

# SIEMENS

## SIMATIC Sensors

### RFID-Systeme Kommunikationsmodul RF182C

#### Betriebsanleitung

Einleitung	1
Beschreibung	2
Montieren	3
Anschließen	4
Parametrieren	5
Kommunikationsschnittstelle	6
Instandhalten und Warten	7
Diagnose	8
Fehlermeldungen	9
Beispiele/Applikationen	10
Technische Daten	11
Maßbilder	12
Anschlusskabel zum Reader / SLG	13
Bestelldaten	14
Befehls- und Quittungstelegramme	A
Adressierung der RFID-Tags	B
Schema der Übertragung der Hexadezimalen Tag-Daten über XML	C
Service & Support	D

## Rechtliche Hinweise

### Warnhinweiskonzept

Dieses Handbuch enthält Hinweise, die Sie zu Ihrer persönlichen Sicherheit sowie zur Vermeidung von Sachschäden beachten müssen. Die Hinweise zu Ihrer persönlichen Sicherheit sind durch ein Warndreieck hervorgehoben, Hinweise zu alleinigen Sachschäden stehen ohne Warndreieck. Je nach Gefährdungsstufe werden die Warnhinweise in abnehmender Reihenfolge wie folgt dargestellt.

 <b>GEFAHR</b>
bedeutet, dass Tod oder schwere Körperverletzung eintreten <b>wird</b> , wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

 <b>WARNUNG</b>
bedeutet, dass Tod oder schwere Körperverletzung eintreten <b>kann</b> , wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

 <b>VORSICHT</b>
mit Warndreieck bedeutet, dass eine leichte Körperverletzung eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

<b>VORSICHT</b>
ohne Warndreieck bedeutet, dass Sachschaden eintreten kann, wenn die entsprechenden Vorsichtsmaßnahmen nicht getroffen werden.

<b>ACHTUNG</b>
bedeutet, dass ein unerwünschtes Ergebnis oder Zustand eintreten kann, wenn der entsprechende Hinweis nicht beachtet wird.

Beim Auftreten mehrerer Gefährdungsstufen wird immer der Warnhinweis zur jeweils höchsten Stufe verwendet. Wenn in einem Warnhinweis mit dem Warndreieck vor Personenschäden gewarnt wird, dann kann im selben Warnhinweis zusätzlich eine Warnung vor Sachschäden angefügt sein.

### Qualifiziertes Personal

Das zu dieser Dokumentation zugehörige Produkt/System darf nur von für die jeweilige Aufgabenstellung **qualifiziertem Personal** gehandhabt werden unter Beachtung der für die jeweilige Aufgabenstellung zugehörigen Dokumentation, insbesondere der darin enthaltenen Sicherheits- und Warnhinweise. Qualifiziertes Personal ist auf Grund seiner Ausbildung und Erfahrung befähigt, im Umgang mit diesen Produkten/Systemen Risiken zu erkennen und mögliche Gefährdungen zu vermeiden.

### Bestimmungsgemäßer Gebrauch von Siemens-Produkten

Beachten Sie Folgendes:

 <b>WARNUNG</b>
Siemens-Produkte dürfen nur für die im Katalog und in der zugehörigen technischen Dokumentation vorgesehenen Einsatzfälle verwendet werden. Falls Fremdprodukte und -komponenten zum Einsatz kommen, müssen diese von Siemens empfohlen bzw. zugelassen sein. Der einwandfreie und sichere Betrieb der Produkte setzt sachgemäßen Transport, sachgemäße Lagerung, Aufstellung, Montage, Installation, Inbetriebnahme, Bedienung und Instandhaltung voraus. Die zulässigen Umgebungsbedingungen müssen eingehalten werden. Hinweise in den zugehörigen Dokumentationen müssen beachtet werden.

### Marken

Alle mit dem Schutzrechtsvermerk ® gekennzeichneten Bezeichnungen sind eingetragene Marken der Siemens AG. Die übrigen Bezeichnungen in dieser Schrift können Marken sein, deren Benutzung durch Dritte für deren Zwecke die Rechte der Inhaber verletzen kann.

### Haftungsausschluss

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

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b> .....	<b>5</b>
1.1	Einleitung .....	5
<b>2</b>	<b>Beschreibung</b> .....	<b>7</b>
<b>3</b>	<b>Montieren</b> .....	<b>13</b>
3.1	Einbaulage, Einbaumaße.....	13
3.2	Peripheriemodul montieren.....	14
3.3	Anschlussblock montieren .....	16
3.4	Beschriftungsschilder austauschen .....	18
3.5	RF182C demontieren.....	18
<b>4</b>	<b>Anschließen</b> .....	<b>21</b>
4.1	Anschlussblock M12, 7/8" verdrahten.....	24
4.2	Anschlussblock Push Pull verdrahten.....	28
4.3	Ethernet und Versorgungsspannung weiterschleifen .....	31
4.4	RF182C an Controller mit Ethernet-Anschluss verdrahten .....	33
4.5	RF182C mit der Funktionserde (PE) verbinden.....	34
<b>5</b>	<b>Parametrieren</b> .....	<b>37</b>
5.1	Adresszuordnung für Ethernet.....	37
5.2	Datenkommunikation zwischen Client und RF182C.....	39
5.3	RF182C im Lieferzustand .....	43
5.4	IP-Adresse vergeben .....	44
5.4.1	Überblick .....	44
5.4.2	Webserver.....	44
5.4.3	Primary Setup Tool .....	47
5.5	Troubleshooting: IP-Adresse vergeben .....	51
<b>6</b>	<b>Kommunikationsschnittstelle</b> .....	<b>53</b>
6.1	Befehlsübersicht .....	53
6.2	Projektierungsparameter des RF182C .....	54
6.3	Input-Parameter des RF182C.....	56
6.4	Befehle des Kommunikationsmoduls.....	57
6.4.1	writeTagData .....	57
6.4.2	readTagData .....	59
6.4.3	initializeTag .....	60
6.4.4	getReaderStatus .....	61
6.4.5	getTagStatus.....	62
6.4.6	setAnt .....	64
6.4.7	heartbeat.....	65

6.5	Asynchrone Telegramme .....	66
6.5.1	tagPresent .....	66
6.5.2	alarm .....	67
<b>7</b>	<b>Instandhalten und Warten.....</b>	<b>69</b>
7.1	Kommunikationsmodul RF182C tauschen.....	69
7.2	Firmware-Update .....	71
7.3	Reader-Update.....	71
<b>8</b>	<b>Diagnose .....</b>	<b>73</b>
8.1	Diagnose mittels Leuchtdioden.....	73
<b>9</b>	<b>Fehlermeldungen.....</b>	<b>77</b>
9.1	Antwort ohne Fehlereintrag.....	77
9.2	Antwort mit Fehlereintrag.....	77
9.3	Fehlermeldungen des RF182C.....	78
9.4	Diagnose über Webserver .....	84
9.4.1	I&M Datensätze speichern/auslesen .....	84
9.4.2	Kommunikationsstatus abfragen.....	85
9.4.3	Ereignis- und Telegrammübersicht.....	86
<b>10</b>	<b>Beispiele/Applikationen.....</b>	<b>87</b>
10.1	Allgemeine Grundlagen der Socketprogrammierung beispielhaft in C.....	87
10.1.1	Voraussetzungen zur Socketprogrammierung.....	87
10.1.2	Grundprinzip Client / Server.....	88
10.1.3	Wichtige Grundbefehle.....	88
10.1.4	Auszüge eines Programmierbeispiels eines Clients in C / Betriebssystem Windows.....	89
10.2	Anwenderapplikation RF182C .....	91
10.2.1	Aufbau Bedienoberfläche.....	92
10.2.2	Auszüge Beispielcode der Anwenderapplikation in C# .....	93
10.2.3	Funktionen der Applikation RF182C .....	101
10.3	Beispielapplikation für PLC nach DIN IEC 61131.....	105
<b>11</b>	<b>Technische Daten.....</b>	<b>107</b>
<b>12</b>	<b>Maßbilder.....</b>	<b>109</b>
12.1	Maßbild RF182C mit Befestigungslöchern .....	109
<b>13</b>	<b>Anschlusskabel zum Reader / SLG .....</b>	<b>111</b>
13.1	Belegung der Standardkabel.....	111
13.2	Selbst konfektionierte Kabel .....	113
<b>14</b>	<b>Bestelldaten .....</b>	<b>115</b>
<b>A</b>	<b>Befehls- und Quittungstelegramme.....</b>	<b>117</b>
<b>B</b>	<b>Adressierung der RFID-Tags .....</b>	<b>129</b>
<b>C</b>	<b>Schema der Übertragung der Hexadezimalen Tag-Daten über XML.....</b>	<b>131</b>
<b>D</b>	<b>Service &amp; Support.....</b>	<b>133</b>

# Einleitung

## 1.1 Einleitung

### Zweck dieser Betriebsanleitung

Die Informationen dieser Betriebsanleitung ermöglichen es Ihnen, das Kommunikationsmodul RF182C an einem Standard-PC oder einer SPS in Betrieb zu nehmen.

### Erforderliche Grundkenntnisse

Zum Verständnis der Betriebsanleitung sind allgemeine Kenntnisse auf dem Gebiet der Automatisierungstechnik und Identifikationssysteme erforderlich.

Außerdem benötigen Sie Grundkenntnisse der Socketprogrammierung (TCP/IP-Kommunikation über Ethernet). Die Socketprogrammierung ist abhängig von der Programmiersprache bzw. vom verwendeten Betriebssystem (Windows, Linux oder Unix).

### Gültigkeitsbereich des Handbuchs

Die Betriebsanleitung ist gültig für das Kommunikationsmodul RF182C.

### Einordnung in die Informationslandschaft

- Informationen zu den anzuschließenden Readern / SLG finden Sie im Handbuch der jeweiligen RFID-Familie.
- Spezielle Informationen zur Parametrierung der Reader RF620R/RF630R in Verbindung mit dem Kommunikationsmodul RF182C finden Sie im "Parametrierhandbuch RF620R/RF630R".

## Wegweiser

Die vorliegende Betriebsanleitung beschreibt die Hardware sowie die Kommunikationsschnittstelle des Kommunikationsmoduls RF182C. Sie besteht aus anleitenden Kapiteln und Kapiteln zum Nachschlagen (z. B. Technische Daten).

Die Betriebsanleitung beinhaltet folgende Themen:

- Anschließen des Kommunikationsmoduls RF182C
- Parametrierung und Programmierung des Kommunikationsmoduls RF182C
- Hinweise zur Diagnose
- Anzeigeelemente des Kommunikationsmoduls RF182C
- Informationen zum Instandhalten und Warten (z. B. Firmware-Update)
- Technische Daten sowie Maßbilder des Kommunikationsmoduls RF182C
- Bestelldaten

## Recycling und Entsorgung

- Das Kommunikationsmodul RF182C ist aufgrund seiner schadstoffarmen Ausrüstung recyclingfähig.
- Für ein umweltverträgliches Recycling und die Entsorgung Ihres Altgerätes wenden Sie sich an einen zertifizierten Entsorgungsbetrieb für Elektronikschrott.

# Beschreibung

# 2

## Anwendungsbereich

Das Kommunikationsmodul RF182C ist eine Baugruppe für den Betrieb von RFID-Komponenten an einem Standard-PC oder SPS via Ethernet.

### Kommunikationsmodul RF182C

Mit Anschlussblock M12, 7/8"



Mit Anschlussblock Push Pull



Beim Betrieb an einem Standard-PC beachten Sie die entsprechenden Anweisungen zur Parametrierung und Einbindung in das System.

Folgende RFID-Familien können mit dem RF182C betrieben werden (nur mit Normaladressierung):

- RF300
- RF600
- MOBY D
- MOBY U

## Merkmale

Am RF182C können bis zu 2 Reader / SLG parallel betrieben werden. Der Anwender kann parallel an 2 Readern / SLG einen Befehl starten.

Der Zugriff auf die Tag-Daten erfolgt über die physikalische Adressierung des Tags.

### Weitere Merkmale

- Schutzart IP67
- Anlagenintegration mit M12, 7/8"-Konzept oder mit Push-Pull-Konzept
- Standardisierte Ethernet-Schnittstelle
- Diagnoseunterstützung über Webserver
- Routingfähig
- Firmware-Update über Webserver
- Unterstützung von Identification&Maintenance-Datensätzen (I&M): Mechanismus zum Auslesen von Informationen über das Kommunikationsmodul und das Abspeichern von Anlageninformationen wie Funktion, Einbaudatum, Einbauort und Kommentare
- Baugruppe unterstützt SNMP

### Aufbau

Das RF182C besitzt das gleiche Gehäuse, wie das RFID-Kommunikationsmodul ASM 456 für PROFIBUS und das RFID-Kommunikationsmodul RF180C für PROFINET.

Für die Ethernet-Verbindung hat das Kommunikationsmodul RF182C einen Anschlussblock in einer der folgenden Ausführungen:

- Anschlussblock in der Ausführung M12, wahlweise mit
  - 5-poligem 7/8" Stecker (Standard) oder
  - 4-poligem 7/8" Stecker (Option)
- Anschlussblock in der Ausführung Push Pull, RJ45

Das folgende Bild zeigt den prinzipiellen Aufbau des RF182C.

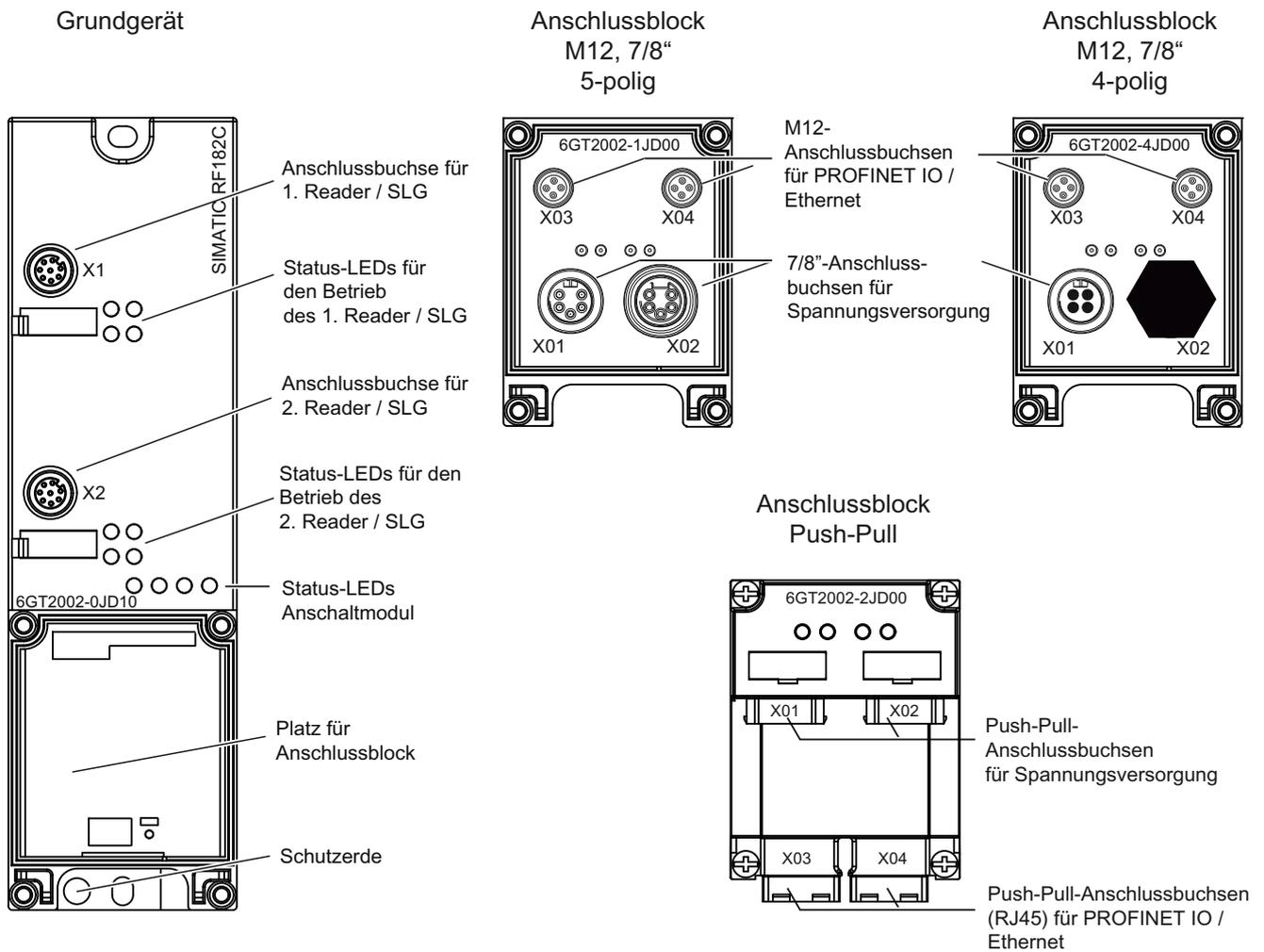


Bild 2-1 Prinzipieller Aufbau des RF182C

## Potentialaufbau

Mit dem RF182C kann ein erdfreier Aufbau einer Anlage erfolgen. Die folgende Schaltung zeigt den internen Zusammenhang der Bezugspotentiale.

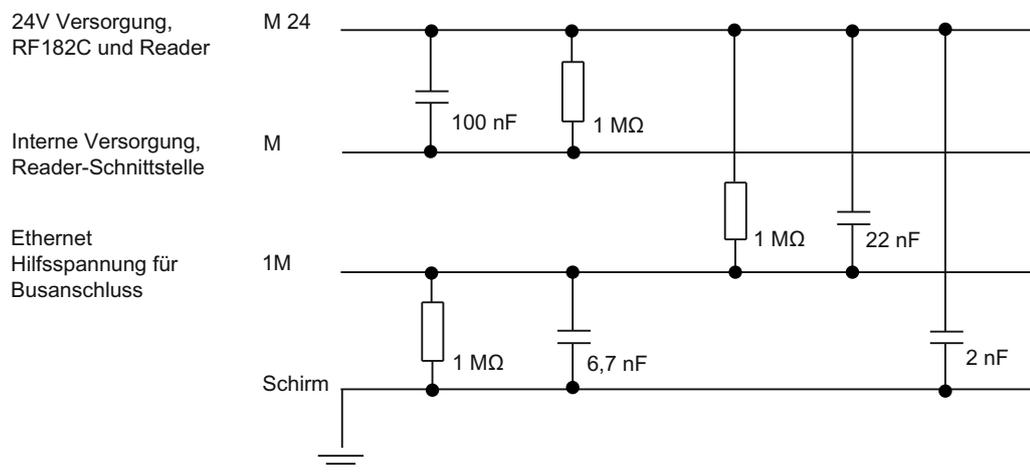


Bild 2-2 Potentialtrennung RF182C

## Integration

Das folgende Bild zeigt die Integration des RF182C mit Anschlussblock M12, 7/8" in ein Automatisierungssystem. Die Integration mit Anschlussblock Push Pull ist genauso wie bei der M12, 7/8"-Technik.

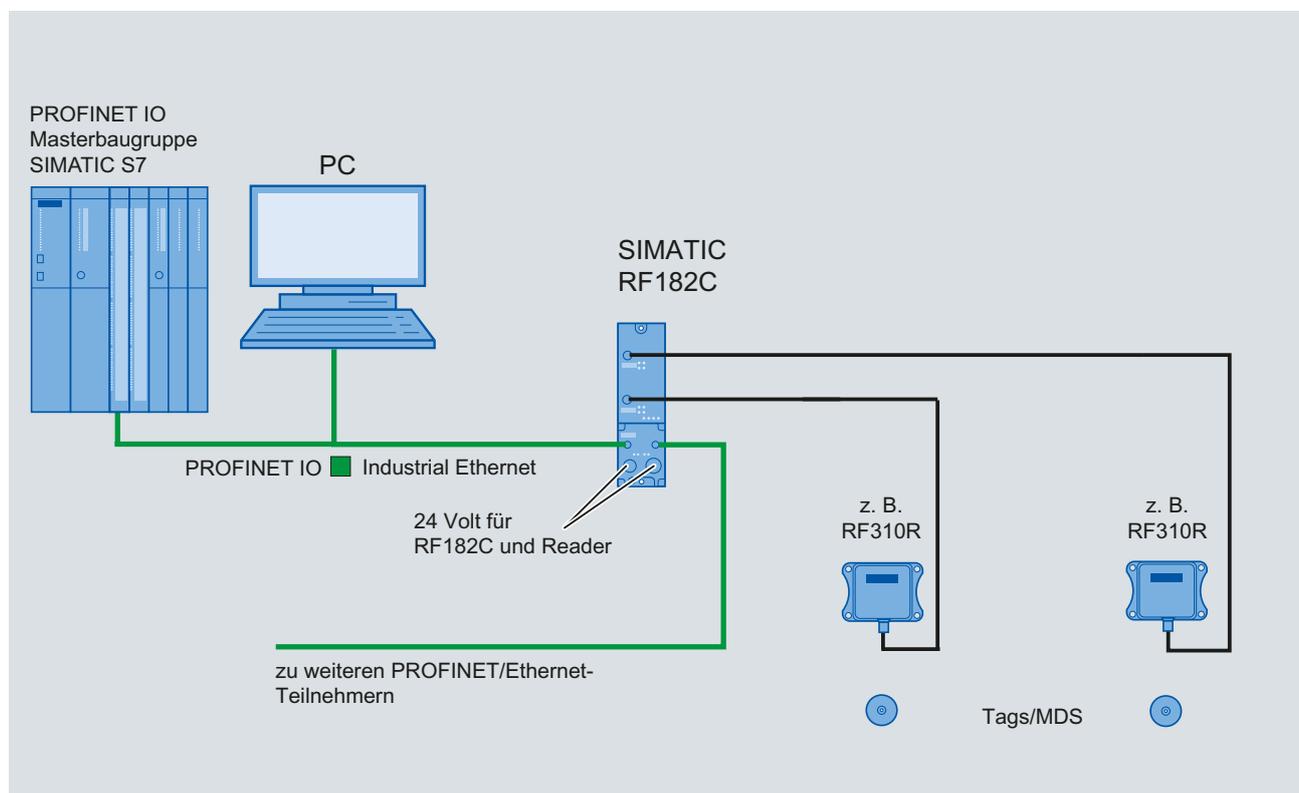


Bild 2-3 Konfigurator RF182C mit Anschlussblock M12, 7/8"



# Montieren

Das Kommunikationsmodul RF182C ist für einfaches Montieren ausgelegt.

## 3.1 Einbaulage, Einbaumaße

### Einbaulage

Für das RF182C bestehen keine Einschränkungen bei der Einbaulage.

### Einbau- und Abstandsmaße

Tabelle 3- 1 Einbaumaße Grundmodul mit Anschlussblock M12,7/8" (ohne Stecker)

<b>Bezeichnung</b>	<b>Maße</b>
Einbaubreite	60 mm
Einbauhöhe	210 mm
Einbautiefe	54 mm

Tabelle 3- 2 Einbaumaße Grundmodul mit Anschlussblock Push Pull (ohne Stecker)

<b>Bezeichnung</b>	<b>Maße</b>
Einbaubreite	60 mm
Einbauhöhe	216 mm
Einbautiefe	100 mm

## 3.2 Peripheriemodul montieren

### Eigenschaften

- Das Basisgerät wird auf einem festen Untergrund montiert

---

#### Hinweis

#### Funktionserde (PE)

Wird eine metallische, geerdete Montagefläche verwendet, so stellt die untere Befestigungsschraube der Baugruppe RF182C bereits eine sichere Erdverbindung her. Eine zusätzliche Erdungsleitung kann damit entfallen. Wenn Sie die Befestigungsschraube als Erdverbindung verwenden, dann muss das Gewinde der Befestigungsschraube bzw. die Auflage der Befestigungsmutter auf der Unterlage lackfrei sein. Somit ist eine niederohmige Verbindung sichergestellt.

---

### Voraussetzungen

Schrauben:

Schraubentyp	Erläuterung
Zylinderschraube M5 nach ISO1207/ISO 1580 (DIN 84 / DIN 85)	Die Schraubenlänge sollte mindestens 20 mm betragen.
Zylinderschraube mit Innensechskant M5 nach DIN 912	Zusätzlich benötigen Sie Unterlegscheiben nach DIN 125.

### Benötigtes Werkzeug

Kreuzschlitz-Schraubendreher mittel bzw. Steckschlüssel 8 mm.

### Vorgehensweise

Befestigen Sie das Basisgerät mit den Schrauben auf einem planen Untergrund. Das Basisgerät muss an beiden Befestigungsstellen (Vorderseite oben und unten) mit dem Untergrund verschraubt werden (Drehmoment 3 Nm).

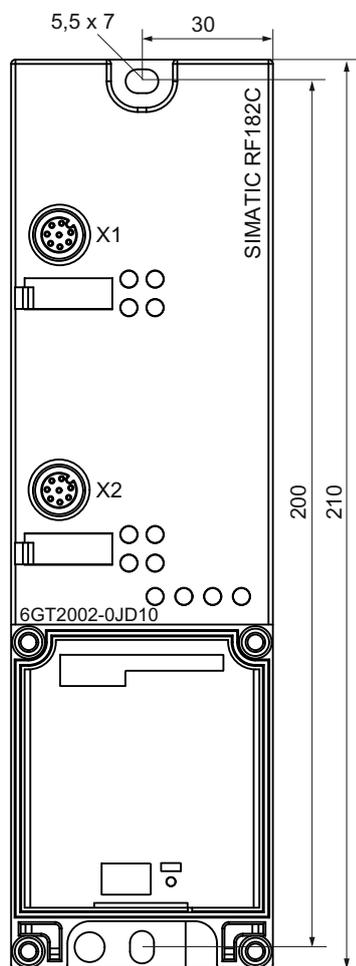


Bild 3-1 Peripheriemodul montieren

## 3.3 Anschlussblock montieren

### Eigenschaften

Der Anschlussblock verbindet das RF182C mit dem Ethernet und versorgt das Basisgerät mit Spannung.

### Voraussetzungen

Das Basisgerät ist montiert

### Benötigtes Werkzeug

Kreuzschlitz-Schraubendreher mittel.

### Anschlussblock montieren

1. Stecken Sie den Anschlussblock auf das Basisgerät

2. Verschrauben Sie den Anschlussblock mit dem Basisgerät (Drehmoment 1 bis 1,3 Nm). Ziehen Sie die Schrauben über kreuz gleichmäßig fest. 4 Schrauben befinden sich bereits im Anschlussblock (siehe Bild).

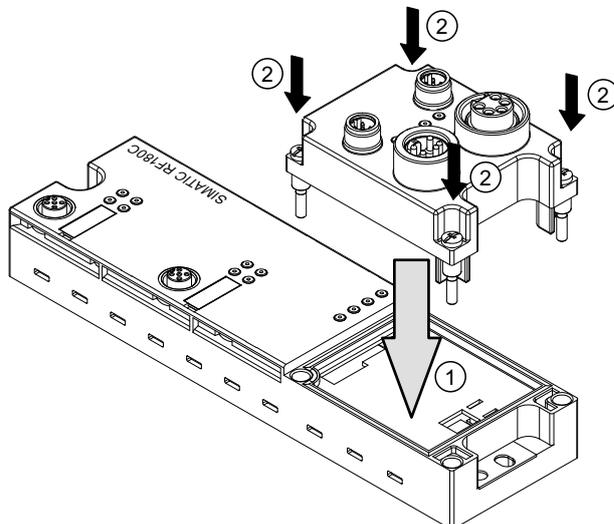


Bild 3-2 Anschlussblock M12, 7/8" auf das Basisgerät stecken und verschrauben

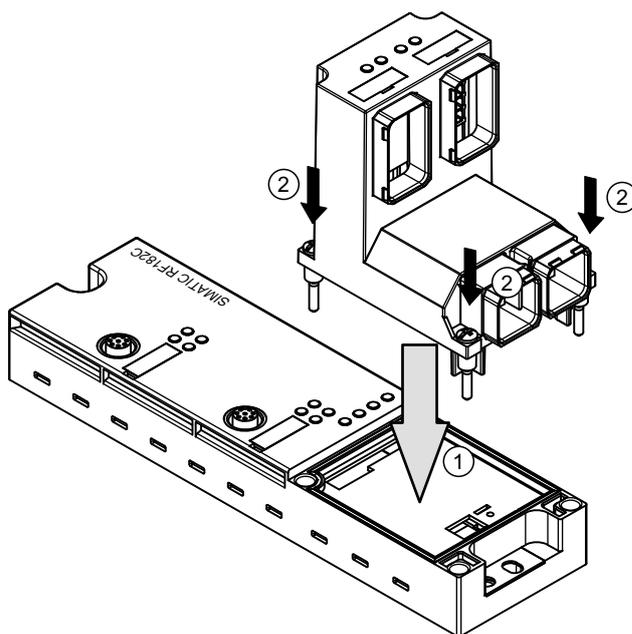


Bild 3-3 Anschlussblock Push Pull auf das Basisgerät stecken und verschrauben

#### Hinweis

Die Schutzart IP65, IP66 bzw. IP67 ist nur dann gewährleistet, wenn der Anschlussblock mit dem Basisgerät verschraubt ist.

## 3.4 Beschriftungsschilder austauschen

### Eigenschaften

Mit den Beschriftungsschildern können Sie jeden Kanal am Basisgerät und den Anschlussblock kennzeichnen. Die Beschriftungsstreifen sind im Auslieferungszustand bereits eingeklipst.

- 2 Beschriftungsschilder am Basismodul
- 1 Beschriftungsschild am Anschlussblock M12,7/8"
- 2 Beschriftungsschilder am Anschlussblock Push Pull

### Voraussetzung

Wenn Sie die Beschriftungsschilder austauschen möchten, dann können Sie diese nachbestellen. Die Bestellnummer finden Sie im Kapitel Bestelldaten (Seite 115) .

### Benötigtes Werkzeug

Schraubendreher 2,5 bis 4 mm.

### Beschriftungsschilder austauschen

1. Drücken Sie den Schraubendreher schräg in die kleine Öffnung am Beschriftungsschild und hebeln dieses heraus.

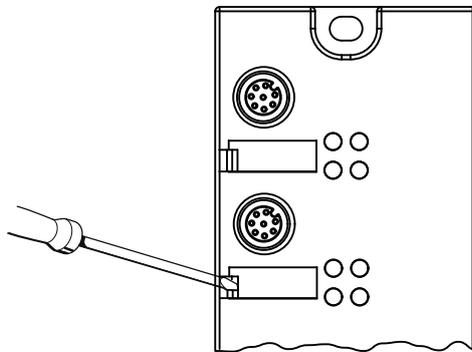


Bild 3-4 Beschriftungsschilder entfernen

2. Drücken Sie mit dem Finger das neue Beschriftungsschild in das Modul.

## 3.5 RF182C demontieren

### Vorgehensweise

Das RF182C ist verdrahtet und in Betrieb.

1. Schalten Sie die Versorgungsspannung für das RF182C ab.

2. Lösen Sie die Verdrahtung am Anschlussblock.
3. Lösen Sie die 4 Befestigungsschrauben am Anschlussblock und ziehen Sie den Anschlussblock vom Basisgerät.
4. Lösen Sie die Verdrahtung am Basisgerät.
5. Lösen Sie die Befestigungsschrauben des Basisgerätes.

---

**Hinweis**

Beachten Sie weiterhin die Informationen im Kapitel Ethernet und Versorgungsspannung weiterschleifen (Seite 31).

---



# Anschließen

## Bestimmungsgemäßer Gebrauch

Bei Anschluss von nicht spezifizierten Geräten an das RF182C kann das angeschlossene Gerät zerstört werden.

### ACHTUNG

Das Gerät darf ohne HUB / Switch **nicht** an das öffentliche Telefonnetz angeschlossen werden, da die Spannungs-Abstände auf 500 V ausgelegt sind.

## Ethernet-Aufbautechniken

Der Aufbau der Ethernet-Kommunikation kann als BUS oder STERN erfolgen. Beachten Sie dazu auch die Hinweise im Kapitel Ethernet und Versorgungsspannung weiterschleifen (Seite 31).

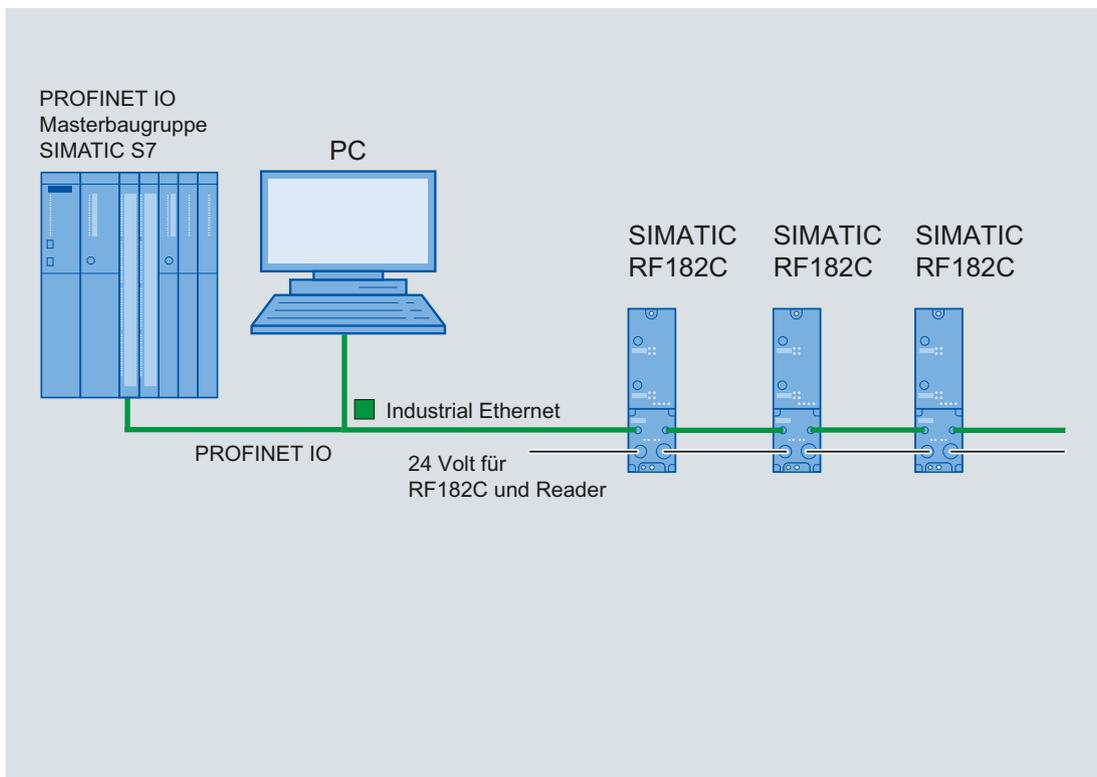


Bild 4-1 RF182C Aufbautechnik BUS

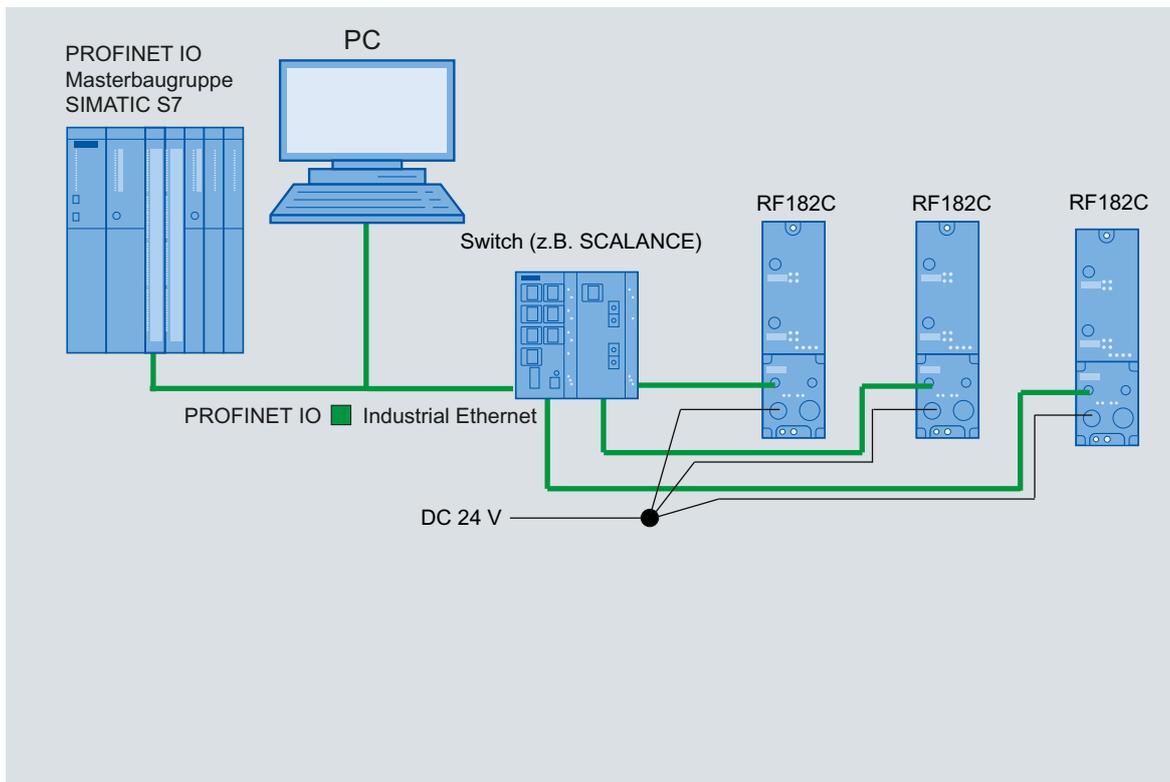


Bild 4-2 RF182C Aufbautechnik STERN

## Reader- / SLG-Anschlussstechnik

Ein Reader / SLG belegt immer eine M12-Anschlussbuchse auf dem RF182C. Für einen optimal einfachen Anschluss des Readers / SLG sorgt deshalb ein fertig konfektioniertes Kabel. Das Anschlusskabel besitzt in der Standardversion eine Länge von 2 m.

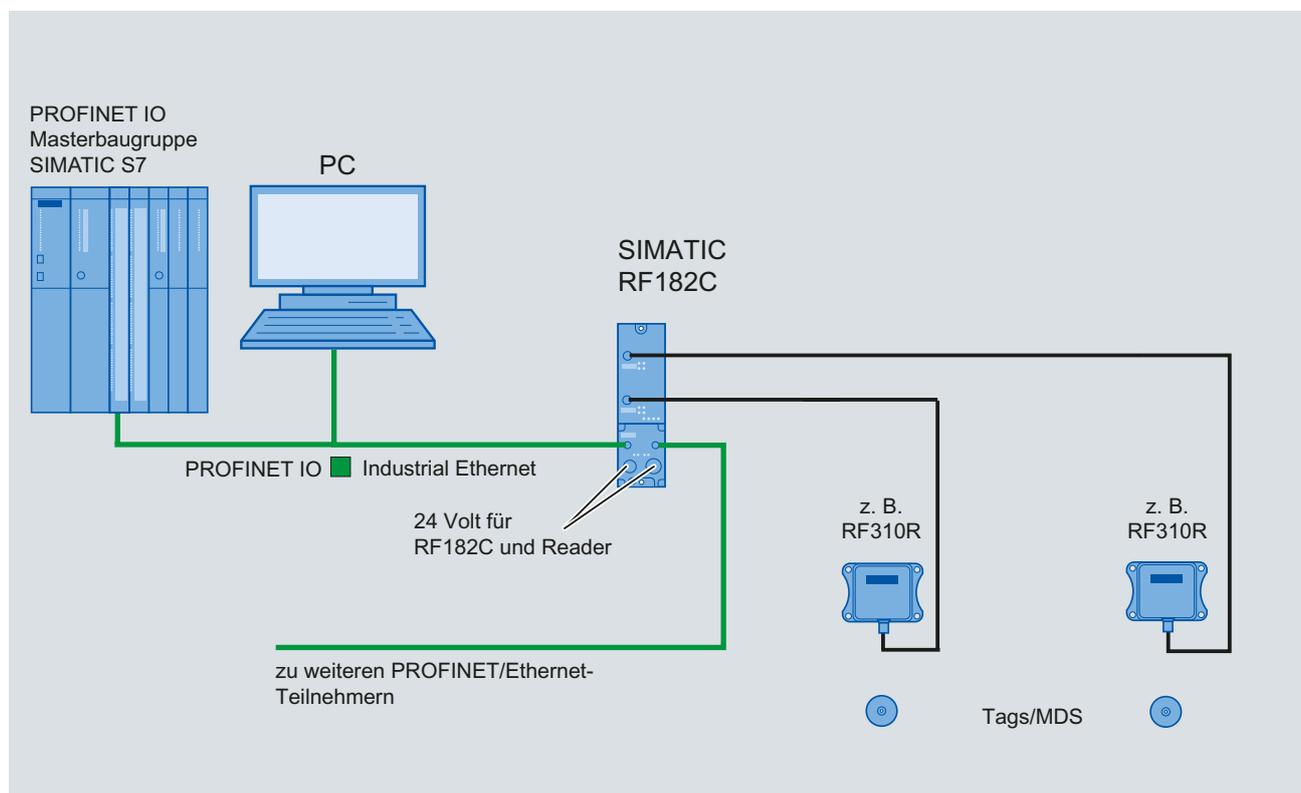


Bild 4-3 Übersicht Verkabelung

## 4.1 Anschlussblock M12, 7/8" verdrahten

### Eigenschaften

- Am Anschlussblock M12, 7/8" schließen Sie die Versorgungsspannungen und Ethernet an:
  - M12-Anschluss in Kodierung D: Ethernet
  - 7/8"-Anschluss: Versorgungsspannungen
- Über die zweiten M12- bzw. 7/8"-Rundbuchsen können Sie die Versorgungsspannungen und das Ethernet weiterschleifen.

### Voraussetzungen

- Verdrahten Sie den Anschlussblock M12, 7/8" bei ausgeschalteter Versorgungsspannung.

### Benötigtes Werkzeug

Abisolierwerkzeug, Schraubendreher zum Verdrahten des M12- bzw. 7/8"-Steckers, falls Sie keine vorkonfektionierte Kabel verwenden.

### Benötigtes Zubehör

- Vorkonfektionierte Kabel mit Stecker
- Falls Sie keine vorkonfektionierte Kabel verwenden:
  - M12: 4-adriges Ethernet-Kabel (Twisted Pair), geschirmt und M12-Stecker 4-polig, Kodierung D (siehe Tabelle *Anschlussbelegung M12-Stecker 4-polig, Kodierung D (Ethernet)*)
  - 7/8": 5-adriges Kabel und 7/8"-Stecker (siehe Tabelle *Anschlussbelegung 7/8"-Stecker (Versorgungsspannungen)*)
- Bestellnummern siehe Kapitel *Bestelldaten*.

**M12, 7/8"-Stecker verdrahten**

In den nachstehenden Tabellen finden Sie die Anschlussbelegung für den M12- und 7/8"-Stecker:

Tabelle 4- 1 Anschlussbelegung M12-Stecker 4-polig, Kodierung D (Ethernet)

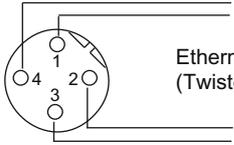
Pin	Belegung		Ansicht M12-Stecker 4-polig, Kodierung D (Verdrahtungsseite)
1	Datenleitung TxP	Datenleitung RxP	<p>Einspeisen und Weiterschleifen Ethernet X3, X4</p>  <p>Es ist unerheblich, über welchen Stecker das Einspeisen und das Weiterleiten geschieht.</p>
2	Datenleitung RxP	Datenleitung TxP	
3	Datenleitung TxN	Datenleitung RxN	
4	Datenleitung RxN	Datenleitung TxN	

Tabelle 4- 2 Anschlussbelegung 7/8"-Stecker 5-polig (Versorgungsspannungen)

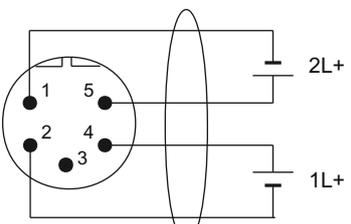
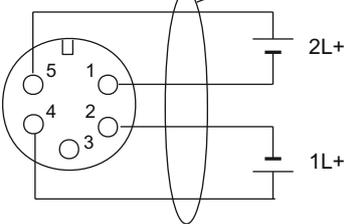
Pin	Belegung	Ansicht 7/8"-Stecker (Verdrahtungsseite)
1	Masse Lastspannungsversorgung (2M)	<p>Einspeisen X01</p>  <p>Weiterschleifen X02</p> 
2	Masse Elektronik- / Geberversorgung (1M)	
3	Funktionserde (PE)	
4	Elektronik- / Geberversorgung (1L+) (Spannungsversorgung für RF182C und Reader/SLG)	
5	Lastspannungsversorgung (2L+) (bei RF182C ungenutzt)	

Tabelle 4- 3 Anschlussbelegung 7/8"-Stecker 5-polig (Versorgungsspannungen)

Pin	Assignment	Ansicht 7/8" Stecker (Verdrahtungsseite)
1	Elektronik- / Geberversorgung (1L+) (Spannungsversorgung für RFID 181EIP und Reader/SLG)	
2	ungenutzt	
3	ungenutzt	
4	Masse Elektronik- / Geberversorgung (1M)	

**Hinweis**

Für den Anschluss der Versorgungsspannung empfehlen wir die im Kapitel Bestelldaten (Seite 115) angegebenen Kabel (Kabel 5 x 1,5 mm<sup>2</sup> konfektioniert mit 7/8"-Steckern).

Falls Sie das Kabel selber konfektionieren möchten, dann sollte der Leiterquerschnitt 1,5 mm<sup>2</sup> betragen.

### M12, 7/8"-Stecker anschließen

1. Drücken Sie den Stecker (M12 bzw. 7/8") in die jeweilige Rundbuchse am Anschlussblock. Achten Sie auf die korrekte Arretierung zwischen Stecker und Buchse (Nut und Feder).
2. Befestigen Sie den Stecker mit dem gerändelten Befestigungsring.

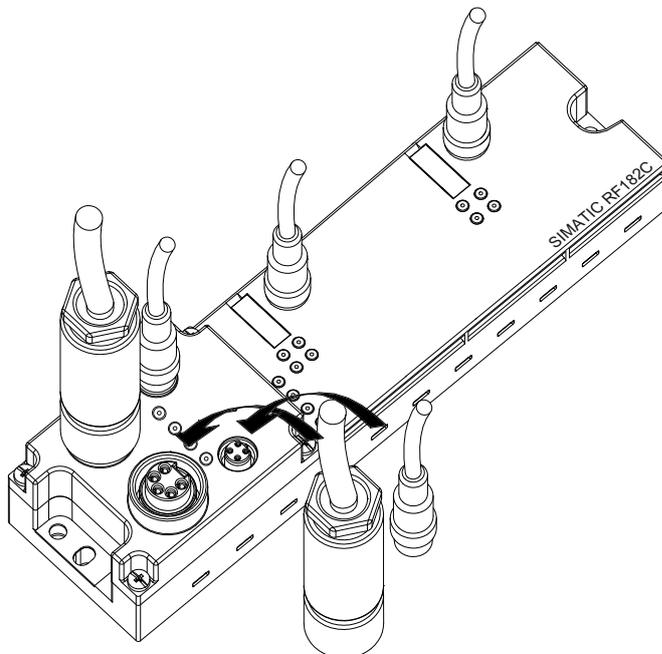


Bild 4-4 M12, 7/8"-Stecker anschließen

### Verschließen nicht benutzter Buchsen

Verschließen Sie alle nicht benutzten Buchsen mit M12 bzw. 7/8" Verschlusskappen, um die Schutzart IP65, IP66 bzw. IP67 zu erreichen. Bestellnummern siehe Kapitel Bestelldaten (Seite 115).

## 4.2 Anschlussblock Push Pull verdrahten

### Eigenschaften

- Am Anschlussblock Push Pull schließen Sie die Versorgungsspannungen und Ethernet an:
  - Push-Pull-Anschluss (RJ45) in Kodierung D: Ethernet
  - Push-Pull-Anschluss: Versorgungsspannungen
- Über die zweiten Push-Pull-Anschlüsse können Sie die Versorgungsspannungen und das Ethernet weiterschleifen.

### Voraussetzungen

- Verdrahten Sie den Anschlussblock Push Pull bei ausgeschalteter Versorgungsspannung.

### Benötigtes Werkzeug

- Schraubendreher
- Abisolierwerkzeug zum Verdrahten der Push-Pull-Anschlussstecker, falls Sie die Kabel selbst konfektionieren.

### Benötigtes Zubehör

- Vorkonfektionierte Kabel mit Push-Pull-Anschlussstecker für 1L+ / 2L+ und RJ45. Die Kabel sind in verschiedenen Längen von entsprechenden Herstellern erhältlich.
- Falls Sie die Kabel selbst konfektionieren:
  - 5-adriges Kabel und Push-Pull-Anschlussstecker für 1L+ / 2L+
  - 4-adriges Kabel, geschirmt (Buskabel) und Push-Pull-Anschlussstecker für RJ45

---

#### Hinweis

Beachten Sie die Dokumentation des Herstellers, wenn Sie die Kabel mit den Push-Pull-Anschlusssteckern konfektionieren.

---

## Push-Pull-Stecker verdrahten

In den nachstehenden Tabellen finden Sie die Anschlussbelegung für die Push-Pull-Stecker:

Tabelle 4- 4 Anschlussbelegung Push-Pull-Anschlussstecker (RJ45)

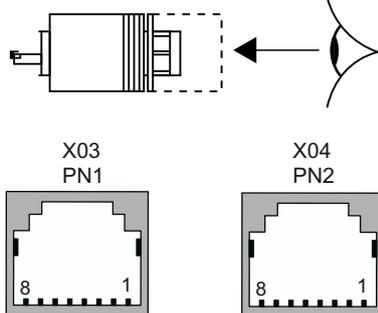
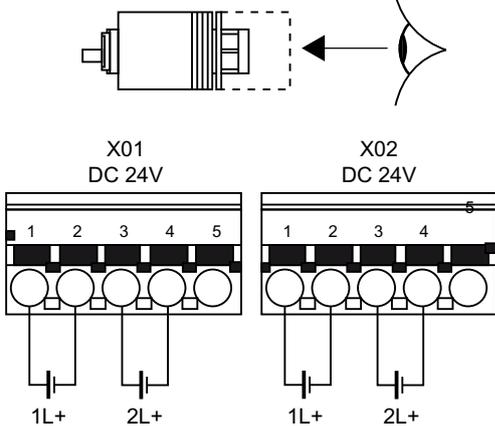
Ansicht Push-Pull-Anschlussstecker (RJ45)	Klemme	Belegung	
 <p>X03 PN1</p> <p>X04 PN2</p>	<b>X03 PN1 zum Einspeisen von Ethernet</b> <b>X04 PN2 zum Weiterschleifen vom Ethernet</b>		
	1	Transmit Data+ TD	Receive Data+ RD
	2	Transmit Data- TD_N	Receive Data- RD_N
	3	Receive Data+ RD	Transmit Data+ TD
	4	Ground GND (RJ45)	
	5	Ground GND (RJ45)	
	6	Receive Data- RD_N	Transmit Data- TD_N
	7	Ground GND (RJ45)	
	8	Ground GND (RJ45)	

Tabelle 4- 5 Anschlussbelegung Push-Pull-Anschlussstecker (Versorgungsspannungen 1L+ und 2L+)

Ansicht Push-Pull-Anschlussstecker (Versorgungsspannungen 1L+ und 2L+)	Klemme	Belegung	
 <p>X01 DC 24V</p> <p>X02 DC 24V</p> <p>1L+</p> <p>2L+</p>	<b>X01 DC 24V zum Einspeisen</b> <b>X02 DC 24V zum Weiterschleifen</b>		
	1	Elektronik- / Geberversorgung 1L+Masse	
	2	Masse Elektronik- / Geberversorgung 1M	
	3	Lastspannungsversorgung 2L+	
	4	Masse Lastspannungsversorgung 2M	
	5	Funktionserde (PE)	

**Hinweis**

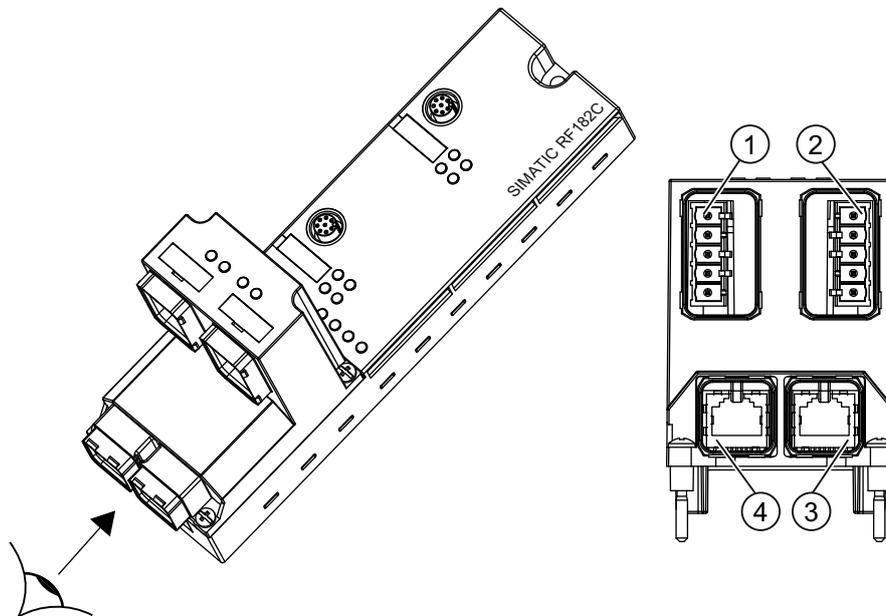
Für den Anschluss der Versorgungsspannung empfehlen wir die im Kapitel *Bestelldaten* (Seite 115) angegebenen Kabel (Kabel 5 x 1,5 mm<sup>2</sup> konfektioniert mit Push-Pull-Steckern).

Falls Sie das Kabel selber konfektionieren möchten, dann sollte der Leiterquerschnitt 1,5 mm<sup>2</sup> betragen.

**Bei einer Stromstärke > 8 A ist ein Leiterquerschnitt von 2,5 mm<sup>2</sup> zwingend vorgeschrieben.**

**Push-Pull-Anschlussstecker anschließen**

Drücken Sie die Push-Pull-Anschlussstecker für 1L+ / 2L+ und RJ45 in die zugehörigen Buchsen (siehe nachfolgendes Bild). Achten Sie dabei auf die korrekte Arretierung zwischen Stecker und Buchse. Die Stecker müssen einrasten.



- ① X01 DC 24 V Push-Pull-Buchse (mit Buchseneinsatz) zum Einspeisen von Elektronik- / Geberversorgung 1L+ und Lastspannungsversorgung 2L+
- ② X02 DC 24 V Push-Pull-Buchse (mit Buchseneinsatz) zum Weiterschleifen von Elektronik- / Geberversorgung 1L+ und Lastspannungsversorgung 2L+
- ④ X03 PN1 Push-Pull-Buchse für RJ45 zum Einspeisen vom Ethernet
- ③ X04 PN2 Push-Pull-Buchse für RJ45 zum Weiterschleifen vom Ethernet

Bild 4-5 Push-Pull-Anschlussstecker anschließen

**Verschließen nicht benutzter Buchsen**

Verschließen Sie alle nicht benutzten Push-Pull-Buchsen mit Abdeckkappen, um die Schutzart IP65, IP66 bzw. IP67 zu erreichen. Bestellnummern siehe Kapitel *Bestelldaten*.

## 4.3 Ethernet und Versorgungsspannung weiterschleifen

### Eigenschaften

Auf dem Anschlussblock befindet sich ein Stecker für die Einspeisung und eine Buchse zum Weiterschleifen der Versorgungsspannung. Stecker und Buchse für die Versorgungsspannung sind intern miteinander verbunden.

Für die Einspeisung und Weiterleitung des Ethernet stehen zwei Buchsen zur Verfügung. Beide Buchsen sind im Anschlussblock nicht miteinander verbunden. Die logische Verbindung beider Buchsen schafft der Switch im Grundgerät.

### Hinweis

Wenn Sie während des Betriebs den Anschlussblock demontieren, dann wird nur die Versorgungsspannung weitergeschleift. Die Datenkommunikation zu den nachfolgenden Geräten wird ab diesem Teilnehmer unterbrochen.

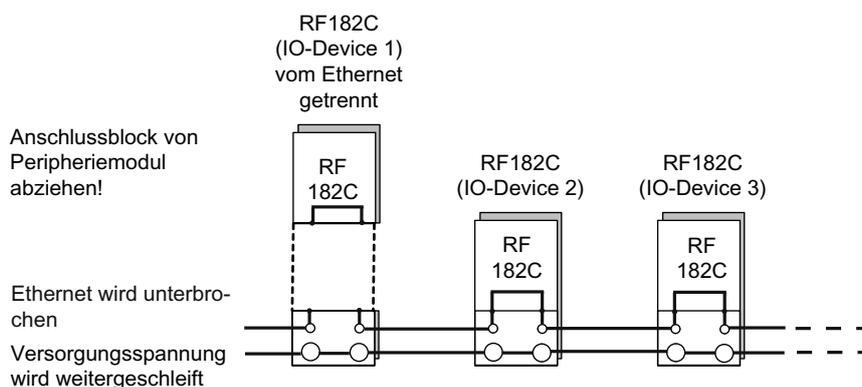


Bild 4-6 Ethernet und Versorgungsspannung weiterschleifen

### VORSICHT

Bei demontiertem Anschlussblock ist die Schutzart IP65, IP66 bzw. IP67 nicht mehr gewährleistet.

### Hinweise zum Verdrahten

- Wenn Sie Ihren Aufbau verdrahten, dann müssen Sie den Einfluss der Kabellänge auf die Versorgungsspannung am RF182C berücksichtigen.

**Beispiel:**

Bei einem 10 m langen Kabel mit  $\varnothing 1,5 \text{ mm}^2$  beträgt der Spannungsabfall 2,5 V bei 10 A Belastung, das entspricht 0,25 V bei 1 A Belastung.

- Der maximale Einspeisestrom des Anschlussblocks M12, 7/8" beträgt 6 A bei 1L+ und 8 A bei 2L+. Diese Werte dürfen nicht überschritten werden.
- Der maximale Einspeisestrom des Anschlussblocks Push Pull beträgt bis 40°C 12 A bei 1L+ und 2L+ und bis 60°C 8 A bei 1L+ und 2L+. Diese Werte dürfen nicht überschritten werden.
- Beachten Sie die Strombelastbarkeit der angeschlossenen Leitungen, die vom Leitermaterial, vom Leiterquerschnitt und von der Umgebungstemperatur abhängt.

<b>VORSICHT</b>
Wenn Sie den maximalen Einspeisestrom und den dazu erforderlichen Kabelquerschnitt nicht einhalten, dann kann dies zu Überhitzung der Kabelisolierung, der Kontakte und zu Schäden am Gerät führen.

<b>VORSICHT</b>
<b>Sachbeschädigung des Geräts</b>
Bei einer Stromstärke $> 8 \text{ A}$ ist ein Leiterquerschnitt von $2,5 \text{ mm}^2$ zwingend vorgeschrieben!

## 4.4 RF182C an Controller mit Ethernet-Anschluss verdrahten

Die Realisierung einer Verbindung vom Ethernet- zu M12-Anschluss ist mit einfachen Mitteln möglich.

### Selbstkonfektionierung eines Ethernet-M12-Kabels

- Sie benötigen ein fertig konfektioniertes PROFINET-/Ethernet-Kabel mit beidseitigen M12-Steckern in doppelter Länge. Weiter benötigen Sie zwei Ethernet-Stecker zur Selbstkonfektionierung. Schneiden Sie das M12-Kabel in der Mitte durch und schließen Sie je einen Ethernet-Stecker an die offenen Kabelenden an. Im Ergebnis erhalten Sie zwei Ethernet-M12-Kabel.
- Sie benötigen die Einzelteile Ethernet Steckverbinder, M12 Steckverbinder und PROFINET-/Ethernet-Standardkabel (unkonfektioniert). Sie finden die Teile in den Bestelldaten. Mit diesen Teilen können Sie Ihr eigenes Kabel in beliebiger Länge dimensionieren.

### Verwendung einer Schrankdurchführung Ethernet-M12

Diese Anschlussvariante ist immer dann zu bevorzugen, wenn die Controller-Elektronik in einen Schaltschrank eingebaut ist. Das folgende Bild zeigt das Anschlussschema.

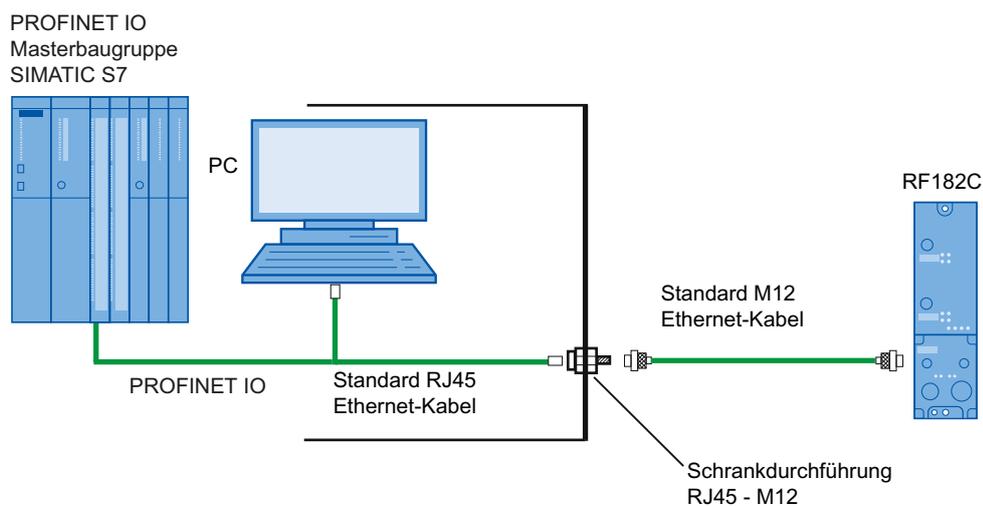


Bild 4-7 Schrankdurchführung

## 4.5 RF182C mit der Funktionserde (PE) verbinden

### Eigenschaften

- Sie müssen das RF182C mit der Funktionserde (PE) verbinden. Für diesen Zweck ist auf dem Kommunikationsmodul eine Erdungsschraube für eine Erdungsleitung vorgesehen.
- Wird eine metallische, geerdete Montagfläche verwendet, so stellt die untere Befestigungsschraube der Baugruppe RF182C bereits eine sichere Erdverbindung her. Eine zusätzliche Erdungsleitung kann damit entfallen.
- Die Verbindung zur Funktionserde (PE) wird auch zur Ableitung der Störströme und für die EMV-Festigkeit benötigt.

### Voraussetzungen

- Sorgen Sie immer für eine niederohmige Verbindung zur Funktionserde (PE).
- Wenn Sie die Befestigungsschraube als Erdverbindung verwenden, dann muss das Gewinde der Befestigungsschraube bzw. die Auflage der Befestigungsmutter auf der Unterlage lackfrei sein. Somit ist eine niederohmige Verbindung sichergestellt.

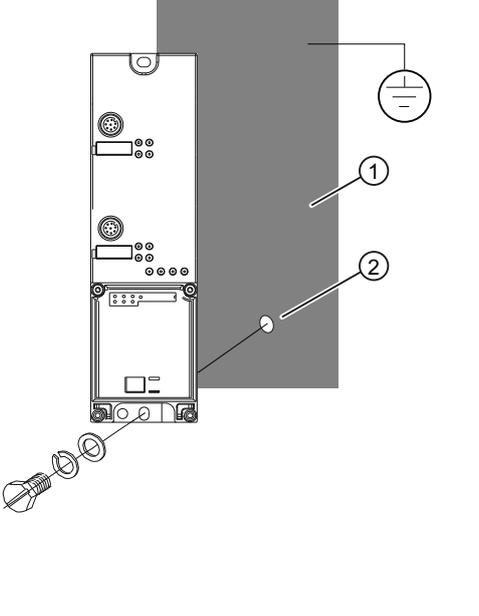
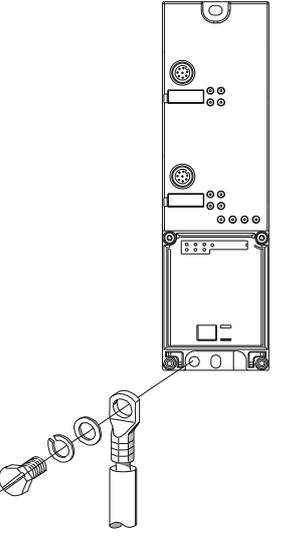
### Benötigtes Werkzeug, nur wenn Erdung über Erdungsleitung vorgesehen ist.

- Schraubendreher
- Abisolierwerkzeug
- Kabelschuhzange

### Benötigtes Zubehör, nur wenn Erdung über Erdungsleitung vorgesehen ist.

- Erdungsschraube M5 x 10 und Beilagscheiben
- Erdungsleitung (Kupfergeflechtleitung) mit Mindestquerschnitt 4 mm<sup>2</sup>
- Kabelschuh

## RF182C mit Funktionserde (PE) verbinden

Standard Erdung über Befestigungsschraube	Optionale Erdung über Erdungsleitung
<p>1. Montieren Sie die Baugruppe auf der geerdeten, metallischen Unterlage, wie im Kapitel <i>Peripheriemodul montieren</i> beschrieben.</p> <p>① Geerdete, metallische Unterlage ② Lackfreies Gewinde bzw. Mutterauflage</p>	<p>1. Isolieren Sie die Erdungsleitung ab und befestigen Sie den Kabelschuh.</p> <p>2. Verschrauben Sie den Kabelschuh am Kommunikationsmodul (Erdungsschraube M5). Das Drehmoment beträgt 3 Nm.</p>
 <p>Das Diagramm zeigt ein Kommunikationsmodul, das auf einer grauen, metallischen Unterlage montiert ist. Ein geerdetes Symbol (eine Kreislinie mit einem horizontalen Balken) ist durch eine Linie mit der Unterlage verbunden. Zwei Pfeile weisen auf die Unterlage hin: einer ist mit '1' beschriftet und zeigt auf die Unterlage, der andere mit '2' beschriftet zeigt auf ein Gewinde an der Unterlage. Ein M5-Schraubenset (Schraube, Unterlegscheibe, Mutter) ist unten links dargestellt, um die Befestigung zu verdeutlichen.</p>	 <p>Das Diagramm zeigt ein Kommunikationsmodul, das an einer Erdungsleitung angeschlossen ist. Die Erdungsleitung ist an einem Ende isoliert und hat einen Kabelschuh. Ein M5-Schraubenset (Schraube, Unterlegscheibe, Mutter) ist unten links dargestellt, um die Befestigung des Kabelschuhs am Modul zu verdeutlichen.</p>



# Parametrieren

## 5.1 Adresszuordnung für Ethernet

Der Reader wird über das Kommunikationsmodul RF182C mit dem Ethernet verbunden. Die Kommunikation zwischen der Applikation im PC (Client) und dem Reader (über das RF182C als Server) funktioniert nur mit einer eindeutigen Adresszuordnung:

### MAC-ADD

Die physikalische Adresse, MAC-ADD (Media Access Address), wird für jedes RF182C vom Hersteller festgelegt.

**Beispiel MAC-ADD:** 67-89-AB-CD-EF-01

Sie finden die MAC-ADD auf der Seitenfläche des RF182C aufgedruckt.

### IP-Adresse

Darüber hinaus benötigt jedes RF182C eine logische Adresse, IP-Adresse (Internet Protokoll), über die es im Netzwerk adressiert wird.

Die IP-Adresse darf im Netzwerk nur einmal vorhanden sein. Sie muss im RF182C parametrieren werden. In der Anwenderapplikation im PC wird die IP-Adresse beim Verbindungsaufbau angegeben.

Die IP-Adresse besteht immer aus 32 Bit und wird im Dezimalformat (Wertebereich von 0 bis 255) dargestellt. Sie ist somit ein Zeichenstring aus vier Zahlenwerten im ASCII-Format, jeweils durch einen Punkt getrennt.

**Beispiel IP-Adresse:** "157.163.170.12";

### Subnetzmaske

Zur Bestimmung des Netzwerks wird die Subnetzmaske (Subnetmask) benötigt. Die Subnetzmaske ist der IP-Adresse ähnlich. Sie besteht aus vier durch einen Punkt getrennten Zahlen (Default-Wert: 255.255.255.0).

**Beispiel Subnetzmaske** "255.255.0.0"

### Socket

Ein Socket ist ein Kommunikationsendpunkt, der durch eine IP-Adresse und einen Port definiert wird.

## Port

Ein Port ist ein Zugangspunkt, über die eine bestimmte Funktion auf dem Gerät angesprochen werden kann. Im Falle der RF182C wird z. B. die Adressierung der Reader über Port-Nummern vorgenommen.

---

### Hinweis

#### Portzuordnung über Webserver

Die Zuordnung der Ports kann über den Webserver verändert werden. Wenn kein Verbindungsaufbau mit der RF182C stattfinden konnte, überprüfen Sie die Port-Einstellung über den Webserver. Die Default-Port-Einstellung lautet: 10001/10002.

Nehmen Sie anschließend den Verbindungsaufbau über die eingestellten Portnummern vor.

---

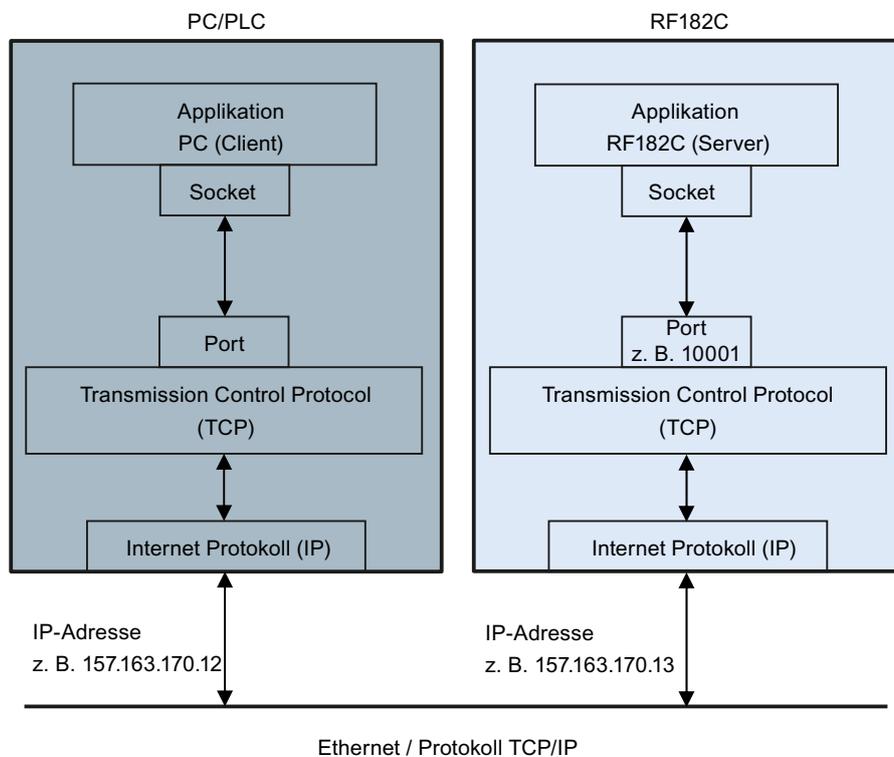


Bild 5-1 TCP/IP-Verbindung

Nähere Informationen zur Socketprogrammierung können Sie dem Kapitel Voraussetzungen zur Socketprogrammierung (Seite 87) entnehmen.

## 5.2 Datenkommunikation zwischen Client und RF182C

### Prinzipieller Ablauf

- Die Baugruppe ist hochgelaufen und wurde noch nicht parametriert.

---

#### Hinweis

#### Verbindungsprobleme?

Wenn kein Verbindungsaufbau mit der RF182C stattfinden konnte, überprüfen Sie über den Webserver (Seite 44) die Kommunikationseinstellungen (IP-Adresse, Portnummer) des Kommunikationsmoduls.

---

- Optional kann an das RF182C ein Konfigurationstelegramm (comDevSetConfig) geschickt werden, um den defaultmäßig im RF182C hinterlegten Kommunikationsmodus zu verändern. Der Client, der zuerst die RF182C konfiguriert, gibt die Einstellungen der RF182C vor.

Eine weitere Konfiguration ist im laufenden Betrieb nicht möglich. Die Baugruppe muss spannungslos geschaltet werden bzw. über Webserver über "Reset" neu gestartet werden, um neu konfiguriert werden zu können.

- Wenn kein Konfigurationstelegramm gesendet wird und nicht über den Webserver die Default-Port-Einstellung verändert wurde, wird das RF182C in der Default-Einstellung betrieben.

Die Default-Werte entnehmen Sie dem folgenden Bild:

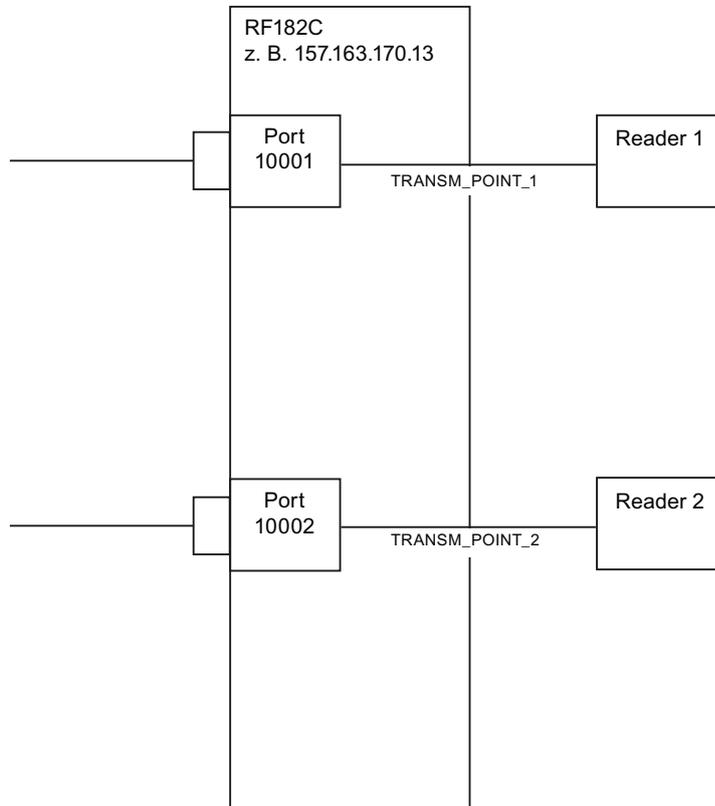


Bild 5-2 Default-Einstellungen des RF182C

Parameter	Wert
Baudrate	115,2 kBaud
Port 10001	Zuweisung Reader 1 (siehe Kapitel Webserver (Seite 44) )
Port 10002	Zuweisung Reader 2 (siehe Kapitel Webserver (Seite 44) )
Asynchrone Telegramme (Alarm-/Anwesenheitsmeldung) werden dem geöffneten Port zugeordnet.	
LED-Supression	NONE
Mode	U/D/RF300/RF600

**ACHTUNG****SLG D11S und D12S mit Standard-Baudrate nicht lauffähig**

Beachten Sie, dass mit der Standard-Baudraten-Einstellung die MODY D-Reader SLG D11S und D12S nicht lauffähig sind.

Wenn Sie diese Reader mit dem Kommunikationsmodul betreiben möchten, parametrieren Sie zuerst das Kommunikationsmodul und stellen dabei die Baudrate von 19,2 kBaud ein.

- Das vom Anwender aufgesetzte RESET-Telegramm wird an den entsprechenden Reader geschickt.
- Der weitere Ablauf erfolgt mit einem Kommandotelegramm je nach Anwendungsfall.
- Nach längeren Telegrammpausen (ca. 3 s) kann die Applikation (Client) selbstständig ein heartbeat-Telegramm (Leitungsüberwachung) senden, um die Verbindung zu testen. Das Kommunikationsmodul RF182C quittiert daraufhin das Telegramm. Sie müssen sich dabei selbstständig um die Verbindungsüberwachung kümmern und dabei das Intervall festlegen, nachdem der Client selbstständig ein heartbeat-Telegramm schicken soll. Falls im Fehlerfall keine Antwort auf das heartbeat-Telegramm von der RF182C gesendet wird, muss anschließend der Client weitere Aktionen veranlassen (Verbindung schließen/Verbindung aufbauen/Reader parametrieren).
- Die Verbindungen (inklusive TCP/IP) können jeweils von beiden Seiten aufgrund von folgenden Ursachen abgebaut werden:
  - Inaktivität (Timeout, keep-alive auf TCP-Ebene)
  - Verbindungsfehler
  - Abbauwunsch
- Nach Abbau der Verbindung vom Server muss der Client die Verbindung wieder aufbauen, einen RESET-Befehl schicken, usw.

### Datenkommunikation

Die folgende Grafik zeigt das Prinzip der Datenkommunikation zwischen Applikation (Client) und RF182C:

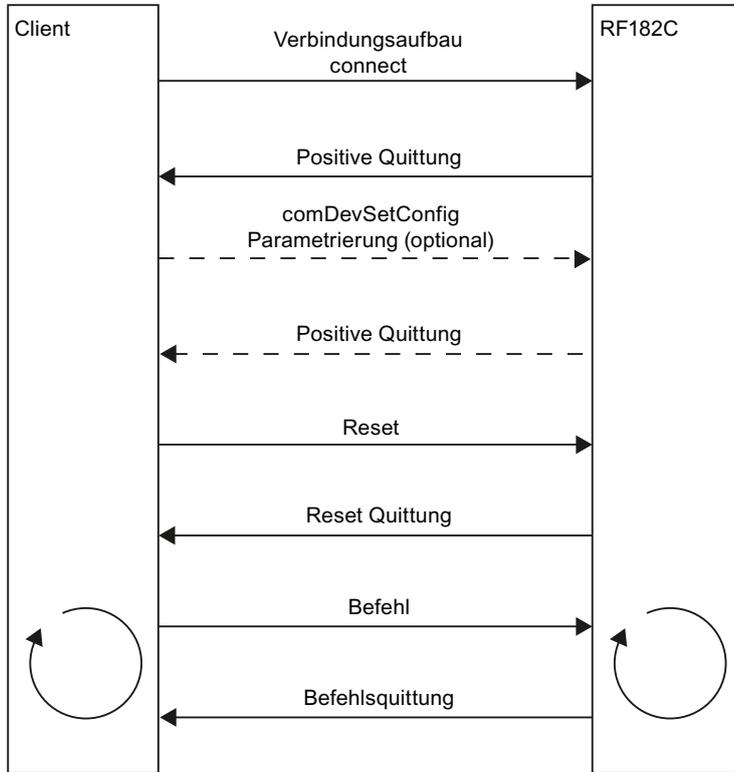


Bild 5-3 Prinzip Datenkommunikation

## 5.3 RF182C im Lieferzustand

Jedes RF182C erhält bereits werksseitig eine eindeutige Geräteidentifikation (MAC-Adresse). Bei der Projektierung und Programmierung wird das Kommunikationsmodul über die IP-Adresse angesprochen.

Deshalb müssen Sie zuerst die Informationen zur IP-Adresse (IP-Adresse und Subnetzmaske) dem RF182C zuweisen, um es im Ethernet-Netzwerk benutzen zu können. Das RF182C kann eine IP-Adresse über PST-Tool oder Webserver erhalten.

### Lieferzustand

- IP-Adresse im Lieferzustand (Default): 192.168.0.100
- Port-Einstellung im Lieferzustand (Default:
  - 10001 Reader 1
  - 10002 Reader 2

Um dem Kommunikationsmodul eine IP-Adresse zuzuweisen, benutzen Sie die Software "Primary Setup Tool" (<http://support.automation.siemens.com/WW/view/de/19440762>) (ab V4-0).

## 5.4 IP-Adresse vergeben

### 5.4.1 Überblick

Sie haben zwei gleichberechtigte Möglichkeiten, dem Kommunikationsmodul RF182C eine IP-Adresse zuzuweisen:

- über das "Primary Setup Tool V4-0"
- über den Webserver des Kommunikationsmoduls

Im Folgenden werden beide alternative Vorgehensweisen kurz beschrieben.

### 5.4.2 Webserver

#### Vorgehensweise

1. Geben Sie die IP-Adresse des Kommunikationsmoduls in das Adressfeld Ihres Browsers ein.

Der Webserver des Kommunikationsmoduls öffnet sich.

### Hinweis

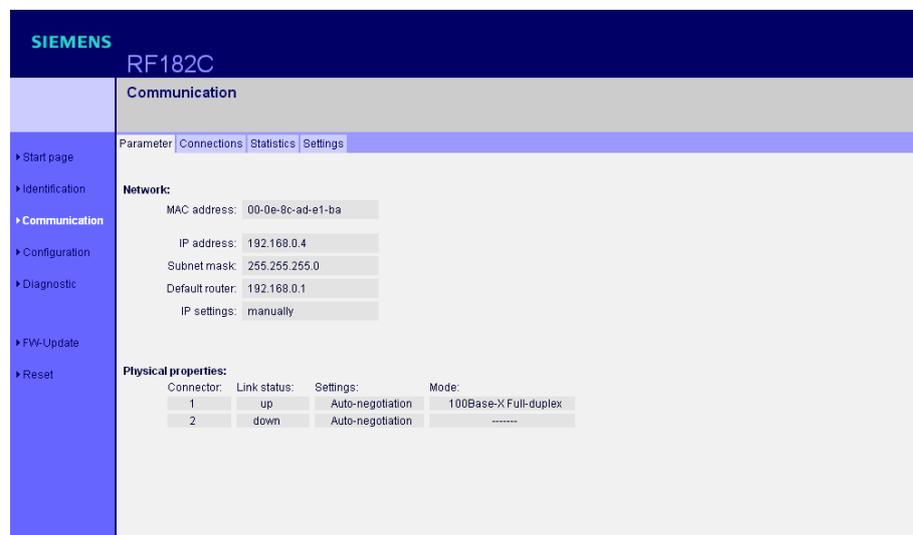
#### Kein Kontakt zum Webserver RF182C

Wenn sich der Webserver des Kommunikationsmoduls nicht öffnet, stellen Sie sicher, dass alle Kabel richtig verbunden sind, und prüfen Sie, ob das Kommunikationsmodul RF182C hochgelaufen ist.

- Überprüfen Sie die IP-Adresse des PCs und die Adresse der Subnetzmaske im Menü "Communication", Register "Parameter".

### Hinweis

Stellen Sie bei einer Verbindung sicher, dass dem PC, Laptop oder PLC eine IP-Adresse des gleichen Subnetzes zugewiesen ist, sofern kein Router verwendet wird. Die IP-Adresse von Kommunikationsmodul und PC müssen die gleiche Subnetzmaske haben.

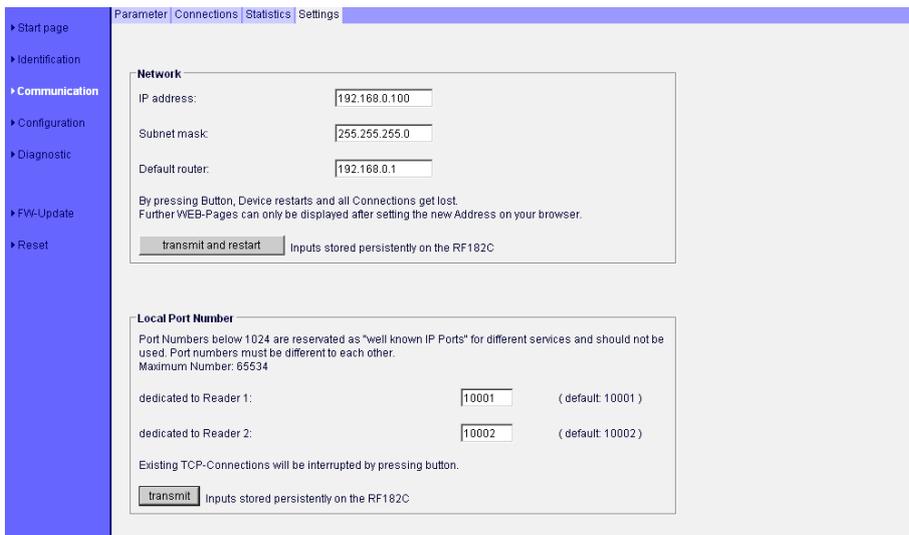


- 3. Im Register "Settings" finden Sie die Einstellungen, die derzeit auf dem Gerät RF182C gültig sind. Hier können Sie die IP-Adresse, Subnetmaske, den Default Router bzw. die Port-Nummern der einzelnen Reader verändert werden.

**ACHTUNG**

**Abbruch der Kommunikation**

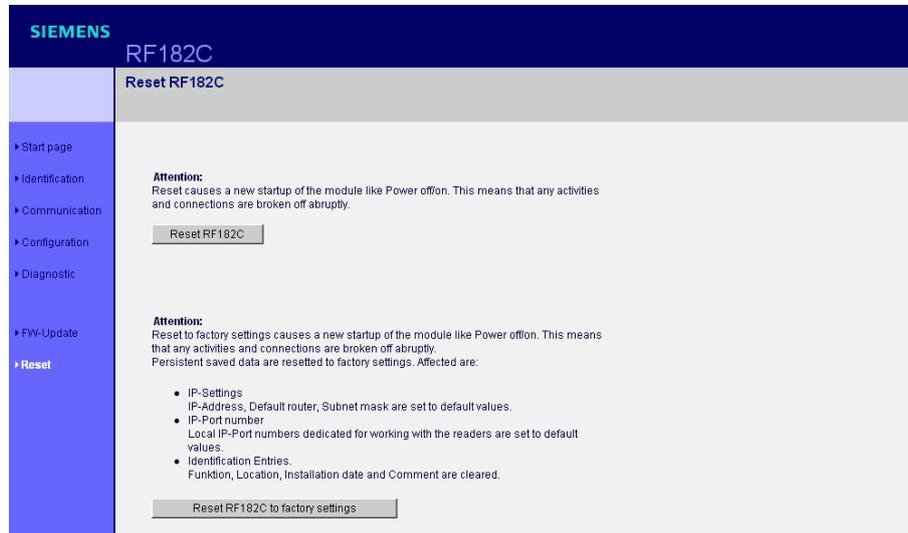
Wenn Sie im laufenden Betrieb die Einstellungen verändern, wird die Kommunikation abgebrochen. Die Anwendung muss sich anschließend wieder mit den neuen Einstellungen vom Kommunikationsmodul RF182C verbinden. Das gleiche gilt für den Webserver.



Beim erneuten Hochlauf des Kommunikationsmoduls sind die Default-Einstellungen nicht mehr aktiv, sondern die geänderten Einstellungen.

## Zurücksetzen auf Werkseinstellungen

Über das Menü "Reset" können Sie alle Einstellungen auf die Werkseinstellungen zurücksetzen.



1. Klicken Sie dazu auf die Schaltfläche "Reset RF182C to factory settings".

---

### Hinweis

#### Soft Reset

Wenn schwerwiegende Kommunikationsfehler auftreten sollten, können Sie auch einen sogenannten "Soft Reset" über die Schaltfläche "Reset 182C" durchführen. Das Kommunikationsmodul startet dann neu wie nach einer Spannungswiederkehr.

---

## 5.4.3 Primary Setup Tool

### Vorgehensweise

1. Rufen Sie das "Primary Setup Tool V4-0" über "Start > SIMATIC > Primary Setup Tool" auf.

2. Wählen Sie im Menü unter "Einstellungen" die Netzwerkkarte des PCs aus und klicken Sie auf die Schaltfläche "Suchen" .

Es erscheint ein Fenster, dass ein Gerät im Netzwerk gefunden wurde.

Wenn kein Kommunikationsmodul RF182C erscheint, stellen Sie sicher, dass alle Kabel richtig verbunden sind und prüfen Sie, ob das Kommunikationsmodul RF182C hochgelaufen ist.

Überprüfen Sie die IP-Adresse des PCs und die Subnetzmaske.

---

#### **Hinweis**

Stellen Sie bei einer Verbindung sicher, dass dem PC, Laptop oder PLC eine IP-Adresse des gleichen Subnetzes zugewiesen ist, sofern kein Router verwendet wird. Die IP-Adresse von Kommunikationsmodul und PC müssen die gleiche Subnetzmaske haben.

---

- Um die Einstellungen des Kommunikationsmoduls anzuzeigen, klicken Sie auf das "+"-Zeichen neben dem Ordner-Symbol:

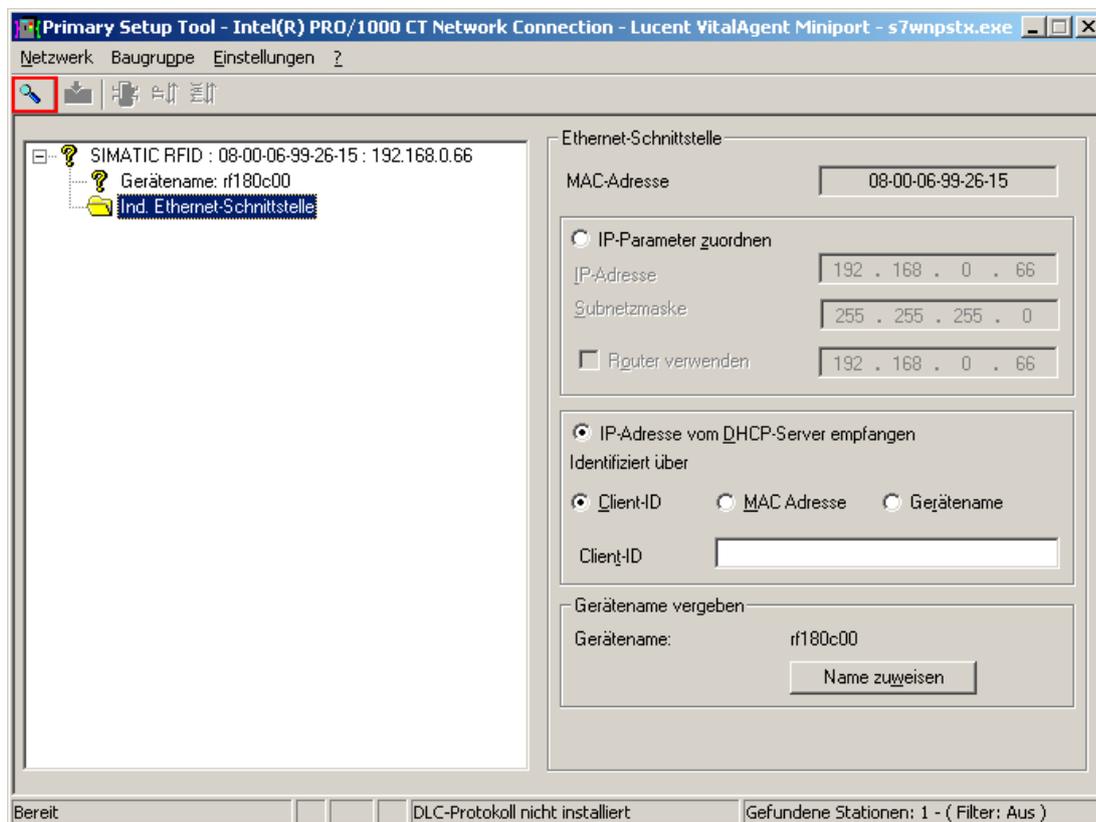


Bild 5-4 Einstellungen Kommunikationsmodul

- Wenn Sie die Daten unter "IP-Parameter zuordnen" verändern möchten, aktivieren Sie die nebenstehende Optionsschaltfläche.
  - Über die Schaltfläche "Name zuweisen" können Sie für das Gerät explizit einen Namen vergeben, z. B. RF182C.
- Wenn Sie die geänderten Daten auf das Kommunikationsmodul übertragen möchten, markieren Sie den übergeordneten Ordner und wählen Sie im Menü "Baugruppe > Laden". Bestätigen Sie das nachfolgende Fenster mit "Ja":

#### ACHTUNG

##### Wartezeit

Warten Sie, bis die IP-Adresse aktualisiert ist. Zum Anzeigen der Änderung müssen Sie die Suchfunktion über die Schaltfläche  aktivieren.

## Ergebnis

Sie können das Gerät nun mit einem Browser oder Anwenderprogramm ansprechen.

---

### Hinweis

#### **IP-Adresse auf Anschlussblock hinterlegt**

Die IP-Adresse wird auf dem Anschlussblock hinterlegt. Wenn Sie das Grundmodul austauschen, muss daher keine neue IP-Adresse vergeben werden.

---

## 5.5 Troubleshooting: IP-Adresse vergeben

Wenn Sie Probleme haben, dem Kommunikationsmodul SIMATIC RF182C eine IP-Adresse zuzuweisen, gehen Sie nach folgender Checkliste vor:

### Vorgehensweise

1. Schließen Sie das RF182C direkt oder über Hub/Switch an Ihren PC / Notebook/ PLC an. Schließen Sie kein anderes Modul/Gerät an das Netzwerk an. Schalten Sie das Kommunikationsmodul RF182C zu diesem Zeitpunkt noch **nicht** ein.
2. Trennen Sie alle anderen Netzwerkkabel von Ihrem PC / Notebook und stellen Sie sicher, dass das RF182C das einzige Netzwerkgerät an Ihrem PC / Notebook ist.
3. Schalten Sie das RF182C jetzt ein. Achten Sie auf die folgenden LEDs:
  - Am Gerät sollten nach dem Hochlauf alle 3 Sekunden die Error-LED 1 und 2 blinken.
  - Die LED "SF" sollte leuchten.
  - Die LED "ON" sollte leuchten.
  - Die LED "DC 24" V sollte leuchten.
  - Die LED "BF" sollte blinken.
  - Eine der zwei grünen Link-LEDs sollte eingeschaltet sein (grün). Eine oder zwei LEDs "RxTx" sollten abhängig von der Kommunikationslast im Netz flackern.
4. Starten Sie die Software "Primary Setup Tool V4-0" und konfigurieren Sie die Netzwerk-Einstellungen, falls nicht schon geschehen.
5. Aktualisieren Sie die Ansicht über die Schaltfläche "Suchen" .  
In der Software "Primary Setup Tool V4-0" sollte jetzt das Kommunikationsmodul zu sehen sein.
6. Klicken Sie in der Software auf das "+"-Zeichen neben dem Ordner-Symbol des Kommunikationsmoduls.
7. Klicken Sie rechts auf die Optionsschaltfläche "IP-Parameter zuordnen".
  - Geben Sie eine gültige IP-Adresse ein.
8. Um die Daten zu übernehmen, markieren Sie den übergeordneten Ordner des Kommunikationsmoduls und wählen Sie im Menü "Baugruppe > Laden". Bestätigen Sie das nachfolgende Fenster mit "Ja".
9. Bei der nächsten Anfrage des "Primary Setup Tool V4-0", sollte dem Kommunikationsmodul die IP-Adresse zugewiesen sein.



## 6.1 Befehlsübersicht

Tabelle 6- 1 Befehlsübersicht

Name Befehl	Beschreibung
comDevSetConfig	Optional: RF182C umprojektieren
reset	RF182C und Reader eines Kanals zurücksetzen und parametrieren
writeTagData	Schreiben auf Adresse des Tags
readTagData	Lesen von Adresse des Tags
initializeTag	Tag initialisieren
getReaderStatus	Status der angeschlossenen Reader
setAnt	Antenne Ein-Aus
heartbeat	Leitungsüberwachung
getTagStatus	Status des Tag

### Hinweis

#### Text-Datei zur Struktur der XML-Befehle

Die Struktur der XML-Befehle dieses Kapitels finden Sie auch in einer Text-Datei (RF182C\_XML\_Commands.txt) abgebildet. Sie finden diese Datei auf der RFID-CD "Software & Documentation". Folgen sie dem Link "CM > ASM > RF182C > Tools".

Damit können Sie mit Copy und Paste die Grundstruktur der Befehle in Ihr Applikationsprogramm übernehmen.

### Hinweis

#### Aufbau der XML-Befehle

Achten Sie darauf, dass zwischen einem öffnenden und einem schließenden XML-Tag immer ein Wert eingetragen sein muss. Sonst wird das Telegramm mit dem Fehler 3550 quittiert.

Beispiel:

```
<baudrate>115200</baudrate>
```

Wenn ein XML-Tag-Paar keinen Wert enthält, lassen Sie das komplette Paar weg.

## 6.2 Projektierungsparameter des RF182C

Die Baugruppe RF182C enthält bereits eine Default Projektierung durch das Werk. In den meisten Fällen kann deshalb dieser Befehl in der Anwendung entfallen.

### XML-Befehl

```
<command>
  <comDevSetConfig>
    <signature>RF182C</signature>
    <protocolVersion>Version</protocolVersion>
    <parameter>
      <transmissionPoint>
        <mode>Mode</mode>
        <baudrate>Baudrate</baudrate>
        <startupLedSupression>LEDSupression</startupLedSupression>
      </transmissionPoint>
    </parameter>
  </comDevSetConfig>
</command>
```

### XML-Antwort

Im Folgenden ist die XML-Antwort ohne Fehlereintrag (Fehlereinträge finden Sie im Kapitel Fehlermeldungen (Seite 77)) beschrieben:

```
<reply>
  <resultCode>Errorcode</resultCode>
  <comDevSetConfig/>
</reply>
```

## Parameter

Parameter	Datentyp	Werte
Version	ASCII	0 ... 9, A ... Z Hier müssen Sie folgende Version des Protokolls als String eintragen: <ul style="list-style-type: none"><li>• V1.0</li></ul>
RF182C	ASCII	Hier müssen Sie folgende Werte eingeben: <ul style="list-style-type: none"><li>• RF182C</li></ul>
Baudrate	ASCII	Hier müssen Sie eine der folgenden Baudraten (in Baud) eintragen: <ul style="list-style-type: none"><li>• 19200</li><li>• 57600</li><li>• 115200</li></ul>
LEDSupression	ASCII	Es blinken beide Kanäle, Kanal 1 oder Kanal 2 <ul style="list-style-type: none"><li>• NONE = Das Hochlaufblinken wird an keinem der beiden Reader Anschlüsse unterdrückt</li><li>• TRANSM_POINT_1 = Das Hochlaufblinken der err_LED am Reader 1 Anschluss wird unterdrückt</li><li>• TRANSM_POINT_2 = Das Hochlaufblinken der err_LED am Reader 2 Anschluss wird unterdrückt</li></ul>
Mode	ASCII	U/D/RF300_DIRECT_ADDRESSING

---

### Hinweis

Baudrate und Mode sind nicht einzeln einstellbar sondern gelten für beide Kanäle.

---

### 6.3 Input-Parameter des RF182C

Der RESET Befehl ist immer nach dem Einschalten/Hochlauf der Baugruppe notwendig. Er enthält die Einstellung für den am RF182C angeschlossenen Reader.

Im Folgenden ist der XML-Befehl für den RESET-Befehl beschrieben:

#### XML-Befehl

```
<command>
  <reset>
    <param>Param</param>
  </reset>
</command>
```

#### XML-Antwort

Im Folgenden ist die XML-Antwort ohne Fehlereintrag (Fehlereinträge finden Sie im Kapitel Fehlermeldungen (Seite 77)) beschrieben:

```
<reply>
  <resultCode>ErrorCode</resultCode>
  <reset/>
</reply>
```

#### Parameter

Parameter	Datentyp	Werte
Param	ASCII-Hex	00 ... FF (ASCII-String mit einer Länge von 16 ASCII-Zeichen; das entspricht 8 Hex-Zahlen) z. B. 00 25 00 00 00 01 00 00 (Anwesenheit eingeschaltet RF300, kein multitag) <sup>1)</sup>

<sup>1)</sup> Der ASCII-String darf kein Leerzeichen enthalten. Im Beispiel wurden die Leerzeichen zur besseren Übersichtlichkeit eingefügt.

Die byte-weise Aufschlüsselung des RESET-Befehls finde Sie im Kapitel Befehls- und Quittungstelegramme (Seite 117) .

## 6.4 Befehle des Kommunikationsmoduls

### 6.4.1 writeTagData

Mit dem Befehl können Teilbereiche oder der komplette Tag beschrieben werden, soweit die Adressbereiche physikalisch linear auf dem Tag vorhanden sind.

Im Folgenden ist der XML-Befehl für den Schreib-Befehl (**writeTagData**) beschrieben:

#### XML-Befehl

```
<command>
  <writeTagData>
    <startAddress>Address</startAddress>
    <data>Data</data>
  </writeTagData>
</command>
```

Informationen über die Speichergrößen der Tags finden Sie im Kapitel Adressierung der RFID-Tags (Seite 129) .

#### XML-Antwort

Im Folgenden ist die XML-Antwort ohne Fehlereintrag (Fehlereinträge finden Sie im Kapitel Fehlermeldungen (Seite 77)) beschrieben:

```
<reply>
  <resultCode>Errorcode</resultCode>
  <writeTagData/>
</reply>
```

---

#### Hinweis

Die Länge der zu schreibenden Daten ergibt sich aus den in der Variablen <data> übertragenen Zeichenanzahl. Beachten Sie, dass jeweils 2 ASCII Zeichen des übertragenen Datenstromes zu einem Hex-Zeichen auf dem Tag im Kommunikationsmodul konvertiert werden.

---

### Parameter

Parameter	Datentyp	Werte
Address	ASCII-Hex	0000 ... FFFF (4 x ASCII) <ul style="list-style-type: none"><li>• 0 bis maximale Länge der (User-Daten - 1)</li><li>• Ab dieser Anfangsadresse werden die User-Daten auf den Tag geschrieben.</li><li>• Beachten Sie, dass dieser Parameter tagabhängig ist. Schlagen Sie Detailinformationen zur Adresse im jeweiligen MOBY-Systemhandbuch oder in Anhang B Adressierung der RFID-Tags (Seite 129) nach.</li></ul>
Data	ASCII-Hex	00...FF (ASCII-String (max. 128KByte ASCII)) <ul style="list-style-type: none"><li>• Nutzdaten, die auf den Tag geschrieben werden sollen.</li></ul>

## 6.4.2 readTagData

Mit dem Befehl können Teilbereiche oder der komplette Tag gelesen werden, soweit die Adressbereiche physikalisch linear auf dem Tag vorhanden sind.

Im Folgenden ist der XML-Befehl für den Lese-Befehl (**readTagData**) beschrieben:

### XML-Befehl

```
<command>
  <readTagData>
    <startAddress>Address</startAddress>
    <dataLength>Datalength</dataLength>
  </readTagData>
</command>
```

Informationen über die Speichergrößen der Tags finden Sie im Kapitel Adressierung der RFID-Tags (Seite 129) .

### XML-Antwort

Im Folgenden ist die XML-Antwort ohne Fehlereintrag (Fehlereinträge sind im Kapitel Fehlermeldungen (Seite 77)) beschrieben:

```
<reply>
  <resultCode>Errorcode</resultCode>
  <readTagData>
    <returnValue>
      <data>Data</data>
    </returnValue>
  </readTagData>
</reply>
```

### Parameter

Parameter	Datentyp	Werte
Address	ASCII-Hex	0000 ... FFFF (4 x ASCII) <ul style="list-style-type: none"> <li>0 bis maximale Länge der (User-Daten -1) Ab dieser Anfangsadresse werden die User-Daten vom Tag gelesen. (Adresse + Datenlänge) muss kleiner sein als die Endadresse.</li> <li>Beachten Sie, dass dieser Parameter tagabhängig ist. Schlagen Sie Detailinformationen zur Adresse im jeweiligen MOBY-Systemhandbuch nach.</li> </ul>
Datalength	ASCII-Hex	0000 ... FFFF (4 x ASCII) Der Inhalt von Datalength bezieht sich auf die Anzahl der zu lesenden Bytes vom Tag. Im Feld "Data" der XML-Antwort wird die doppelte Anzahl von Zeichen übertragen.
Data	ASCII-Hex	max. 128 KByte ASCII pro Befehl

### 6.4.3 initializeTag

Mit diesem Befehl wird der komplette User memory Bereich des Speichers gelöscht bzw. mit einem vorgegebenen Wert überschrieben.

Im Folgenden ist der XML-Befehl für den Befehl zum Initialisieren von Tags (**initializeTag**) beschrieben:

#### XML-Befehl

```
<command>  
  <initializeTag>  
    <memorySize>MemorySize</memorySize>  
    <defaultValue>Value</defaultValue>  
  </initializeTag>  
</command>
```

#### XML-Antwort

Im Folgenden ist die XML-Antwort ohne Fehlereintrag (Fehlereinträge finden Sie im Kapitel Fehlermeldungen (Seite 77)) beschrieben:

```
<reply>  
  <resultCode>ErrorCode</resultCode>  
  <initializeTag/>  
</reply>
```

#### Parameter

Parameter	Datentyp	Werte
MemorySize	ASCII-Hex	0000 ... FFFF (4 x ASCII) <ul style="list-style-type: none"><li>• Speichergröße des zu initialisierenden Tags</li><li>• Beachten Sie, dass dieser Parameter tagabhängig ist. Schlagen Sie Detailinformationen zur Speichergröße im jeweiligen MOBY-Systemhandbuch oder im Anhang B Adressierung der RFID-Tags (Seite 129) nach.</li></ul>
Value	ASCII-Hex	00 ... FF (2 x ASCII) <ul style="list-style-type: none"><li>• Hexwert mit dem der Tag beschrieben wird.</li></ul>

#### 6.4.4 getReaderStatus

Der Befehl liefert im Ergebnis Diagnose-/Statusinformation vom angeschlossenen Reader.

Im Folgenden ist der XML-Befehl für den Reader-Status (**getReaderStatus**) beschrieben:

##### XML-Befehl

```
<command>
  <getReaderStatus>
    <mode>Mode</mode>
  </getReaderStatus>
</command>
```

##### XML-Antwort

Im Folgenden ist die XML-Antwort ohne Fehlereintrag (Fehlereinträge sind im Kapitel Fehlermeldungen (Seite 77)) beschrieben:

```
<reply>
  <resultCode>Errorcode</resultCode>
  <getReaderStatus>
    <returnValue>
      <data>Data</data>
    </returnValue>
  </getReaderStatus>
</reply>
```

##### Parameter

Parameter	Datentyp	Werte
Mode	ASCII-Hex	00 ... FF (2 x ASCII) 01 = Reader-Status 02 = MOBY U (SLG-Diagnose I, Funktionsaufrufe) 03 = MOBY U (SLG-Diagnose II, Fehlermeldungen) 04 = MOBY U (SLG-Diagnose III, identifizierte MDS) 05 = MOBY U (SLG-Diagnose IV, Kommunikationsgüte) 06 = RF300 Reader-Diagnose 07 = RF600 Reader-Diagnose
Data	ASCII-Hex	0000 ... FFFF (ASCII-String (max. 400 x ASCII)) Ein Quittungstelegramm zum Reader-Status wird zurückgeliefert.

##### Quittungstelegramm Reader-Status

Über den Parameter "Data" wird ein Quittungstelegramm über den Reader-Status zurückgeliefert. Je nach eingestelltem "Mode" im Befehl wird ein anderes Quittungstelegramm zurückgeliefert. Alle möglichen Quittungstelegramme finden Sie im Kapitel Befehls- und Quittungstelegramme (Seite 117).

### 6.4.5 getTagStatus

Dieser Befehl liefert im Ergebnis verschiedene Statusinformation von einem im Feld befindlichen Tag.

Im Folgenden ist der XML-Befehl für den Tag-Status (**getTagStatus**) beschrieben:

#### XML-Befehl

```
<command>
  <getTagStatus>
    <mode>Mode</mode>
    <week>Week</week>
    <year>Year</year>
  </getTagStatus>
</command>
```

---

#### Hinweis

##### Nur bei MOBY U

Die kursiv dargestellten XML-Tags *<week>* und *<year>* gelten nur für das MOBY-System MOBY U. Bei anderen MOBY-Systemen müssen diese kursiv dargestellten Tags weggelassen werden!

---

#### XML-Antwort

Im Folgenden ist die XML-Antwort ohne Fehlereintrag (Fehlereinträge sind im Kapitel Fehlermeldungen (Seite 77)) beschrieben:

```
<reply>
  <resultCode>Errorcode</resultCode>
  <getTagStatus>
    <returnValue>
      <data>Data</data>
    </returnValue>
  </getTagStatus>
</reply>
```

#### Parameter

Parameter	Datentyp	Werte
Mode	ASCII-Hex	00 = MOBY U 01 = RF300: Typ- und Schreibschutzstatus 02 = RF300: Diagnosedaten 03 = RF300/MOBY D: Typ- und Schreibschutzstatus für ISO-Tags 04 = RF600: Diagnosedaten
Week	ASCII-Hex	00 ... FF (2 x ASCII)
Year	ASCII-Hex	00 ... FF (2 x ASCII)
Data	ASCII-Hex	0000 ... FFFF (ASCII String: max. 400 x ASCII)

### **Quittungstelegramm Tag-Status**

Über den Parameter "Data" wird ein Quittungstelegramm über den Tag-Status zurückgeliefert. Je nach eingestelltem "Mode" im Befehl wird ein anderes Quittungstelegramm zurückgeliefert. Alle möglichen Quittungstelegramm finden Sie im Kapitel Befehls- und Quittungstelegramme (Seite 117).

### 6.4.6 setAnt

Im Folgenden ist der XML-Befehl für das Ein-/Ausschalten der Antenne(n) (**setAnt**) beschrieben:

#### XML-Befehl

```
<command>
  <setAnt>
    <mode>Mode</mode>
  </setAnt>
</command>
```

#### XML-Antwort

Im Folgenden ist die XML-Antwort ohne Fehlereintrag (Fehlereinträge sind im Kapitel Fehlermeldungen (Seite 77)) beschrieben:

```
<reply>
  <resultCode>Errorcode</resultCode>
  <setAnt/>
</reply>
```

#### Parameter

Parameter	Datentyp	Wert
Mode	ASCII	<b>D/U/RF300:</b> 01 = Antenne einschalten 02 = Standby; Antenne ausschalten <b>RF600:</b> 00 = Antenne 1 aus und Antenne 2 aus 01 = Antenne 1 ein und Antenne 2 aus 02 = Antenne 1 aus und Antenne 2 ein 03 = Antenne 1 und 2 ein

### 6.4.7 heartbeat

Der Befehl heartbeat hat keine Parameter. Die Auswertung von resultCode in der XML-Antwort zeigt, ob die Verbindung bis zum Kommunikationsmodul noch in Ordnung ist. Eine fehlende Verbindung zwischen Kommunikationsmodul und Reader führt zu einem Alarm und/oder einer Fehlermeldung.

Im Folgenden ist der XML-Befehl für die Verbindungsüberwachung (**heartbeat**) beschrieben:

#### XML-Befehl

```
<command>  
  <heartbeat/>  
</command>
```

#### XML-Antwort

Im Folgenden ist die XML-Antwort ohne Fehlereintrag

```
<reply>  
  <resultCode>0000</resultCode>  
  <heartbeat/>  
</reply>
```

#### Siehe auch

Fehlermeldungen (Seite 77)

## 6.5 Asynchrone Telegramme

### 6.5.1 tagPresent

Die Notification "tagPresent" wird nur dann gemeldet, wenn dies vorher im RESET Telegramm entsprechend eingeschaltet wurde. Bei eingeschalteter Anwesenheit wird dieses Telegramm gesendet, wenn ein Tag in das Feld eines Readers kommt oder wenn er es verlässt.

#### XML-Telegramm

```
<notification>
  <id>Sequencenumber</id>
  <origin>Origin</origin>
  <tagPresent>
    <tagCount>TagCount</tagCount>
  </tagPresent>
</notification>
```

#### Parameter

Parameter	Datentyp	Werte
Sequencenumber	ASCII-Hex	0000 ... FFFF (4 x ASCII) Die Sequencenumber ist eine Zahl, die nach dem Einschalten der RF182C automatisch auf 0000 gesetzt wird. Bei jedem Versenden einer Notification auf dem gleichen TCPIP-Kanal wird die Sequencenumber um 1 erhöht. Nach FFFF fängt diese Nummer ebenfalls wieder bei 0000 an.
Origin	ASCII	<ul style="list-style-type: none"> <li>• TRANSM_POINT_1 - Kanal1/Reader1</li> <li>• TRANSM_POINT_2 - Kanal2/Reader2</li> </ul>
TagCount	ASCII-Hex	4 x ASCII 0000 = kein Tag anwesend 0001 = ein Tag befindet sich im Feld des Readers 0002 = zwei Tags befinden sich im Feld des Readers usw. bis max 00FF

## 6.5.2 alarm

Steht kein laufender Befehl an, sendet das Kommunikationsmodul RF182C einen Alarm.

### XML-Telegramm

```
<alarm>
  <id>Sequenznummer</id>
  <origin>Origin</origin>
  <deviceName>Devicename</deviceName>
  <deviceTime>Time</deviceTime>
  <content>
    <code>Errorcode</code>
  </content>
</alarm>
```

### Parameter

Parameter	Datentyp	Werte
Sequenznummer	ASCII-Hex	0000 ... FFFF (4 x ASCII) Die Sequenznummer ist eine Zahl, die nach dem Einschalten der RF182C automatisch auf 0000 gesetzt wird. Bei jedem Versenden eines Alarms auf dem gleichen TCP/IP-Kanal wird die Sequenznummer um 1 erhöht. Nach FFFF fängt diese Nummer ebenfalls wieder bei 0000 an.
Origin	ASCII	0 .. 9, A... Z Ursprungskanal TRANSM_POINT_1 oder TRANSM_POINT_2
DeviceName	ASCII	Name des Geräts (Default: RF182C) Der Device-Name kann maximal 256 Byte lang werden.
Time	ASCII-Hex	00000000...FFFFFFFF (8 x ASCII) Die Zeit des Alarm in Millisekunden Diese Zeit wird beim Einschalten des RF180C auf 0000 zurückgesetzt. Sie kann vom Anwender nicht gesetzt werden. Es ist demnach eine relative Zeit
Errorcode	ASCII-Hex	0000 ... FFFF (4 x ASCII) Aus dem Errorcode geht die Ursache hervor, welche dieses Alarmtelegramm ausgelöst hat. Nähere Informationen zu den Errorcodes finden Sie im Kapitel Fehlermeldungen des RF182C (Seite 78).



## 7.1 Kommunikationsmodul RF182C tauschen

### Ausgangssituation

- Das Kommunikationsmodul RF182C ist montiert. Ein neues Kommunikationsmodul RF182C desselben Typs soll montiert werden.
- RF182C ist verdrahtet und in Betrieb.

### Vorgehensweise

1. Lösen Sie die 4 Befestigungsschrauben am Anschlussblock und ziehen Sie den Anschlussblock vom Kommunikationsmodul.

---

#### Hinweis

Wenn Sie während des Betriebs den Anschlussblock demontieren, dann wird nur die Stromversorgung weitergeschleift. Die Ethernet-Kommunikation wird ab diesem Teilnehmer während des Baugruppentauschs unterbrochen. Zusätzliche Informationen dazu finden Sie im Kapitel Ethernet und Versorgungsspannung weiterschleifen (Seite 31).

---

2. Lösen Sie die geschraubten M12 Steckverbindungen am Kommunikationsmodul zu den Readern.
3. Lösen Sie die Befestigungsschrauben des Kommunikationsmoduls und legen Sie es ab.
4. Setzen Sie das neue Kommunikationsmodul ein und schrauben Sie es fest.
5. Stecken Sie den Anschlussblock auf das neue Kommunikationsmodul und schrauben Sie die 4 Befestigungsschrauben fest.

### Ergebnis

Da die IP-Adresse des Kommunikationsmoduls im Anschlussblock gespeichert bleibt, wird das neue Kommunikationsmodul RF182C vom Ethernet-Controller in den Datenverkehr aufgenommen.

---

#### Hinweis

Wird neben dem Basisgerät auch der Anschlussblock getauscht, so kann es vorkommen, dass das RF182C nicht mehr automatisch anläuft. Fahren Sie in diesem Fall mit dem folgenden Punkt fort.

---

### **Was tun, wenn das RF182C nicht mehr ansprechbar ist**

Wenn neben dem Basismodell auch der Anschlussblock getauscht wird, kann es vorkommen, dass das RF182C nicht mehr ansprechbar ist. Das wird durch eine permanent leuchtende oder blinkende BF LED angezeigt.

Überprüfen Sie in diesem Fall die Netzwerkprojektierung. Laden Sie (z. B. mit dem PST-Tool) die notwendigen Netzwerkparameter in das RF182C.

Überprüfen Sie über den Webserver die Diagnosemeldungen bzw. kontrollieren Sie die Einstellungen der IP-Adresse bzw. die Einstellung der Portnummer.

### **Siehe auch**

Parametrieren (Seite 37)

## 7.2 Firmware-Update

Die Firmware für das Kommunikationsmodul RF182C ist über die Ethernet-Schnittstelle aktualisierbar. Sie können das Firmware-Update über den Webserver des Kommunikationsmoduls starten.

### Voraussetzungen

- Das Kommunikationsmodul ist über Ethernet an den PC angeschlossen.
- Beenden Sie alle Applikationen bevor Sie das Firmware-Update starten.

### Vorgehensweise

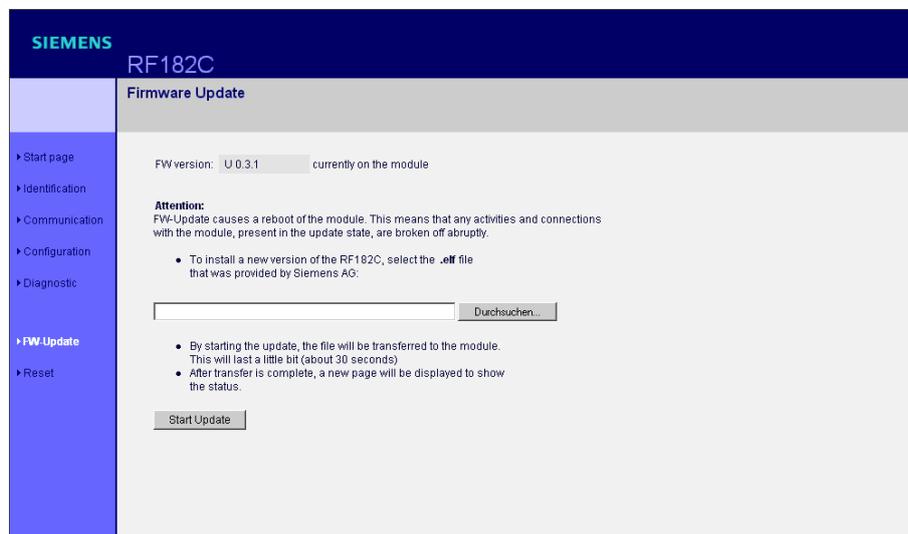


Bild 7-1 Firmware-Update

1. Speichern Sie die Update-Datei (z. B. "RF182C\_V\_2\_0\_0.elf"), die Sie von Siemens erhalten haben, in einem gewünschten Verzeichnis ab.
2. Geben Sie die IP-Adresse des Kommunikationsmoduls in das Adressfeld Ihres Browsers ein.  
Der Webserver des Kommunikationsmoduls öffnet sich.
3. Klicken Sie auf die Schaltfläche "Durchsuchen".
4. Wählen Sie Update-Datei (RF182C\_V\_2\_0\_0.elf).
5. Starten Sie das Firmware-Update über die Schaltfläche "Start Update".

## 7.3 Reader-Update

In Vorbereitung.

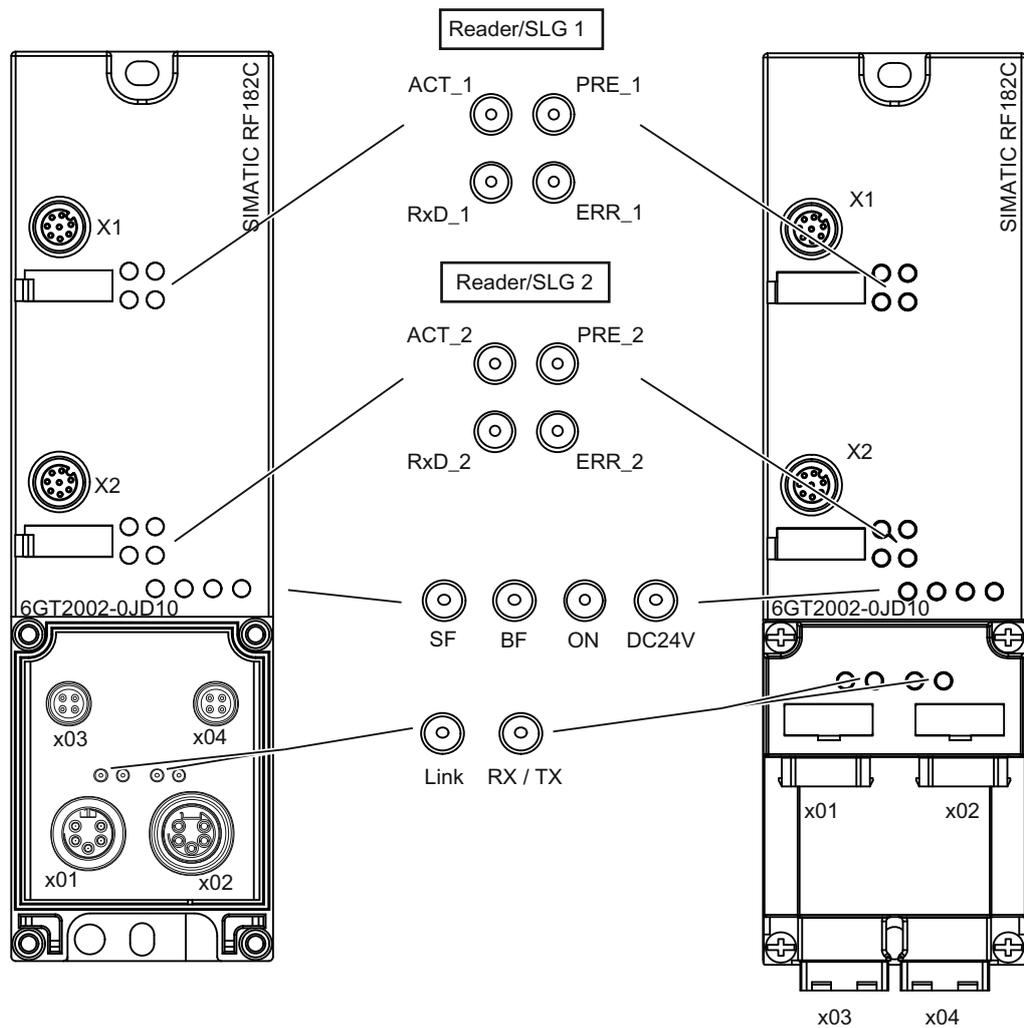


## 8.1 Diagnose mittels Leuchtdioden

Im folgenden Bild sind die Leuchtdioden des RF182C detailliert dargestellt.

Mit Anschlussblock M12, 7/8"

Mit Anschlussblock Push Pull



8.1 Diagnose mittels Leuchtdioden

Tabelle 8- 1 Status-LEDs für RF182C

LED	Bedeutung
ON	Leuchtet, wenn das RF182C den Anlauf fehlerfrei beendet hat.
DC 24 V	Leuchtet, wenn am RF182C die 24 V-Versorgungsspannung angeschlossen ist.
ACT_1, ACT_2	reserviert
ERR_1, ERR_2	Ein Blinkmuster zeigt den zuletzt aufgetretenen Fehler an. Die Blinkmuster sind im Kapitel Fehlermeldungen (Seite 77) beschrieben.
PRE_1, PRE_2	Zeigt die Anwesenheit eines Tag / MDS an.
RxD_1, RxD_2	Zeigt eine laufende Kommunikation zum Reader / SLG an. Störungen am Reader / SLG können diese Anzeige ebenfalls leuchten lassen.

Tabelle 8- 2 LED-Anzeige für Ethernet-Diagnose

BF	SF	Fehlerursache	Fehlerbehandlung
An	–	<ul style="list-style-type: none"> <li>Kommunikationsmodul befindet sich im Anlauf.</li> <li>Kein Kabel gesteckt.</li> </ul>	Überprüfen Sie das Ethernet-Kabel.
blinkt	An	<ul style="list-style-type: none"> <li>Es besteht keine TCP-Verbindung zum Ethernet-Controller.</li> <li>Kein Client hat sich verbunden.</li> </ul>	Überprüfen Sie Ihre Ethernet-Konfiguration mit dem PST-Tool.
Aus	An	<ul style="list-style-type: none"> <li>Es liegt ein Fehler vor.</li> <li>Baugruppe hat keinen Reset- oder comDevSetConfig-Befehl auf einem Kanal empfangen.</li> </ul>	Schicken Sie einen Reset- oder comDevSetConfig-Befehl.
Aus	Aus	<ul style="list-style-type: none"> <li>Normalbetrieb</li> </ul>	–

– = Status ist nicht relevant

Tabelle 8- 3 LEDs am Anschlussblock

Link (grün)	Tx / Tx (gelb)	Bedeutung
Aus	Aus	Keine physikalische Verbindung über Ethernet.
Ein	Aus	Physikalische Verbindung über Ethernet, kein Datenverkehr
Ein	Blinkt	Physikalische Verbindung über Ethernet mit Datenverkehr
Aus	Ein	Kurzzeitiger Zustand beim Einschalten

Die Tabelle gilt für den linken und rechten Ethernet-Anschluss gleichermaßen.

An den Leuchtdioden PRE, ERR, ACT, SF und ON werden weitere Betriebszustände des Kommunikationsmoduls angezeigt:

Tabelle 8- 4 LED-Anzeige für Betriebszustände

ON	BF	SF	PRE_1	ERR_1	ACT_1	PRE_2	ERR_2	ACT_2	Beschreibung
Aus	Aus	Aus	Aus	Aus	Aus	Aus	Aus	Aus	Hochlauf aktiv
An	An	An	An	An	An	An	An	An	LED-Test im Anlauf (Start Ethernet)
Aus	Aus	An	An	An	Aus	An	An	Aus	interner Fehler
Aus	Aus	An	An	Aus	An	An	Aus	An	Checksummenfehler der Firmware
Aus	Aus	An	Aus	blinkt langsam	Aus	Aus	blinkt langsam	Aus	Firmware-Update (blinkt mit jedem beschriebenen Bereich)
An	blinkt	An	Aus	blinkt 1 x gem. Tab. 8-1	Aus	Aus	blinkt 1 x gem. Tab. 8-1	Aus	Hochlauf nicht parametrierbar. Kein Client hat sich verbunden.



# Fehlermeldungen

## 9.1 Antwort ohne Fehlereintrag

Im Folgenden ist zunächst die XML-Antwort ohne Fehlereintrag beschrieben:

### XML-Antwort

```
<reply>
  <resultCode>0000</resultCode>
  <Name des abgesetzten Befehls z. B. reset/>
</reply>
```

## 9.2 Antwort mit Fehlereintrag

Im Folgenden ist zunächst die XML-Antwort mit Fehlereintrag beschrieben:

### XML-Antwort

```
<reply>
  <resultCode>Errorcode</resultCode>
  <Name des abgesetzten Befehls z. B. reset/>
</reply>
```

Die folgende Tabelle beschreibt die möglichen Fehlercodes (**resultCodes**). Die Fehlercodes werden in 4 Bytes codiert.

## 9.3 Fehlermeldungen des RF182C

Tabelle 9- 1 Fehlermeldungen des RF182C über die Variable "resultCode"

Errorcode ASCII-HEX	Blinken der ERR-LED 1)	Beschreibung
0000	–	kein Fehler Standardwert, wenn alles o.k. ist
	1x	kein Fehler Kommunikationsmodul hat einen Hochlauf ausgeführt und wartet auf einen RESET-Befehl.
0001	2x	Anwesenheitsfehler: Der MDS ist aus dem Übertragungsfenster des SLG gefahren. Der MOBY-Befehl wurde nur teilweise abgearbeitet. Lesebefehl: Es werden keine Daten an den Client geliefert. Schreibbefehl: Der MDS, der gerade das Feld verlassen hat, beinhaltet einen unvollständigen Datensatz. <ul style="list-style-type: none"> <li>Arbeitsabstand von SLG zu MDS wird nicht eingehalten</li> <li>Projektierungsfehler: Zu bearbeitender Datensatz ist zu groß (im dynamischen Betrieb)</li> </ul> Der folgende Befehl wird automatisch auf dem nächsten MDS ausgeführt. Es ist ein Lese-, oder Schreib-Befehl möglich. <ul style="list-style-type: none"> <li>mit Timeout: kein MDS im Feld</li> </ul>
0002	2x	Anwesenheitsfehler: Ein MDS ist an einem SLG vorbeigefahren und wurde mit keinem MOBY-Befehl bearbeitet. Diese Fehlermeldung wird nicht sofort gemeldet. Vielmehr wartet das Kommunikationsmodul auf den nächsten Befehl (Lesen, Schreiben). Dieser Befehl wird sofort mit diesem Fehler beantwortet. Dies bedeutet, dass ein Lese- oder Schreibbefehl nicht bearbeitet wird. Erst der nächste Befehl wird wieder normal vom Kommunikationsmodul ausgeführt. Ein RESET-Befehl vom Client setzt diesen Fehlerzustand ebenfalls zurück. Im Reset-Parameter option_1 ist das Bit 2 gesetzt oder ein reset-Befehl wurde abgeschickt und es befindet sich kein MDS im Übertragungsfenster.
0003	3x	Fehler in der Verbindung zum SLG; das SLG antwortet nicht. <ul style="list-style-type: none"> <li>Kabel zwischen Kommunikationsmodul und SLG ist falsch verdrahtet oder Kabelbruch</li> <li>24 V-Versorgungsspannung ist nicht angeschlossen oder abgeschaltet bzw. kurzzeitig ausgefallen</li> <li>automatische Sicherung auf dem Kommunikationsmodul hat angesprochen</li> <li>Hardware defekt</li> <li>Anderes SLG ist in der Nähe und ist aktiv geschaltet</li> <li>Störeinkopplung auf SLG-Leitung</li> <li>nach der Fehlerbehebung RESET-Befehl durchführen</li> </ul>
0004	4x	Fehler im Speicher des MDS Der MDS wurde noch nie beschrieben oder hat durch einen Ausfall der Batterie seinen Speicherinhalt verloren. <ul style="list-style-type: none"> <li>MDS wechseln (wenn das Batterie-Bit gesetzt ist)</li> <li>MDS mit dem STG initialisieren</li> <li>MDS neu initialisieren (siehe Kapitel "Parametrierung der Befehle").</li> </ul>

Errorcode ASCII-HEX	Blinken der ERR-LED <sup>1)</sup>	Beschreibung
0005	5x	unbekannter Befehl Der Client gibt einen nicht interpretierbaren Befehl an das Kommunikationsmodul. <ul style="list-style-type: none"> <li>• Überprüfen Sie den XML Befehl</li> <li>• Der MDS hat Adressfehler gemeldet</li> </ul>
0006	6x	Feldstörung am SLG Das SLG empfängt Störimpulse aus der Umgebung. <ul style="list-style-type: none"> <li>• Der Abstand zwischen zwei SLG ist zu klein und entspricht nicht den Projektierungsrichtlinien</li> <li>• Das Verbindungskabel zum SLG wird gestört, ist zu lang oder entspricht nicht der Spezifikation</li> <li>• MOBY U: MDS hat während der Kommunikation das Feld verlassen.</li> <li>• MOBY U: Die Kommunikation zwischen SLG und MDS wurde durch Störeinflüsse abgebrochen (z. B. Person/Fremdkörper bewegt sich zwischen SLG und MDS).</li> </ul>
0007	7x	zu viele Sendefehler Der MDS konnte den Befehl oder die Schreibdaten vom Kommunikationsmodul nach mehreren Versuchen nicht richtig empfangen. <ul style="list-style-type: none"> <li>• Der MDS steht genau im Grenzbereich des Übertragungsfensters</li> <li>• Die Datenübertragung zum MDS wird durch externe Störungen beeinflusst</li> </ul>
0008	8x	CRC-Sendefehler <ul style="list-style-type: none"> <li>• Der Mithörempfang hat beim Senden einen Fehler erkannt.               <ul style="list-style-type: none"> <li>– Ursache wie bei Fehler 0006</li> </ul> </li> <li>• Der MDS meldet sehr oft CRC-Fehler.               <ul style="list-style-type: none"> <li>– Der MDS steht im Grenzbereich des SLG</li> <li>– Der MDS und/oder das SLG haben einen Hardwaredefekt</li> </ul> </li> </ul>
0009	9x	nur bei Initialisierung: CRC-Fehler beim Quittungsempfang vom MDS <ul style="list-style-type: none"> <li>• Ursache wie bei Fehler 0006</li> </ul>
000A	10x	nur bei Initialisierung: MDS kann den Initialisierungs-Befehl nicht durchführen <ul style="list-style-type: none"> <li>• MDS ist defekt</li> </ul>
000B	11x	MOBY U: Speicher des MDS ist nicht korrekt lesbar
000C	12x	Speicher des MDS kann nicht beschrieben werden <ul style="list-style-type: none"> <li>• Speicher des MDS ist defekt</li> <li>• EEPROM-MDS wurde zu oft beschrieben und hat sein Lebensende erreicht</li> </ul>
000D	13x	Adressfehler Der Adressbereich des MDS wird überschritten. <ul style="list-style-type: none"> <li>• Überprüfen Sie den XML Befehl</li> <li>• Der MDS ist nicht vom richtigen Typ</li> <li>• RF300: Versuch des schreibenden Zugriffs auf schreibgeschützte Bereiche (OTP-Bereich)</li> </ul>

Errorcode ASCII-HEX	Blinken der ERR-LED <sup>1)</sup>	Beschreibung
000E	14x	ECC-Fehler (nur möglich, wenn ECC_mode = TRUE) Die Daten können nicht vom MDS gelesen werden. <ul style="list-style-type: none"> <li>• Daten des MDS sind verloren gegangen (MDS defekt)</li> <li>• Der MDS wurde nicht mit ECC-Treiber initialisiert <ul style="list-style-type: none"> <li>– MDS initialisieren</li> </ul> </li> <li>• MDS mit EEPROM hat sein Lebensende erreicht; die Daten sind verloren gegangen <ul style="list-style-type: none"> <li>– MDS austauschen</li> </ul> </li> <li>• → Beim Beschreiben wurde der MDS aus dem Feld bewegt <ul style="list-style-type: none"> <li>– Der MDS ist nicht richtig positioniert</li> </ul> </li> <li>• → Befehl zum Kommunikationsmodul wurde vom Anwender falsch aufgesetzt</li> </ul>
000F	1x	Hochlaufmeldung eines am Kommunikationsmodul angeschlossenen Readers <ul style="list-style-type: none"> <li>• RESET durchführen</li> </ul>
0011	–	Kurzschluss oder Überlastung der 24 V-Ausgänge (Fehlercode, Anwesenheit) <ul style="list-style-type: none"> <li>• Der betroffene Ausgang wird abgeschaltet</li> <li>• Bei Gesamtüberlastung werden alle Ausgänge abgeschaltet</li> <li>• Ein Rücksetzen ist nur durch das Aus- und Wiedereinschalten der 24 V-Versorgungsspannung möglich</li> <li>• Anschließend RESET starten</li> </ul>
0012	18x	Interner Kommunikationsmodul-Kommunikationsfehler. <ul style="list-style-type: none"> <li>• Stecker-Kontaktproblem auf dem Kommunikationsmodul</li> <li>• Hardware des Kommunikationsmodul hat einen Defekt <ul style="list-style-type: none"> <li>– Kommunikationsmodul zur Reparatur einschicken</li> </ul> </li> <li>• Nach Fehlerbehebung RESET starten</li> </ul>
0013	19x	Es sind nicht genügend Puffer im Kommunikationsmodul/SLG U für die Zwischenspeicherung des Befehls vorhanden.
0014	20x	Kommunikationsmodul/SLG-interner Fehler. <ul style="list-style-type: none"> <li>• Programmablauffehler auf dem Kommunikationsmodul</li> <li>• Versorgungsspannung des Kommunikationsmodul aus- und wiedereinschalten</li> <li>• Nach Fehlerbehebung RESET starten</li> <li>• MOBY U: Watchdog-Fehler auf SLG</li> </ul>
0015	21x	Fehlerhafte Parametrierung des Kommunikationsmodul/SLG <ul style="list-style-type: none"> <li>• Parametrierung überprüfen</li> <li>• RESET-Befehl ist falsch parametriert</li> <li>• Nach einem Hochlauf hat das Kommunikationsmodul noch keinen RESET erhalten</li> </ul>
0016	22x	<ul style="list-style-type: none"> <li>• Das Kommunikationsmodul kann den Befehl nicht bearbeiten.</li> <li>• Client-Befehl (z. B. Lesen) mit zu großer Länge der Nutzdaten aufgesetzt</li> </ul>
0017	23x	Kommunikationsfehler <ul style="list-style-type: none"> <li>• Client-Befehl, der zu diesem Fehler führt, überprüfen</li> <li>• Nach Fehlerbehebung RESET starten</li> </ul>
0018	–	Es ist ein Fehler aufgetreten, der mit einem RESET quittiert werden muss. <ul style="list-style-type: none"> <li>• Der RESET-Befehl ist fehlerhaft</li> <li>• Nach Fehlerbehebung RESET absetzen</li> </ul>

Errorcode ASCII-HEX	Blinken der ERR-LED <sup>1)</sup>	Beschreibung
0019	25x	<p>vorheriger Befehl ist aktiv bzw. Pufferüberlauf</p> <p>Der Anwender hat an das Kommunikationsmodul einen neuen Befehl geschickt, obwohl der letzte Befehl noch aktiv war.</p> <ul style="list-style-type: none"> <li>• aktiver Befehl kann nur durch RESET abgebrochen werden</li> <li>• zwei Client-Aufrufe wurden mit den gleichen Parametern parametrisiert</li> <li>• Nach Fehlerbehebung RESET starten</li> <li>• Beim Arbeiten mit Befehlswiederholung werden keine Daten vom MDS abgeholt. Der Datenpuffer im Kommunikationsmodul ist übergelaufen. Es sind MDS-Daten verloren gegangen.</li> </ul>
001C	28x	<p>Die Antenne am SLG ist abgeschaltet. In diesem Zustand wurde ein MDS-Befehl zum Kommunikationsmodul gestartet.</p> <ul style="list-style-type: none"> <li>• mit dem Befehl "Antenne Ein/Aus" die Antenne einschalten</li> <li>• die Antenne ist eingeschaltet (ausgeschaltet) und hat einen weiteren Einschaltbefehl (Ausschaltbefehl) erhalten</li> </ul>
001D	–	<p>Es sind mehr MDS im Übertragungsfenster, als das SLG gleichzeitig bearbeiten kann.</p> <ul style="list-style-type: none"> <li>• mit Client kann nur 1 MDS zu einer Zeit bearbeitet werden</li> </ul>
001E	30x	<p>Fehler beim Bearbeiten der Funktion</p> <ul style="list-style-type: none"> <li>• Kommunikationsmodul Hardware defekt: Bei RESET erhält das Kommunikationsmodul falsche Daten</li> </ul>
001F	–	<p>Laufender Befehl durch RESET abgebrochen</p> <ul style="list-style-type: none"> <li>• Die Kommunikation mit dem MDS wurde mit RESET abgebrochen</li> <li>• Dieser Fehler kann nur bei einem RESET zurückgemeldet werden</li> </ul>

<sup>1)</sup> Das Blinken der ERR-LED kann entweder auf dem Kommunikationsmodul oder auf dem Reader erfolgen.

Tabelle 9- 2 Fehlermeldungen des RF182C

Fehlercode	Blinken der ERR-LED	Beschreibung
3214	-	Fatal Error. Interner Firmwarefehler des Kommunikationsmodules <ul style="list-style-type: none"> <li>• Versorgungsspannung des Kommunikationsmodules aus- und wiedereinschalten</li> <li>• gegebenenfalls FW-Update mit neuer Version durchführen.</li> </ul>
3221	-	Konfigurierung erforderlich <ul style="list-style-type: none"> <li>• Konfigurationsdaten zum Kommunikationsmodul schicken bzw. überprüfen</li> </ul>
3222	-	Konfigurierungskonflikt des Kommunikationsmoduls. <ul style="list-style-type: none"> <li>• Das Konfigurationsmodul wurde bereits konfiguriert, bzw. läuft im Defaultmodus.</li> </ul>
3223	-	Konfigurierung fehlerhaft <ul style="list-style-type: none"> <li>• Konfigurationsdaten zum Kommunikationsmodul überprüfen</li> </ul>
3321	-	Interner Verarbeitungsfehler <ul style="list-style-type: none"> <li>• Versorgungsspannung des Kommunikationsmoduls aus- und wiedereinschalten</li> <li>• gegebenenfalls FW-Update mit neuer Version durchführen</li> </ul>
3322	-	Bufferüberlauf im Kommunikationsmodul <ul style="list-style-type: none"> <li>• Der Anwender schickt zu viele Befehle hintereinander.</li> <li>• Das Kommunikationsmodul kann nicht schnell genug Daten zum Anwender übertragen, bzw. Anwender nimmt nicht schnell genug Daten entgegen.</li> </ul>
3323	-	Bufferüberlauf des Notifikationbuffers <ul style="list-style-type: none"> <li>• Ereignisse, die über Notifikationstelegramme mitgeteilt werden (Anwesenheit) treffen zu schnell hintereinander ein.</li> <li>• Das Kommunikationsmodul kann nicht schnell genug Daten zum Anwender übertragen, bzw. Anwender nimmt nicht schnell genug Daten entgegen.</li> </ul>
3324	-	Bufferüberlauf des Alarmbuffers <ul style="list-style-type: none"> <li>• Ereignisse, die über Alarmbuffer mitgeteilt werden (Fehlermeldungen) treffen zu schnell hintereinander ein.</li> <li>• Das Kommunikationsmodul kann nicht schnell genug Daten zum Anwender übertragen, bzw. Anwender nimmt nicht schnell genug Daten entgegen.</li> </ul>
3325	-	Fehler in der internen Datenverarbeitung des Kommunikationsmoduls <ul style="list-style-type: none"> <li>• Reset-Befehl starten.</li> </ul>
3326	-	Fehler des internen Readerschnittstelle des Kommunikationsmoduls <ul style="list-style-type: none"> <li>• Reset-Befehl starten.</li> </ul>
3417	-	Längenfehler beim Empfang von Daten über die TCP/IP Verbindung. - Prüfen ob die TCP/IP Datenübertragung auf der Leitung korrekt verläuft
3421	-	Fehler in der Konfiguration der Verbindung <ul style="list-style-type: none"> <li>• Verbindung wurde nicht konfiguriert</li> <li>• Zulässige Anzahl der Verbindungen wurde überschritten</li> <li>• Konfiguration überprüfen bzw. korrigieren.</li> </ul>
3422	-	Fehler beim Senden von Daten über die TCP/IP Verbindung - Prüfen ob die TCP/IP Datenübertragung auf der Leitung korrekt verläuft
3423	-	Timeout beim Empfang von Daten über die TCP/IP Verbindung. <ul style="list-style-type: none"> <li>• Die Datenübertragung eines Befehls von Anwenderseite schneller machen.</li> </ul>

Fehlercode	Blinken der ERR-LED	Beschreibung
3424	-	Fehler beim Empfang des TCP-Kanals <ul style="list-style-type: none"> <li>• Verbindung wurde abgebrochen</li> <li>• TCP/IP Protokollfehler</li> <li>• Erlaubte Verbindungsanzahl an einem Port wurde überschritten</li> <li>• Prüfen der Socket-Programmierung auf Anwenderseite bzw. Fehlersuche mit Netzwerk- Analysator (Wireshark).</li> </ul>
3425	-	Bufferüberlauf Empfang <ul style="list-style-type: none"> <li>• Vorherige Befehle sind noch aktiv</li> <li>• Prüfen ob maximale Anzahl möglicher Befehle überschritten wurde.</li> </ul>
3521	-	Falsche UID <ul style="list-style-type: none"> <li>• Die Länge der UID ist nicht korrekt</li> <li>• Es ist eine UID erforderlich, die nicht vom Anwender übertragen wurde</li> <li>• UID prüfen</li> </ul>
3523	-	Konvertierungsfehler <ul style="list-style-type: none"> <li>• Überprüfen Sie, ob einzelne Zeichen der Nutzdaten eines Befehls den zulässigen Bereich für ASCII Hex nicht überschreiten. ( Bereich 0 – 9 bzw. A – F ). Die Anzahl der Zeichen muss gerade Zahl sein.</li> </ul>
3550	-	Fehler im Aufbau der XML-Struktur eines Befehlsstelegrammes. <ul style="list-style-type: none"> <li>• Befehlsaufbau überprüfen</li> </ul>

---

### Hinweis

#### Vorgehensweise beim Fehlerhandling

Wenn Fehlermeldungen auftreten, führen Sie einen Reset-Befehl aus, um die Fehler zu beheben.

Wenn diese Vorgehensweise keine Abhilfe schafft, schalten Sie die Baugruppe spannungslos.

---

## 9.4 Diagnose über Webserver

### 9.4.1 I&M Datensätze speichern/auslesen

Über Identification&Maintenance-Datensätze können Sie interne Informationen auf der Baugruppe speichern und bei Bedarf wieder abrufen.

#### Datensätze speichern

Über das Menü "Identification" im Register "Settings" können verschiedene Einstellungen vorgenommen und auf dem Kommunikationsmodul abgespeichert werden. Über die Schaltfläche "transmit and save" können Sie die Einstellungen abspeichern.

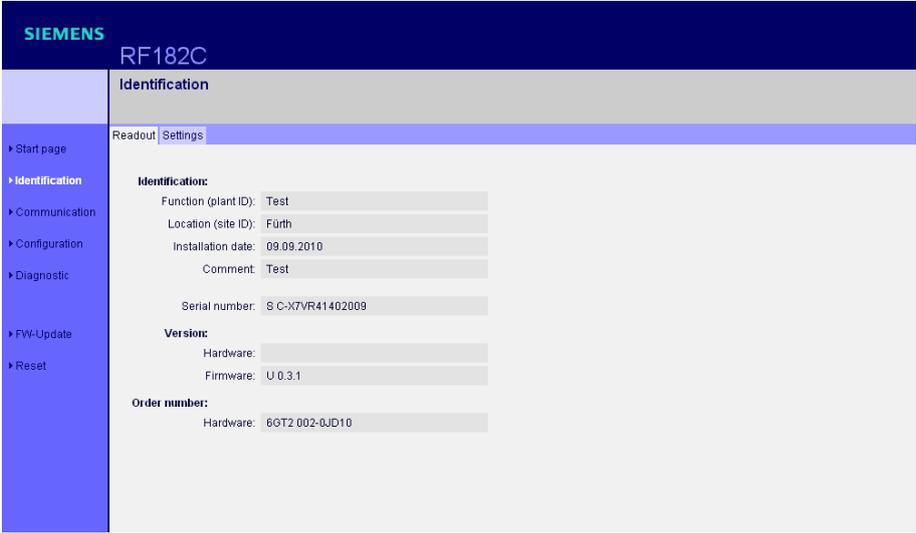
The screenshot shows the SIEMENS RF182C web interface. The top header is dark blue with the SIEMENS logo and the model name RF182C. Below the header, there is a navigation bar with 'Readout' and 'Settings' tabs. A left sidebar contains a menu with options: Start page, Identification (highlighted), Communication, Configuration, Diagnostic, FW-Update, and Reset. The main content area is titled 'Identification' and contains a form with the following fields:

- Funktion:  max. 32 Characters
- Location:  max. 22 Characters
- Installation date:  e.g. YYYY-MM-DD hh:mm, max. 16 Characters
- Comment:  max. 54 Characters

At the bottom of the form is a button labeled 'transmit and save' with the text 'Inputs stored persistently on the RF182C' next to it.

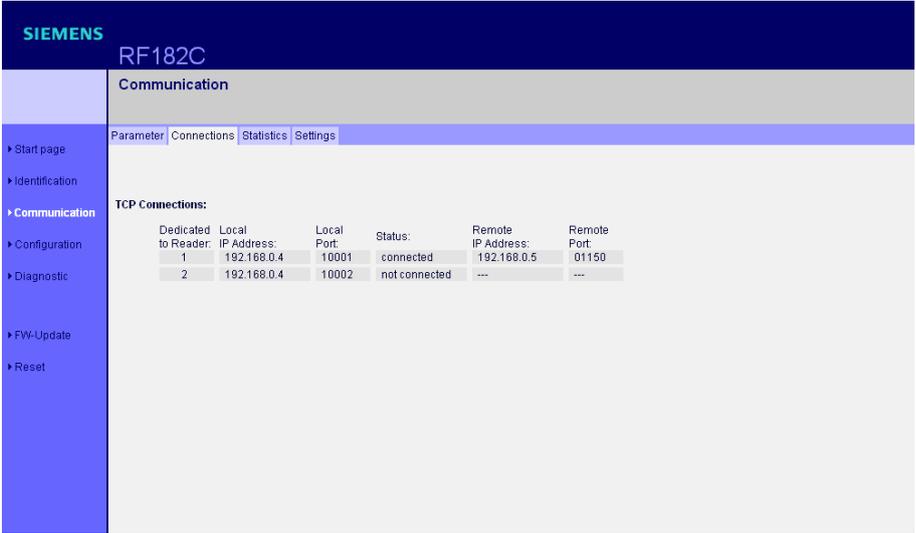
### Datensätze auslesen

Um die Datensätze von des Kommunikationsmoduls auszulesen, wechseln Sie in das Menü "Identification" in das Register "Readout".



### 9.4.2 Kommunikationsstatus abfragen

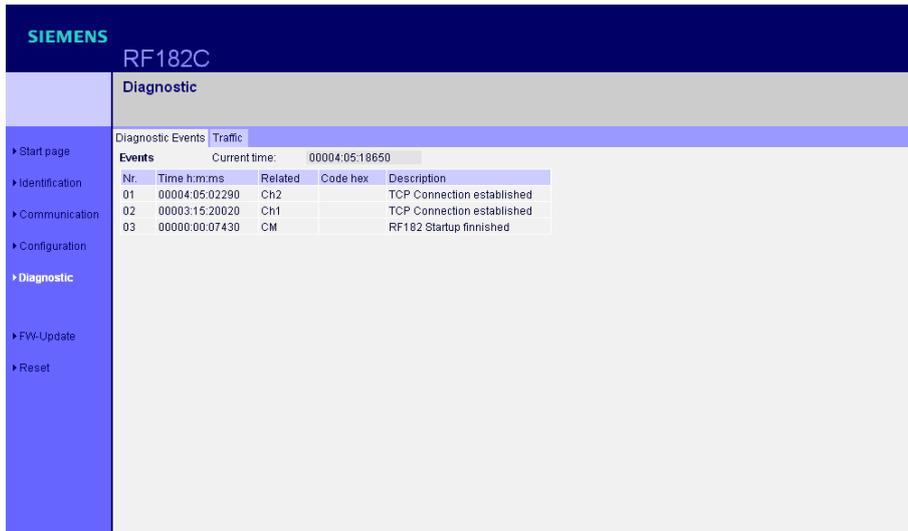
Über das Menü "Communication" im Register "Connection" können Sie den Kommunikationsstatus des Kommunikationsmoduls RF182C abfragen. Sie sehen den Zustand der Anwenderverbindung.



### 9.4.3 Ereignis- und Telegrammübersicht

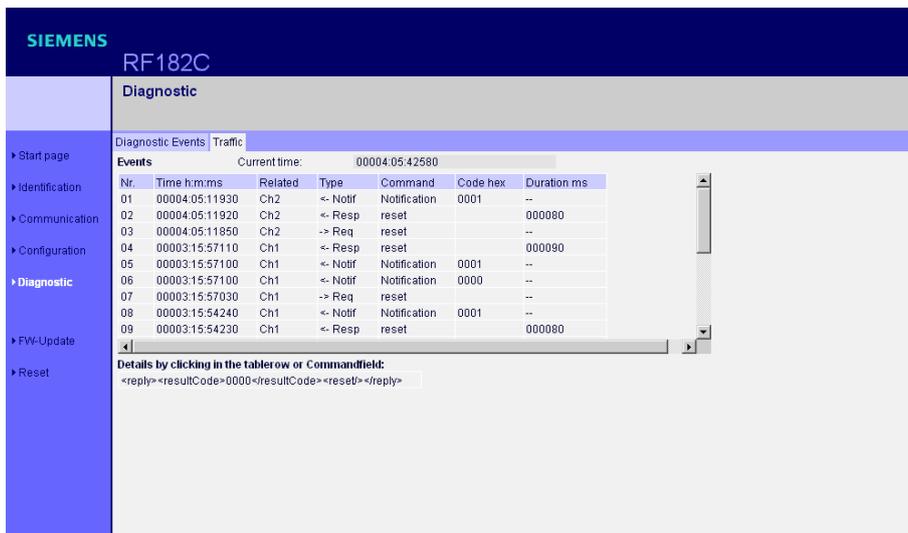
#### Ereignisübersicht

Über das Menü "Diagnostic" im Register "Events" können Sie die Ereignisse der Baugruppe abfragen.



#### Telegrammübersicht

Über das Menü "Diagnostic" im Register "Traffic" können Sie die letzten zwanzig Telegramme des Kommunikationsmoduls abfragen.



## Beispiele/Applikationen

### 10.1 Allgemeine Grundlagen der Socketprogrammierung beispielhaft in C

#### 10.1.1 Voraussetzungen zur Socketprogrammierung

##### Definition

Unter Socket wird der Endpunkt einer Kommunikation verstanden. Alle modernen Betriebssysteme verfügen heute über ein Socket-Applikations-Interface. Diese Software-Schnittstelle ermöglicht einen Zugriff auf den TCP/IP-Stack auf dem verwendeten Betriebssystem.

##### Voraussetzungen

- Es muss ein Betriebssystem/Programmiersprache verwendet werden, das Netzwerkprogramme mit Sockets unterstützt.
- Sockets werden unterstützt von den Betriebssystemen Microsoft Windows 95/98/ME/2000/XP/Vista, Linux, Unix. Andere Betriebssysteme müssen auf die Verwendbarkeit von Netzwerkprogrammen betrachtet werden.
- Sockets werden von vielen Programmiersprachen verwendet (z. B. C, C++, C#, Delphi, VB, Java). Andere Programmiersprachen müssen auf die Verwendbarkeit von Netzwerkprogrammen betrachtet werden.

### 10.1.2 Grundprinzip Client / Server

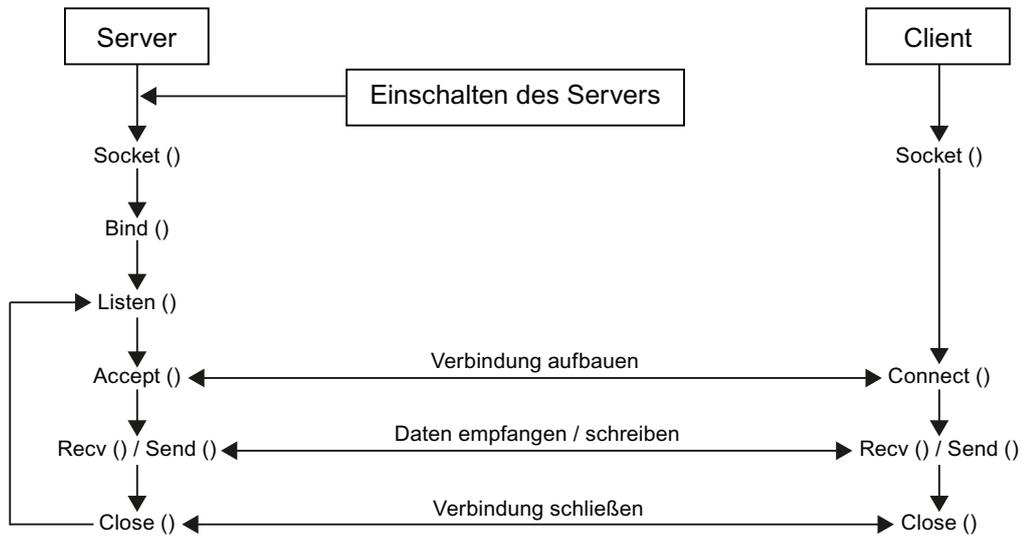


Bild 10-1 Grundprinzip Client/Server

### 10.1.3 Wichtige Grundbefehle

Befehle/Funktionen	Beschreibung
Socket()	Initialisierung und Parameterübergabe eines neuen Sockets
Bind()	Zuweisung des Anwenders (IP-Adresse und Port-Nummer) zu einem Socket
Listen()	Bereitschaft zur Verbindungsaufnahme
Connect()	Funktion der Verbindungsaufnahme vom Client über TCP/IP zum Server
Accept()	Wartefunktion vom Server bis eine Verbindungsaufnahme von einem Client eintrifft
Send()	Senden von Daten
Recv()	Empfangen von Daten
Close()	Schliessen des Sockets nach Beendigung der Datenübertragung. Beim Arbeiten mit dem RF182C geschieht ein Close normalerweise nur beim Herunterfahren der Anlage.

#### Hinweis

Es wurden hier keine Parameter und Rückgabewerte der einzelnen Funktionen aufgeführt, da diese sich in den Betriebssystemen/Programmiersprachen unterscheiden können.

Die Namen der Funktionen dienen zur Veranschaulichung, sie können sich in den Betriebssystemen/Programmiersprachen unterscheiden. Diese Auszüge der Funktionen beschränkt sich auf die Grundprinzipien der Client-Server Darstellung – es wird hier nur ein Teilauszug der Funktionen wiedergegeben.

### 10.1.4 Auszüge eines Programmierbeispiels eines Clients in C / Betriebssystem Windows

Unter Windows muss die Header-Datei "winsock.h" bzw. die Bibliothek "wsock32.lib" mit eingebunden werden.

Vor Aufruf der Windows Sockets müssen diese initialisiert werden.

#### **/\*Init Windows Sockets\*/**

```
{
WSADATA wsadata;

    if( WSASStartup( MAKEWORD( 1,1 ), &wsadata ) == 0 )
    {
        /*Initialisierung erfolgreich*/
    }
    else
    {
        /*Fehler bei der Initialisierung*/
    }
}
```

#### **/\*Auszug aus der Main-Funktion\*/**

```
SOCKET Client;          // Variable für Socket Handle
SOCKADDR_IN adr;       // Variable zum Speichern der Zielinformationen
char caBuf[ 1500 ];    // in dieser Variable wird der XML-Befehl bzw. das XML-Ergebnis
                       // gespeichert.
int nLen;              // Länge der Sende/Empfangsdaten
```

#### **/\*Initialisierung der Zielinformationen vor dem Verbindungsaufbau\*/**

```
adr.sin_family = AF_INET;    // Auswahl der Adressfamilie / Internet
//verbindungsorientiert
adr.sin_port =              // Zuweisung Port Nummer:10001
adr.sin_addr.s_addr =      // Zuweisung der IP-Adresse: 192.168.0.100
/* Anlegung eines Sockets - Funktion liefert den Handle vom Socket zurück
SOCK_STREAM - verbindungsorientiertes Protokoll TCP*/
Client = socket( AF_INET, SOCK_STREAM, 0 );
```

### **/\*Verbindungsaufbau\*/**

```
if( connect(Client, (SOCKADDR*)&adr, sizeof( adr ) ) < 0)
{
    //Fehler aufgetreten
}
else
{
    //Verbindung OK
}
```

### **/\*Senden von Daten\*/**

```
nLen = send( Client, caBuf, nLen, 0 );
if( nLen > 0 )
{
    // Daten wurden erfolgreich gesendet
}
else
{
    // Fehler beim Senden
}
```

### **/\*Empfangen von Daten\*/**

```
nLen = recv( Client, caBuf, sizeof( caBuf ) - 1, 0 );
if( nLen > 0 )
{
    // Es wurden Daten empfangen
    // im Array caBuf können die Daten gelesen werden
}
else
{
    // Fehler beim Empfang
// ggf. Verbindungsabbau
}
```

### **/\*Verbindung schließen\*/**

```
closesocket( Client );
```

## 10.2 Anwenderapplikation RF182C

Zur komfortablen Bedienung steht Ihnen eine Applikation für den PC zur Verfügung. Die Applikation ist erstellt in C# und ist direkt ablauffähig auf jedem Windows PC. Sie können in Ihrer Entwicklungsumgebung den PC laden, modifizieren und erweitern. Diese Applikation kann als Basis für Ihre Anwendung dienen.

Sie finden die Applikation auf der RFID-CD "Software&Documentation", ab Ausgabestand 2009. Folgen Sie auf der Bedienoberfläche den Links "CM/ASM > RF182C > Demo". Sie können das Programm direkt starten.

### Voraussetzungen

- Betriebssystem: Windows XP
- Entwicklungsumgebung: Microsoft Visual Studio 2008

### Vorgehensweise

Die Applikation ist direkt über die Oberfläche der CD startbar.

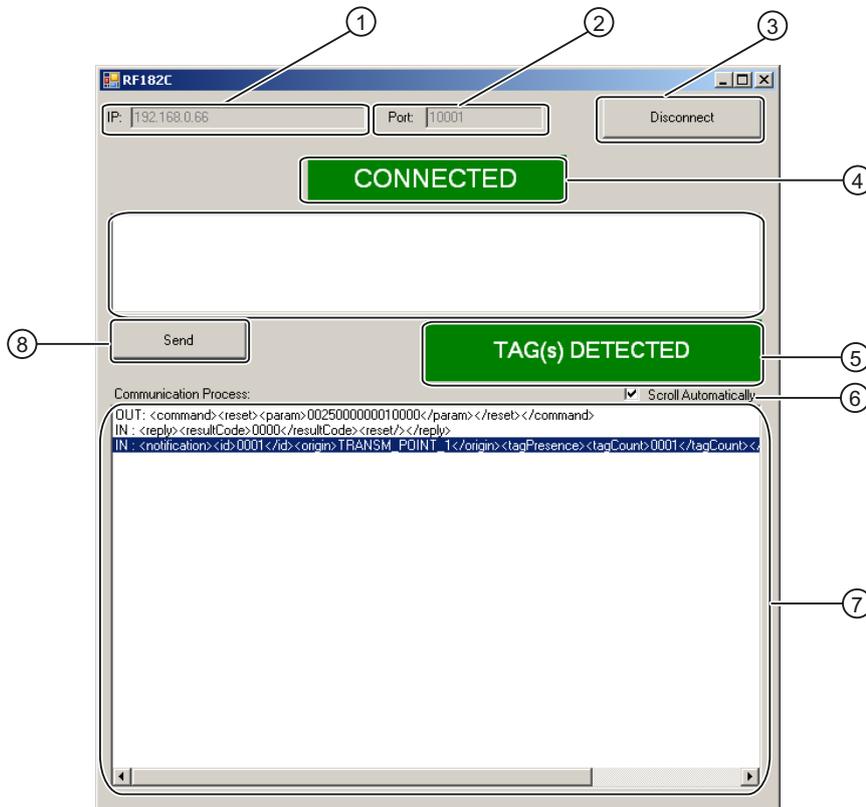
Die Sourcen der Applikation sind in einem Zip-File auf der CD hinterlegt. Sie finden die Datei unter "Daten/Tools > Applications/RF182C".

### Funktionen

Mit dieser Applikation stehen Ihnen folgende Funktionen zur Verfügung:

- Verbindung aufbauen
- Verbindung abbauen
- Hinweise zur Verbindung
- Eingabefenster zum Eingeben der Befehle
- Übertragen der Befehle zum Kommunikationsmodul RF182C
- Anzeigefenster zum Mitverfolgen des Kommunikationsprozesses der RF182C (Quittungen und Fehlermeldungen)

### 10.2.1 Aufbau Bedienoberfläche



- ① Eingabefenster IP-Adresse
- ② Eingabefenster Portnummer
- ③ Schaltfläche zum Aufbau der Verbindung bzw. Abbau der Verbindung
- ④ Hinweis: Verbunden bzw. nicht verbunden
- ⑤ Hinweis: Anwesenheit oder keine Anwesenheit
- ⑥ Optionskästchen: Ausgabefenster scrollt automatisch bzw. scrollt nicht automatisch
- ⑦ Ausgabefenster Empfangsdaten
- ⑧ Schaltfläche zum Verschicken der Daten im Eingabefenster

Bild 10-2 Benutzeroberfläche

## 10.2.2 Auszüge Beispielcode der Anwenderapplikation in C#

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using System.Net;
using System.Net.Sockets;
using System.Web;

namespace RF182CApp

    public partial class MainForm : Form
    {
        // Socket to realise TCP/IP connction
        private Socket Connection = null;
        // connectet (true) or not (false)
        private bool ConnectionState = false;
        // buffer for received data
        private String ReceiveBuffer = "";
        // Indicator for an error in asynchronous receive threads
        bool AsyncError = false;

        public MainForm()
        {
            InitializeComponent();
        }

        /* This routine is called when the window in initialized. */
        private void MainForm_Load(object sender, EventArgs e)
        {
            //We're not connected at the beginning
            SetConnectionState(false);
        }

        /* This routine is called when the Connect / Disconnect button
        * is clicked.
        * */
        private void bConnect_Click(object sender, EventArgs e)
        {
            if (ConnectionState == true) //already connected --> close connection
            {
                /* Connection may be dsiposed ahead (i.e. if an error
                * occured before.
                * */
                if (Connection != null)
                {
                    //Close the socket
                    Connection.Close();
                }
            }
        }
    }
}
```

```
        }
        SetConnectionState(false);
    }
    else //not connected --> connect
    {
        //Try to connect the socket as client.
        if (Connect() == true)
        {
            // Connection was established sucessfully
            SetConnectionState(true);
        }
    }
}

/* This function en- / disables controls needed to set up
 * the connection and those needed to communicate separately from
 * each other.
 * */
private void Enable(bool enableConnectionData, bool enableCommunicationData)
{
    editIP.Enabled = enableConnectionData;
    editPort.Enabled = enableConnectionData;

    editMessage.Enabled = enableCommunicationData;
    buttonSend.Enabled = enableCommunicationData;
}

/* Sets the connection state to true (connected) or false
 * (not connected) with all necessary consequences (starting/stopping
 * timers, changing colours, texts and the serviceability of the
 * controls.
 * */
private void SetConnectionState(bool state)
{
    if (state == true)
    {
        ConnectionState = true;
        // Visualisation
        Enable(false, true);
        labelConnectionState.Text = "CONNECTED";
        labelConnectionState.BackColor = Color.Green;
        buttonConnect.Text = "Disconnect";
        // Start UpdateTimer
        UpdateTimer.Start();
    }
    else
    {
        ConnectionState = false;
        //Stop UpdateTimer
        UpdateTimer.Stop();
        //Now, we don't know anything about tags in the field
    }
}
```

```
        SetTagDetectionState(TagDetectionState.UNDEFINED);
        //There may be uncollected data in the buffer --> call ParseBuffer()
        ParseBuffer();
        // Visualasiation
        Enable(true, false);
        labelConnectionState.Text = "NOT CONNECTED";
        labelConnectionState.BackColor = Color.Red;
        buttonConnect.Text = "Connect";
        AsyncError = false;
    }
}

/* Connects a stream based TCP/IP socket as client. */
private bool Connect()
{
    try
    {
        //Collect port and IP from the window.
        int port = 0;
        if (Int32.TryParse(editPort.Text, out port) == false || port <= 0)
        {
            MessageBox.Show(editIP.Text + " is not a legal port");
            return false;
        }

        IPHostEntry hostEntry = null;

        // Get host related information.
        hostEntry = Dns.GetHostEntry(editIP.Text);

        // Loop through the AddressList to obtain the supported AddressFamily. This is to avoid
        // an exception that occurs when the host IP Address is not compatible with the address
        family
        // (typical in the IPv6 case).
        foreach (IPAddress address in hostEntry.AddressList)
        {
            IPEndPoint ipe = new IPEndPoint(address, port);
            // stream based TCP/IP socket
            Socket tempSocket = new Socket(ipe.AddressFamily, SocketType.Stream, ProtocolType.Tcp);

            // the actual connect
            tempSocket.Connect(ipe);

            if (tempSocket.Connected)
            {
                //a connection was established sucessfully
                Connection = tempSocket;
                Connection.ReceiveTimeout = 25;
                break;
            }
        }
        if (Connection == null) return false;
    }
}
```

```

        //Start asynchronous receive
        ReceiveString s = new ReceiveString();
        Connection.BeginReceive(s.buffer, 0, ReceiveString.BufferSize, 0, new
        AsyncCallback(ReceiveCallback), s);
        return true;

    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message, "Connecting Failed");
        return false;
    }
}

/* This routine is called if there is new data in the asynchronous
 * receive available. The data is stored in ReceiveBuffer, so that
 * it can be collected by the synchronous timer UpdateTimer.
 * */
private void ReceiveCallback(IAsyncResult res)
{
    try
    {
        if (ConnectionState == true)
        {
            // Collect data
            int size = Connection.EndReceive(res);
            ReceiveString s = (ReceiveString)res.AsyncState;
            // Without a lock we might cause race situations
            lock (ReceiveBuffer)
            {
                //Store data in buffer
                ReceiveBuffer += Encoding.ASCII.GetString(s.buffer, 0, size);
            }

            //Start new asynchronous receive
            ReceiveString rs = new ReceiveString();
            Connection.BeginReceive(rs.buffer, 0, ReceiveString.BufferSize, 0, new
            AsyncCallback(ReceiveCallback), rs);
        }
    }
    catch (Exception ex)
    {
        // An error occurred --> report it
        AsyncError = true;
        MessageBox.Show(ex.Message);
    }
}

//This routine is called when the Send button is clicked
private void buttonSend_Click(object sender, EventArgs e)
{

```

```
//only do something if there is a message specified
if (editMessage.Text != "")
{
    // convert to byte array
    byte[] buffer = Encoding.ASCII.GetBytes(editMessage.Text.ToCharArray());
    try
    {
        // Send!
        int count = Connection.Send(buffer);
        // Did we send everything?
        if (count != buffer.Length)
        {
            MessageBox.Show("Sending failed!");
            SetConnectionState(false);
        }
        //Show the message in the list.
        AppendOutMessage(editMessage.Text);
    }
    catch (Exception ex)
    {
        SetConnectionState(false);
        //An error occurred
        MessageBox.Show(ex.Message);
    }
    //Empty the editbox
    editMessage.Text = "";
}

}

/* This routine is called when UpdateTimer ticks. This happens
 * synchronous to the thread owning the dialog, hence we can
 * manipulate the list.
 * */
private void UpdateTimer_Tick(object sender, EventArgs e)
{
    if (AsyncError == true)
    {
        //An error occurred!
        SetConnectionState(false);
    }
    else
    {
        ParseBuffer();
    }
}

/* Parses ReceiveBuffer for complete XML telegrams and appends
 * them to the list.
 * */
private void ParseBuffer()
{
    // Without a lock we might cause race situations
```

```

// (ReceiveCallback is asynchronous)
lock (ReceiveBuffer)
{
    for (XMLTag tag = FirstTag(); tag != null; tag = FirstTag())
    {
        // Extract the parsed message and append it
        String message = ReceiveBuffer.Substring(tag.startIndex, tag.length);
        AppendInMessage(message);
        // Remove parsed message from ReceiveBuffer
        ReceiveBuffer = ReceiveBuffer.Substring(tag.startIndex + tag.length);
        // See if we got information about detected tags
        if (message.Contains("<tagCount>"))
        {
            if (message.Contains("<tagCount>0000</tagCount>"))
            {
                //There are no tags in the field
                SetTagDetectionState(TagDetectionState.NO);
            }
            else
            {
                // There are tag in the field
                SetTagDetectionState(TagDetectionState.YES);
            }
        }
    }
}

// Looks for the first complete XML telegram in ReceiveBuffer.
private XMLTag FirstTag()
{
    // Is there a reply, a notification or an alarm
    int index1 = ReceiveBuffer.IndexOf("<reply>");
    int index2 = ReceiveBuffer.IndexOf("<notification>");
    int index3 = ReceiveBuffer.IndexOf("<alarm>");

    if( index1==-1 && index2==-1 &&index3==-1) return null; //No XML tag found

    if (index1 == -1) index1 = Int32.MaxValue;
    if (index2 == -1) index2 = Int32.MaxValue;
    if (index3 == -1) index3 = Int32.MaxValue;

    //Assume that the first tag is an alarm
    String endTag="</alarm>";
    XMLTag tag = new XMLTag();
    tag.type="alarm";
    tag.startIndex = index3;

    //See if this is true and change it if necessary
    if (index1 < index2 && index1 < index3)
    {

```

```
        // first tag is a reply
        endTag = "</reply>";
        tag.type = "reply";
        tag.startIndex = index1;
    }
    else if (index2 < index3)
    {
        // first tag is a notification
        endTag = "</notification>";
        tag.type = "notification";
        tag.startIndex = index2;
    }

    //Is the complete message in the buffer?
    int endIndex = ReceiveBuffer.IndexOf(endTag, tag.startIndex);
    if (endIndex == -1) return null;

    tag.length = endIndex - tag.startIndex + endTag.Length;

    return tag;
}

// Appends an outgoing message to the list
private void AppendOutMessage(String message)
{
    //Remove all CR and LF
    message.Replace("\r", "");
    message.Replace("\n", "");
    //Add message
    int index = listProcess.Items.Add("OUT: " + message);
    //Scroll if desired
    if (checkScroll.Checked == true)
    {
        listProcess.SelectedIndex = index;
    }
}

// Appends an ingoing message to the list
private void AppendInMessage(String message)
{
    //funcionality just lie AppendOutMessage
    message.Replace("\r", "");
    message.Replace("\n", "");
    int index = listProcess.Items.Add("IN : " + message);
    if (checkScroll.Checked == true)
    {
        listProcess.SelectedIndex = index;
    }
}

/* Call this function if a tag appeared / disappeared. */
private void SetTagDetectionState(TagDetectionState state)
```

```

    {
        switch (state)
        {
            case TagDetectionState.YES:
                labelTagDetected.Text = "TAG(s) DETECTED";
                labelTagDetected.BackColor = Color.Green;
                break;
            case TagDetectionState.NO:
                labelTagDetected.Text = "NO TAG DETECTED";
                labelTagDetected.BackColor = Color.Red;
                break;
            case TagDetectionState.UNDEFINED:
                labelTagDetected.Text = "";
                labelTagDetected.BackColor = BackColor; //Color of the dialog
                break;
        }
    }
}

/* As long as no tag presence notification was send, we have no
 * information whether there is a tag in the field or not.
 * --> three states
 * */
enum TagDetectionState
{
    YES, NO, UNDEFINED
}

// This class is for handling asynchronous communication processes
internal class ReceiveString
{
    public readonly static int BufferSize = 512;
    public byte[] buffer = new byte[BufferSize];
}

// This class describes a XML block in ReceiveBuffer
internal class XMLTag
{
    //Index of first character
    public int startIndex;
    //Block length
    public int length;
    // Type ( reply, notification or alarm, since only telegrams
    // from the ASM are handeld within this structure)
    public String type;
}
}

```

### 10.2.3 Funktionen der Applikation RF182C

Die folgenden Screenshots zeigen die verschiedenen Funktionen der Applikation RF182C:

#### Beispielhaftes Arbeiten mit der Applikation

1. Geben Sie zuerst IP-Adresse und Portnummer in die entsprechenden Eingabefelder ein.

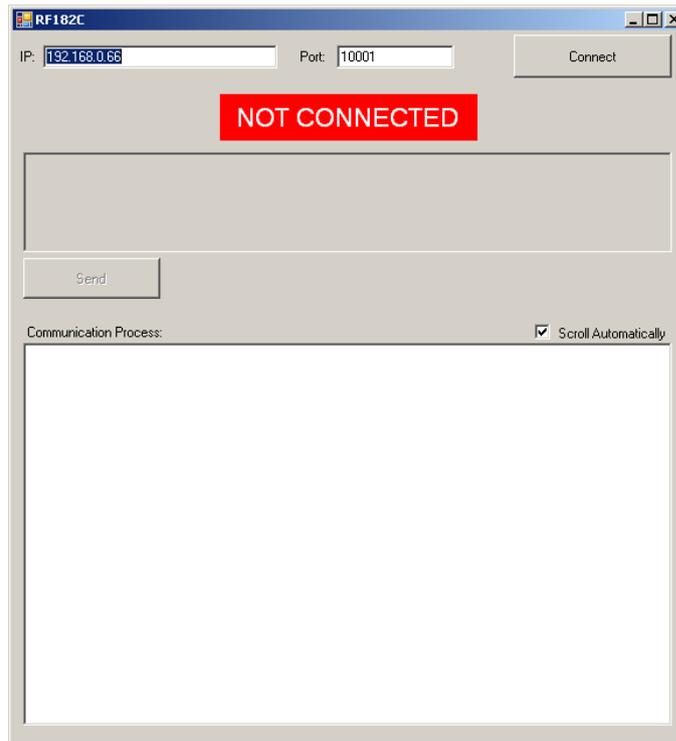


Bild 10-3 RF182C nicht verbunden

2. Klicken Sie anschließend auf die Schaltfläche "Connect".

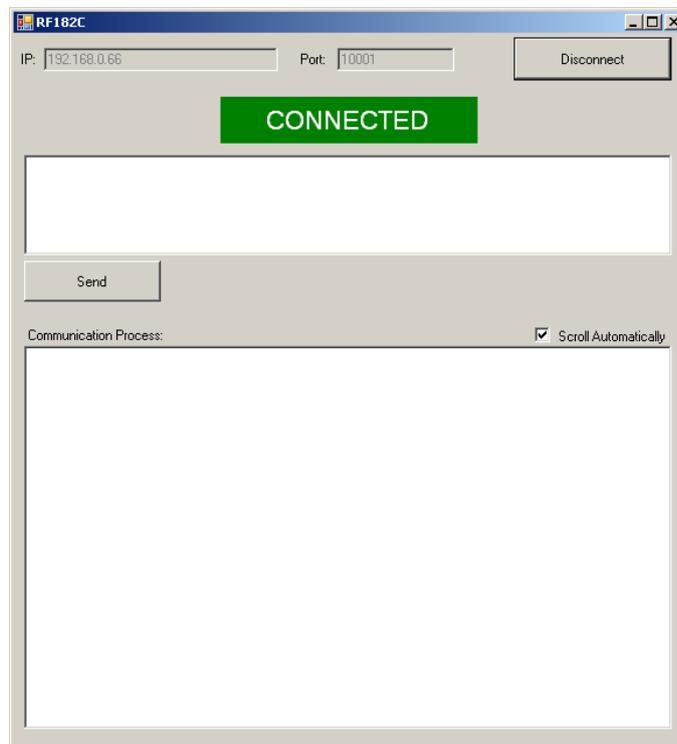


Bild 10-4 Verbindung erfolgreich

Der Verbindungsaufbau mit der RF182C war erfolgreich. Das Sendefenster leert sich nach erfolgreichem Senden.

3. Geben Sie einen Reset-Befehl ins Eingabefenster ein. Klicken Sie auf die Schaltfläche "Send".



Bild 10-5 Reset-Befehl

Das Kommunikationsmodul RF182C arbeitet im Defaultmodus.

- Das nächste Fenster zeigt, dass der Reset-Befehl erfolgreich verschickt wurde. Es wurde von der RF182C eine Quittung mit dem Errorcode "0000" (Alles okay) gesendet.

Es wird angezeigt, dass ein Tag im Antennenfeld des Readers erkannt wurde (0001).

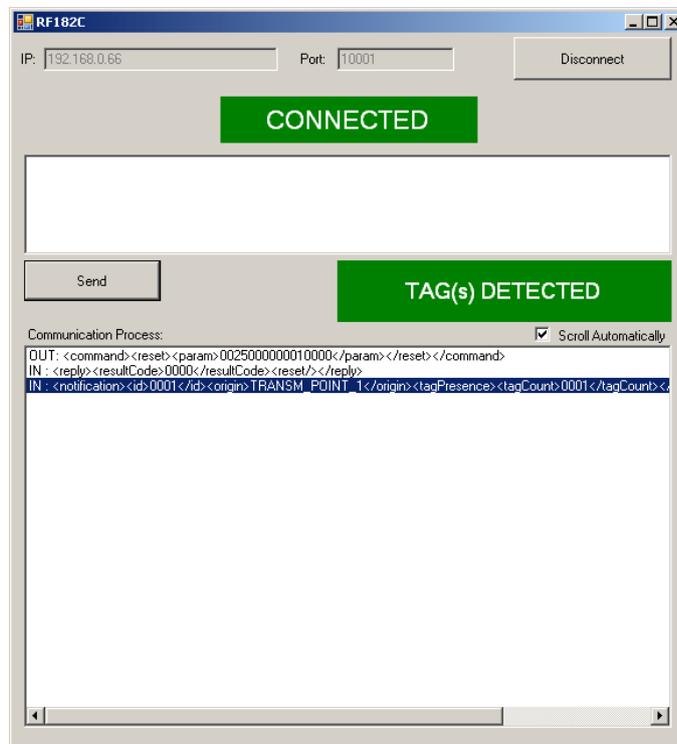


Bild 10-6 Tag erkannt

5. Das nächste Fenster zeigt den Fall, dass der Tag das Antennenfeld des Readers wieder verlassen hat (0000).

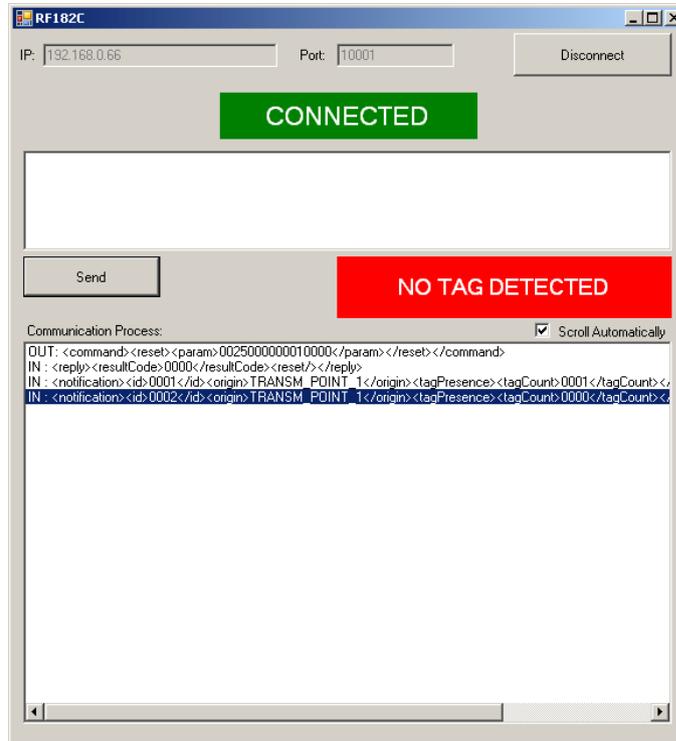


Bild 10-7 Tag aus dem Feld

## 10.3 Beispielapplikation für PLC nach DIN IEC 61131

In Vorbereitung.





<b>Umgebungstemperatur</b>	
im Betrieb	0 bis +60 °C
bei Transport und Lagerung	-40 bis +70 °C
<b>Maße (B x H x T) in mm</b>	
nur Basisgerät	60 x 210 x 30
Basisgerät mit M12, 7/8"-Anschlussblock	60 x 210 x 54
Basisgerät mit Push-Pull-Anschlussblock	60 x 216 x 100
<b>Gewicht</b>	
Basisgerät	ca. 210 g
Anschlussblock M12, 7/8"	ca. 230 g
Anschlussblock Push Pull	ca. 120 g
<b>Mechanische Umgebungsbedingungen</b>	
Einbaulage	Alle Einbaulagen möglich
Schwingen, Vibration im Betrieb	Nach IEC 61131-2: 0,75 mm (10Hz bis 58 Hz) 10 g (58 Hz bis 150 Hz)
Stoßfestigkeit, Schock im Betrieb	Nach IEC 61131-2: 30 g
<b>Schutzart</b>	IP67
<b>MTBF (Mean Time Between Failures) in Jahren</b>	
Basisgerät	121
Anschlussblock	1100
<b>Zulassungen</b>	cULus (File E116536) FCC Code of Federal Regulations, CFR 47, Part 15, Sections 15.107 and 15.109 (Class A)

- 1) Bei allen Versorgungs- und Signalspannungen muss es sich um Schutzkleinspannung handeln (SELV / PELV nach EN 60950)  
DC 24 V-Versorgung: sichere (elektrische) Trennung der Kleinspannung (SELV / PELV nach EN 60950)
- 2) Die Stromversorgung muss den erforderlichen Strom von max. 500 mA für kurzzeitige Spannungsausfälle  $\leq 20$  ms liefern.
- 3) **Bei einer Stromstärke > 8 A ist ein Leiterquerschnitt von 2,5 mm<sup>2</sup> zwingend vorgeschrieben.**

## Maßbilder

### 12.1 Maßbild RF182C mit Befestigungslöchern

Maßbild eines RF182C mit Busanschlussblock M12, 7/8" PN

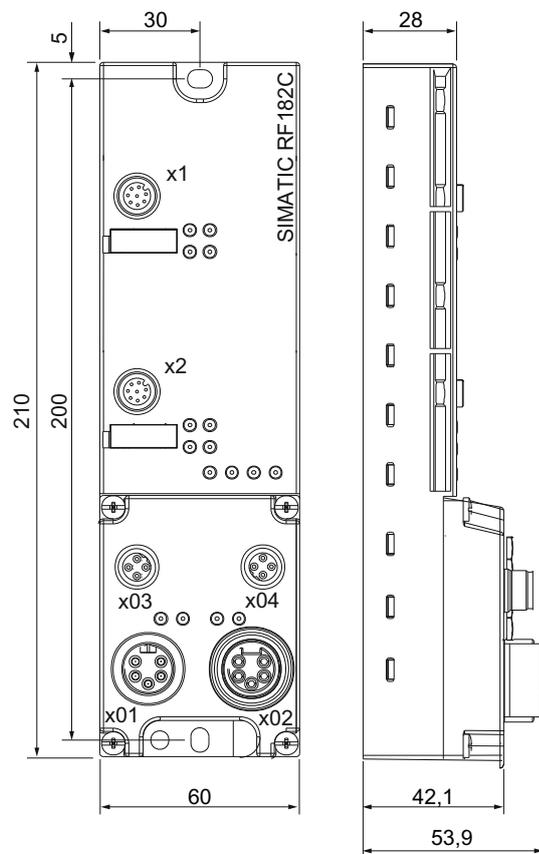


Bild 12-1 Maßbild eines RF182C mit Busanschlussblock M12, 7/8" PN

Maßbild eines RF182C mit Busanschlussblock Push Pull

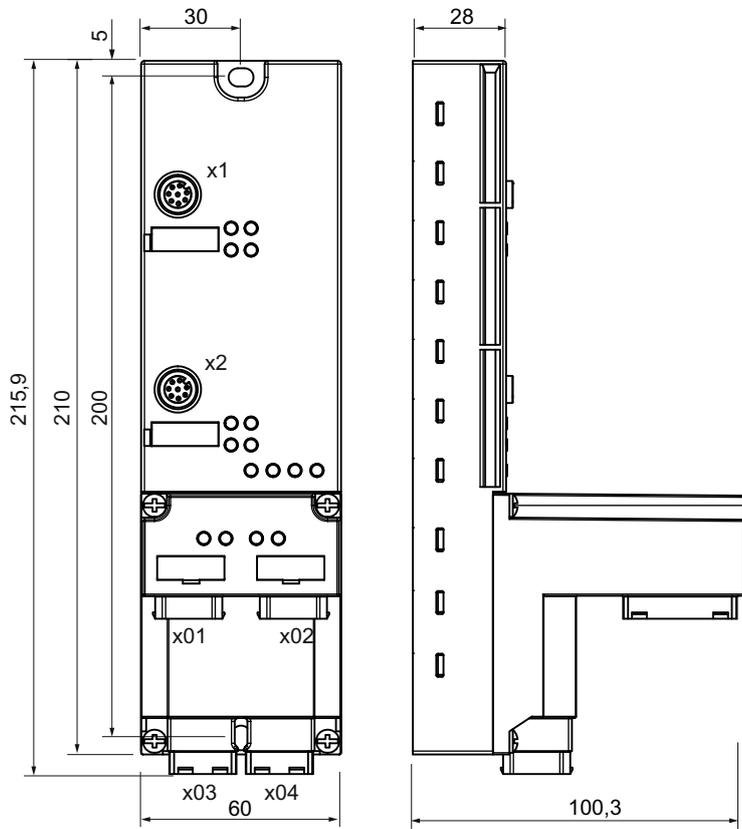


Bild 12-2 Maßbild eines RF182C mit Busanschlussblock Push Pull

## Anschlusskabel zum Reader / SLG

### 13.1 Belegung der Standardkabel

#### Verfügbare Kabel



Bild 13-1 Anschlusskabel M12 → Reader / SLG; l = 2 m, 5 m (MOBY U)

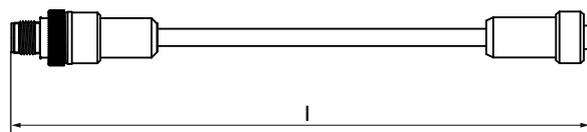


Bild 13-2 Anschlusskabel / Verlängerungskabel M12 ↔ M12; l = 2 m, 5 m, 10 m, 20 m, 50 m

- Anschlusskabel RF300/RF600; MOBY D nur 6GT2602-0AB10-0AX0
- Verlängerungskabel für alle RFID-Systeme

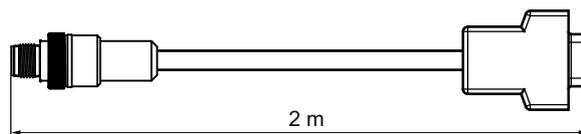


Bild 13-3 Anschlusskabel M12 → Sub-D (MOBY D alle SLG außer 6GT2602-0AB10-0AX0)

#### Maximale Kabellänge

Das RF182C kann mit jeder Reader / SLG-Konfiguration mit einer maximalen Kabellänge von 50 m betrieben werden.

Längere Anschlusskabel bis 1000 m sind teilweise möglich. Hierzu muss jedoch der Stromverbrauch der angeschlossenen Reader / SLG berücksichtigt werden. Hinweise finden Sie in den jeweiligen Systemhandbüchern.

Das Hintereinanderschalten von mehr als zwei Teilstücken zu einem langen Kabelstück sollte wegen der zusätzlichen Kontaktübergangswiderstände vermieden werden.

### Steckerbelegung

Tabelle 13- 1 Anschlusskabel M12 ↔ Reader / SLG

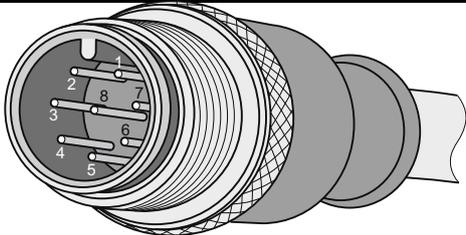
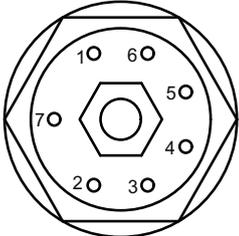
M12-Stecker (Stift)			Reader / SLG-Stecker (Buchse)
	1	2	
	2	5	
	3	3	
	4	4	
	5	6	
	6	1	
	7	–	
	8	7	

Tabelle 13- 2 Anschlusskabel / Verlängerungskabel M12 ↔ M12

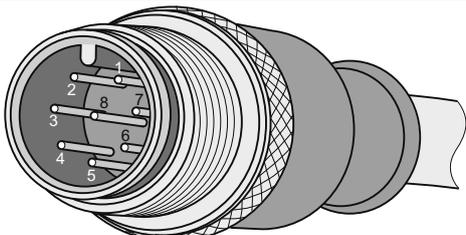
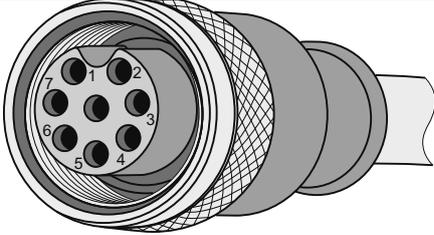
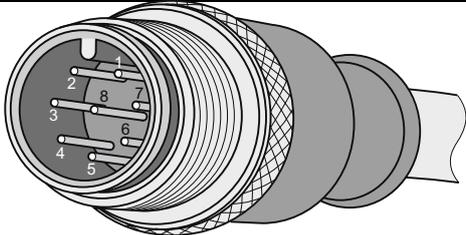
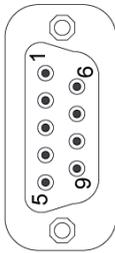
M12-Stecker (Stift)			M12-Stecker (Stift)
	1	1	
	2	2	
	3	3	
	4	4	
	5	5	
	6	6	
	7	7	
	8	8	

Tabelle 13- 3 Anschlusskabel M12 ↔ Sub-D 9-pol.

M12-Stecker (Stift)			Sub-D-Stecker (Buchse)
	1	–	
	2	5	
	3	7	
	4	3	
	5	2	
	6	6	
	7	–	
	8	1, 8	

Beachte:  
 Reader / SLG mit Sub-D-Stecker müssen über einen zusätzlichen Stecker mit DC 24 V versorgt werden.

## 13.2 Selbst konfektionierte Kabel

Für Anwender, die ihr Kabel individuell selbst konfektionieren wollen, steht ein Reader / SLG-Anschlussstecker mit Schraubklemmen zur Verfügung (siehe jeweiliges Systemhandbuch). Kabel und Reader / SLG-Anschlussstecker können nach dem Katalog *FS 10 Sensoren für die Fertigungsautomatisierung* bestellt werden.

### Kabelaufbau

Für selbst konfektionierte Kabel benötigen Sie Kabel mit folgenden Spezifikationen:

7 x 0,25 mm<sup>2</sup>  
 LiYC11Y 7 x 0,25

### Anschlussstecker

M12-Stecker können im einschlägigen Fachhandel (z. B. Fa. Binder) bezogen werden.

### Pinbelegung

Die Pin-Belegung ist in der folgenden Tabelle aufgelistet.

Tabelle 13- 4 Pin-Belegung

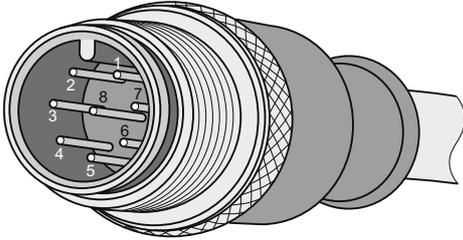
M12-Stecker (Stift)	Pin	Signal	Adernfarbe
	1	1L+ (+ 24 V)	Datenblatt des Kabelherstellers beachten
	2	-RxD	
	3	0 V	
	4	RxD	
	5	TxD	
	6	-TxD	
	7	Frei	
	8	Funktionserde (PE) / Schirm	



Tabelle 14- 1 Bestelldaten RF182C und Zubehör

<b>RF182C</b>	
Kommunikationsmodul RF182C max. 2 SLG bzw. Reader anschließbar	6GT2002-0JD10
Anschlussblock M12, 7/8" (5-polig)	6GT2002-1JD00
Anschlussblock M12, 7/8" (4-polig)	6GT2002-4JD00
Anschlussblock Push Pull, RJ45	6GT2002-2JD00
Beschriftungsschilder 20 x 7 mm (1 Packung = 340 Stück)	3RT1900-1SB20
<b>Zubehör für Anschlussblock M12 7/8" (5-polig)</b>	
IE Steckleitung für PROFINET/Ethernet (vorkonfektionierte schleppfähige Leitung mit zwei M12-Steckern 4-polig, Kodierung D)	6XV1870-8Axxx <sup>1)</sup>
7/8"-Steckleitung für Versorgungsspannung (5 x 1,5 mm <sup>2</sup> ) (vorkonfektionierte schleppfähige Energieleitung mit zwei 5-poligen 7/8"-Steckern)	6XV1822-5Bxxx <sup>1)</sup>
Schleppfähige Energieleitung (5 x 1,5 mm <sup>2</sup> ) (unkonfektionierte; Länge min. 20 m, Länge max. 1000 m)	6XV1830-8AH10
Anschlussstecker 7/8" für Versorgungsspannung; (1 Packung = 5 Stück) <ul style="list-style-type: none"> <li>• mit Stifteinsatz</li> <li>• mit Buchseneinsatz</li> </ul>	6GK1905-0FA00 6GK1905-0FB00
RJ45-Steckverbinder mit Metallgehäuse und FC Anschlussstechnik, 180 ° Kabelabgang;(1 Packung = 1 Stück)	6GK1901-1BB10-2AA0
Schaltschrankdurchführung zum Überbrücken von M12 (D-Kodiert) Anschlussstechnik (IP65) auf RJ45 Anschlussstechnik (IP20) ;(1 Packung = 5 Stück)	6GK1901-0DM20-2AA5
M12 Steckverbinder mit Metallgehäuse und Schnellanschlussstechnik, 180 ° Kabelabgang (D-Kodiert) ; (1 Packung = 1 Stück)	6GK1901-0DB10-6AA0
Verschlusskappen M12	3RX9802-0AA00
Verschlusskappen 7/8" (1 Packung = 10 Stück)	6ES7194-3JA00-0AA0
PROFINET-/Ethernet-Standard-Kabel 2x2, Typ A, unkonfektionierte; Mindestbestellmenge 20 m	6XV1840-2AH10
<b>Zubehör für Anschlussblock M12 7/8" (4-polig)</b>	
Kabel für Versorgungsspannung mit 7/8" 4-poligen Steckern konfektionierte (nur für 6GT2002-4JD00)	Nicht bei Siemens bestellbar

<b>Zubehör für Anschlussblock Push Pull</b>	
Schleppfähige Energieleitung (5 x 1,5 mm <sup>2</sup> ) (unkonfektioniert; Länge min. 20 m, Länge max. 1000 m)	6XV1830-8AH10
Push-Pull-Anschlussstecker für 1L+/ 2L+, unkonfektioniert	6GK1907-0AB10-6AA0
Push-Pull-Anschlussstecker für RJ45, unkonfektioniert	6GK1901-1BB10-6AA0
Abdeckkappen für Push-Pull-Buchsen (1L+/ 2L+), 5 Stück je Packung, 1 Stück	6ES7194-4JA50-0AA0
Abdeckkappen für Push-Pull-Buchsen RJ45, 5 Stück je Packung, 1 Stück	6ES7194-4JD50-0AA0
PROFINET-/Ethernet-Standard-Kabel 2x2, Typ A, unkonfektioniert; Mindestbestellmenge 20 m	6XV1840-2AH10
<b>Zubehör RFID</b>	
SLG-Kabel MOBY U; 2 m	6GT2091-0FH20
SLG-Kabel MOBY U; 5 m	6GT2091-0FH50
SLG-Kabel MOBY D; 2 m	6GT2691-0FH20
Reader-Kabel RF300, RF600 Verlängerungskabel RF300 / RF600/ MOBY E / U / D; 2 m	6GT2891-0FH20
Reader-Kabel RF300, RF600 Verlängerungskabel RF300 / RF600 / MOBY U / D; 5 m	6GT2891-0FH50
Reader-Kabel RF300, RF600 Verlängerungskabel RF300 / RF600 / MOBY U / D; 10 m	6GT2891-0FN10
Reader-Kabel RF300, RF600 Verlängerungskabel RF300 / RF600 / MOBY U / D; 20 m	6GT2891-0FN20
Reader-Kabel RF300, RF600 Verlängerungskabel RF300 / RF600 / MOBY U / D; 50 m	6GT2891-0FN50
Reader-Kabel für RF300; Stecker am Reader abgewinkelt; 2m	6GT2891-0JH20
RFID-CD "Software&Documentation"	6GT2080-2AA10

1) Diese Kabel stehen in verschiedenen Längensvarianten zur Verfügung. Siehe hierzu Katalog IK PI.

## Befehls- und Quittungstelegramme

In diesem Kapitel finden Sie Detailinformationen zu einigen Befehlen in Kapitel Kommunikationsschnittstelle (Seite 53). Es sind hier nur diejenigen Befehle aufgeführt, die eine spezielle Kodierung von Befehlen und Ergebnissen benötigen.

---

### Hinweis

Spezielle Informationen zu Telegrammergänzungen bei den Readern RF620R/RF630R finden Sie im Anhang des "Parametrierhandbuchs RF620R/RF630R".

---

# RESET

Befehlstelegramm  
an MOBY-ASM

standby	Param	00	dili	multitag	fcon	ftim
---------	-------	----	------	----------	------	------

Max. Anzahl d. parallel im Feld bearbeitbaren Tags; zugelassener Wert:  
00 01 hex

MOBY U:  
field\_ON\_control (siehe Input-Parameter)  
BERO-Betriebsart  
00 hex = ohne BEROs; keine SLG  
Synchronisation  
01 hex = field\_ON\_time schaltet das Feld aus  
02 hex = 1. BERO schaltet das Feld ein;  
2. BERO schaltet das Feld aus  
03 hex = SLG-Synchronisation über Kabelverbindung aktiviert (siehe Handbuch für Projektierung, Montage und Service für MOBY U)

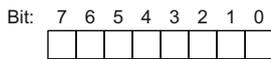
MOBY D, RF300 nicht verwendet (00 hex)

MOBY U:  
distance\_limiting\_Reichweitenbegrenzung (siehe Input-Parameter)  
05; 0A; 0F; 14; 19; 1E; 23 hex = 0,5; 1,0; 1,5; 2,0; 2,5; 3,0; 3,5 m  
85; 8A; 8F; 94; 99; 9E; A3 hex = dito mit reduzierter Sendeleistung

MOBY D: HF-Leistung (siehe Input-Parameter)  
nur für SLG D10S in 0,25 W-Schritten einstellbar  
02 hex = 0,5 W ... 10 hex = 4 W (default) ...28 hex = 10 W

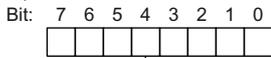
RF300:  
distance\_limiting nur bei RF380R Änderung der Ausgangsleistung  
02 hex = 0,5 W      06 hex = 1,5 W  
03 hex = 0,75 W    07 hex = 1,75 W  
04 hex = 1,0 W     08 hex = 2,0 W  
05 hex = 1,25 W

Option 1: nur MOBY RF300



0 = unbelegt

1 = ERR-LED am SLG rücksetzen  
0 = ERR-LED am SLG nicht rücksetzen



5 = MOBY U/D bzw. RF300 - ohne Multitag  
res.

Anwesenheitskontrolle und MDS-Steuerung (MDS\_control\_)  
000 = keine ANW-Kontrolle  
001 = keine MDS-Steuerung; ANW-Kontrolle über Firmware (Default)

MOBY U:

scanning\_time\_; Standby-Zeit für den MDS (siehe Input-Parameter)  
00 hex = kein Standby-Betrieb  
01 hex ... C8 hex = 7 ms ... 1400 ms Standby-Zeit

MOBY D, RF300: nicht verwendet (00 hex)

MOBY U:  
field\_ON\_time\_ (siehe Input-Parameter)  
00 hex = ohne BEROs  
01 hex ... FF hex = 1 ... 255 s Einschaltzeit für das SLG-Feld

MOBY D:  
MDS-Typ (siehe Input-Parameter)  
00 hex = I-Code 1 (z. B. MDS D139)  
01 hex = ISO-MDS  
02 hex = (Dual-Treiber, ICode1 und ISO)  
03 hex = (MDS D324-Optimierung, nur bei SLG D10S)  
04 hex = (MDS D4xx-Optimierung, nur bei SLG D11S/12S)

RF300:  
field\_ON\_time (siehe Input-Parameter)  
00 hex = RF300 Mode (kein ISO)  
01 hex = herstellerunabhängiger Tag  
03 hex = ISO my-d (Infineon SRF 55V10P)  
04 hex = ISO (Fujitsu MB89R118)  
05 hex = ISO I-Code SLI (NXP SL2 ICS20)  
06 hex = ISO Tag-it HFI (Texas Instruments)  
07 hex = ISO (ST LR12K)

### GetReaderStatus mode = 01

Ergebnistelegamm  
mode = 01

01	SLG-Status
----	------------

die Bedeutung des SLG-Status  
ist in folgender Tabelle  
beschrieben

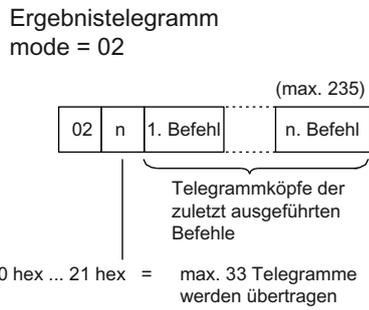
Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Param	S-Info	HW	HW-V		LD-V		FW		FW-V		TR		TR-V		INT	baud	res	res

Byte	18	19	20	21	22	23	24	25	26	27	
Param	res	dili	multi	fcon	fon	sync	ant	stand_ by	MDS control		

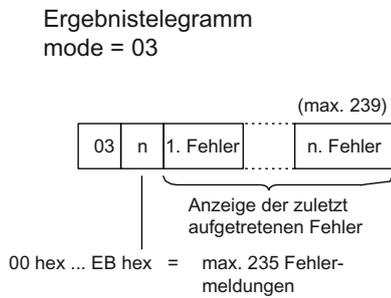
Parametername	Kommentar
S-Info	Reader status mode information = 01
HW	Hardware-Typ
HW-V	Hardware-Version
LD-V	Urlader-Version
FW	Firmware-Typ
FW-V	Firmware-Version
TR	Treiber-Typ
TR-V	Treiber-Version
INT	Schnittstelle (RS 232/RS 422)
Baud	Baudrate
Dili	Reichweiten-Leistungseinstellung
multi	Multitag Reader
fcon	field_ON_control: BERO-Betriebsart (RF300: res)
fon	Field_on_time: <ul style="list-style-type: none"> <li>• MOBY U: BERO-Zeit</li> <li>• MOBY D: Tag-Typ</li> <li>• RF300-ISO: Tag-Typ</li> </ul>
sync	Semaphorensteuerung (Synchronisation mit Reader (RF300: res))
ant	Status Antenne
stand_by	Standby-Zeit, nachdem ein Befehl ausgeführt wurde (RF300: res)
MDS control	Anwesenheitsmodus

Nähere Informationen können Sie der jeweiligen Reader-Beschreibung entnehmen.

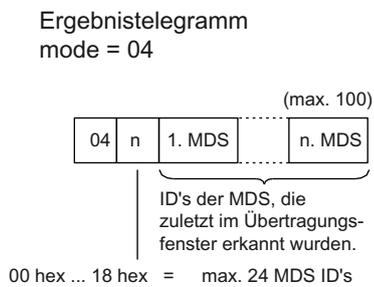
### GetReaderStatus mode = 02



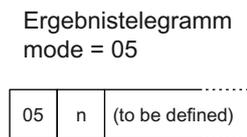
### GetReaderStatus mode = 03



### GetReaderStatus mode = 04



### GetReaderStatus mode = 05



### GetReaderStatus mode = 06

Ergebnistelegamm  
mode = 06

06	Diagnosedaten
----	---------------

Die Bedeutung der  
Diagnosedaten ist in  
folgender Tabelle  
beschrieben

<b>Byte</b>	0	1	2	3	4	5	6	7	8 ... 26
<b>Param</b>	S-Info	FZP	ABZ	CFZ	SFZ	CRCFZ	BSTAT	ASMfZ	res.

Parametername	Kommentar
S-Info	Reader status mode information = 06
FZP	Fehlerzähler, passiv (Ruhefehlerzähler)
ABZ	Abbruchzähler
CFZ	Codefehlerzähler
SFZ	Signaturfehlerzähler
CRCFZ	CRC-Fehlerzähler
BSTAT	Aktueller Befehlsstatus
ASMfZ	Schnittstellenfehlerzähler zu ASM
res.	Reserve

**GetReaderStatus mode = 07**

Ergebnistelegamm  
mode = 07

07	SLG-Status
----	------------

die Bedeutung des SLG-Status  
ist in folgender Tabelle  
beschrieben

<b>Byte</b>	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
<b>Param</b>	S-Info	HW	HW-V		res0		FW		FW-V		TR		H	Min	Sec	res1	SLG-V	Baud

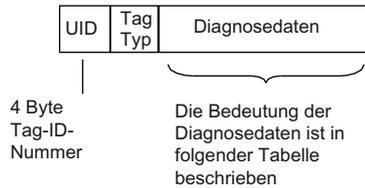
<b>Byte</b>	18	19	20	21	22	23	24	25	26	
<b>Param</b>	res2	dili_SLG	multi	field_on_control	field_on_time	expert	ant	scanning_time	MDS control	

Parametername	Kommentar
S-Info	Reader status mode information = 07
HW	Hardware-Typ
HW-V	Hardware-Version
res0	Reserviert
FW	Firmware-Typ
FW-V	Firmware-Version
TR	Treiber-Typ
H	Stunden
Min	Minuten
Sec	Sekunden
res1	Reserviert
SLG-V	Reader-Version
Baud	Baudrate
res2	Reserviert
dili_SLG	Eingestellte Sendeleistung
multi	Multitag Reader
field_on_control	Eingestellter Kommunikationstyp
field_on_time	Eingestellter Kanal
expert	Experten-Modus
ant	Status Antenne
scanning_time	Radio Kommunikationsprofil
MDS control	Anwesenheitsmodus

Nähere Informationen können Sie der jeweiligen Reader-Beschreibung entnehmen.

**GetTagStatus mode = 00**

Ergebnistelegamm  
mode = 00



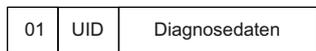
Byte	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
Param	UID				MDS_type	sum_sub					sum_search		week	year	battery		ST

Parametername	Kommentar
UID	UID (Tag-Nummer, EPC ID)
MDS_type	MDS-Typ
sum_sub	Summe der Teilrahmenzugriffe
sum_search	Summe der searchmode
week	Datum der letzten Änderung der Sleep_time (Kalenderwoche)
year	Datum der letzten Änderung der Sleep_time (Kalenderjahr)
battery	Restliche Batteriebensdauer in Prozent
ST	Eingestellter Sleep time Wert im MDS

Nähere Informationen können Sie der jeweiligen Reader-Beschreibung entnehmen.

### GetTagStatus mode = 01

Ergebnistelegamm  
mode = 01



8 Byte  
Tag-ID-  
Nummer  
(unique  
identifier)

Die Bedeutung der  
Diagnosedaten ist in  
folgender Tabelle  
beschrieben

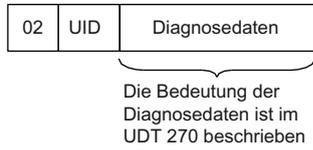
<b>Byte</b>	0	1	2	3	4	5	6	7	8	9	10 ... 15
<b>Param</b>	status	UID 1 4			UID 5 8				Lock_- state	res.	

Parametername	Kommentar
status	Tag status mode information = 01
UID 1 4	UID (Tag-Nummer, EPC ID)
UID 5 8	
Lock_state	Schreibschutz-Status EEPROM
res.	Reserve

Nähere Informationen können Sie der jeweiligen Reader-Beschreibung entnehmen.

### GetTagStatus mode = 02

Ergebnistelegamm  
mode = 02



Byte	0	1	2	3	4	5	6	7	8	9	10	11	12 ... 15
Param	status	UID 1 4			UID 5 8				LFD	FZP	FZA	res.	

Parametername	Kommentar
status	Tag status mode information = 02
UID 1 4	UID (Tag-Nummer, EPC ID)
UID 5 8	
LFD	Beziehung zwischen Grenzwert Leistungsflussdichte und aktuellem Messwert
FZP	Fehlerzähler, passiv (Ruhefeldzähler)
FZA	Fehlerzähler, aktiv (Fehler während der Kommunikation)
ANWZ	Anwesenheitsfehler
res.	Reserve

**GetTagStatus mode = 03**

Ergebnistelegamm  
mode = 03



Die Bedeutung der Diagnose-  
daten ist im Systemhandbuch  
RF300 beschrieben

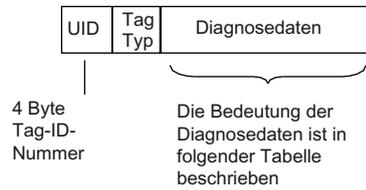
<b>Byte</b>	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
<b>Param</b>	status	UID 1 4			UID 5 8				IC_version	size		lock_state	block_size	nr_of_blocks	

Parametername	Kommentar
status	Tag status mode information = 02
UID 1 4	UID (Tag-Nummer)
UID 5 8	immer 0
IC_version	Chip Version (bei my-d = 00h)
size	Speichergröße in Byte
lock_state	Lockstatus, OTP-Information: pro Block (4x4 Bytes oder 2x8 Bytes) ein Bit (Bit=1: Block ist gesperrt)
block_size	Blockgröße des Transponders
nr_of_blocks	Blockanzahl

Nähere Informationen können Sie in der jeweiligen Reader Beschreibung entnehmen.

### GetTagStatus mode = 04

Ergebnistelegamm  
mode = 04



<b>Byte</b>	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
<b>Param</b>	status	UID 1 4			UID 5 8				ant	RSSI	H	Min	Sec	res	res1	res2	

Parametername	Kommentar
status	Tag status mode information = 04
UID 1 4	UID (Tag-Nummer, EPC ID)
UID 5 8	
ant	Antenne, die das MDS erkannt hat
RSSI	RSSI-Schwellwert
H	Stunden
Min	Minuten
Sec	Sekunden
res	Reserve
res1	Reserve
res2	Reserve

Nähere Informationen können Sie der jeweiligen Reader-Beschreibung entnehmen.



## Adressierung der RFID-Tags

### Adressraum der MDS-Varianten für MOBY U

Tabelle B- 1 Adressraum MOBY U

Speicher	Adressierung	Hexadezimalzahl 16 Bit	Integer-Zahl
<b>2 KByte Daten- speicher</b>	Anfangsadresse	0000	+0
	Endadresse	07FF	+2047
	SpeichergroÙe	08 00	2048
	<b>OTP-Speicher lesen</b> (Schreiben ist nur einmalig möglich. Der OTP-Speicher von MOBY U kann nur komplett bearbeitet werden. D. h. die Anfangsadresse muss immer mit dem Wert FFF0 hex und die Länge mit dem Wert 10 hex angegeben werden.)		
	Anfangsadresse	FFF0	-16
	Länge	10	+16
ID-Nr.: (festcodiert 4 Byte; das Auslesen ist nur mit dem Befehl MDS-Status möglich)			
<b>32 KByte Daten- speicher</b>	Anfangsadresse	0000	+0
	Endadresse	7FFF	+32767
	SpeichergroÙe	80 00	32768
	<b>OTP-Speicher lesen</b> (Schreiben ist nur einmalig möglich)*		
	Anfangsadresse	FFF0	-16
	Länge	10	+16
ID-Nr.: (festcodiert 4 Byte; das Auslesen ist nur mit dem Befehl MDS-Status möglich)			

### Adressraum der MDS-Varianten für MOBY D

Adressraum der RF300-Transponder siehe Systemhandbuch MOBY D (<http://support.automation.siemens.com/WW/view/de/13628689>), Kapitel "Mobile Datenspeicher > Einführung".

### Adressraum der Transponder-Varianten für RF300

Adressraum der RF300-Transponder siehe Systemhandbuch SIMATIC RF300 (<http://support.automation.siemens.com/WW/view/de/21738946>), Kapitel "RF300-Transponder > Speicheraufbau der RF300-Tags" und Kapitel "ISO-Transponder > Speicheraufbau der ISO-Tags".

### Adressraum der Transponder-Varianten für RF600

Adressraum der RF600-Transponder siehe Parametrierhandbuch SIMATIC RF620R/RF630R (<http://support.automation.siemens.com/WW/view/de/33287195>), Kapitel "Beispiele/Applikationen > Speicheraufbau".



# Schema der Übertragung der Hexadezimalen Tag-Daten über XML



Die Adressen für die Speicheradressierung auf dem Tag sind hexadezimal. Die Daten auf dem Tag sind ebenfalls hexadezimal hinterlegt. Bei den Befehlen Read/Write werden die Daten jedoch als ASCII-Zeichen übertragen. Eine Konvertierung Hex nach ASCII-Hex und umgekehrt geschieht automatisch im Kommunikationsmodul. Das folgende Beispiel zeigt das Schema der Kodierung:

## Befehl zum RF182C

```
<readTagData>
  <startAddress>0000</startAddress>
  <dataLength>0004</dataLength>
</readTagData>
```

## Daten im Tag

Adresse [hex]	Daten [hex]	Daten [ADC]
0000	4D	,M'
0001	4F	,O'
0002	42	,B'
0003	59	,Y'
...		

## Ergebnis des RF182C

```
<readTagData>
<returnValue>
  <data>4D4F4259</data>
</returnValue>
</readTagData>
```

**Optional: Hinterlegung der Ergebnisdaten im Datenbaustein einer Steuerung**

Adresse [dez]	Daten [hex]	Daten [ASC]
N	24	,4'
N+1	44	,D'
N+2	34	,4'
N+3	46	,F'
N+4	34	,4'
N+5	32	,2'
N+6	35	,5'
N+7	39	,9'

# Service & Support



## Technical Support

Die Spezialisten des Technical Support beraten und unterstützen Sie bei Anfragen zu Funktionen und Handhabung unserer RFID-Produkte.

Sie erreichen uns weltweit Mo-Fr, werktags: 8-17 Uhr MEZ unter:

Telefon: ++49 (0) 180 5050-222

Fax: ++49 (0) 180 5050-223

## Internet

Sie erreichen uns im Internet unter:

Support-Homepage ([www.siemens.com/automation/service&support](http://www.siemens.com/automation/service&support))

Support-Anfragen beantworten wir Ihnen unter:

Web-Formular für Support Request ([www.siemens.de/automation/support-request](http://www.siemens.de/automation/support-request))

Allgemeine Neuigkeiten zum Kommunikationsmodul RF182C oder einen Überblick über unsere weiteren Identifikationssysteme finden Sie im Internet unter der Adresse:

RFID-Homepage ([www.siemens.de/simatic-sensors/rf](http://www.siemens.de/simatic-sensors/rf))

