dipl.Ing.(FH) Hans Schnellhammer

Wasserburg, 31.01.2021



---

## Referenzsystem-Bestimmungsgemäßer Betrieb

Die Multifunktionsmodule EXDUL-593E und EXDUL-593S sind nicht selbständig betreibbare Geräte, dessen CE-Konformität nur bei gleichzeitiger Verwendung von zusätzlichen Computerkomponenten beurteilt werden kann. Die Angaben zur CE-Konformität beziehen sich deshalb ausschließlich auf den bestimmungsgemäßen Einsatz der Multifunktionsmodule in folgendem Referenzsystem:

Schaltschrank:	Vero IMRAK 3400	804-530061C 802-563424J 802-561589J
19" Gehäuse:	Vero PC-Gehäuse	145-010108L
19" Gehäuse:	Zusatzelektronik	519-112111C
Motherboard:	GA-586HX	PIV 1.55
Floppy-Controller:	auf Motherboard	
Floppy:	TEAC	FD-235HF
Grafikkarte:	Advantech	PCA-6443
Schnittstellen:	EXDUL-593E EXDUL-593S	A-372320 A-372310