



Your Global Automation Partner

TN-UHF-...-LNx UHF-Reader

Betriebsanleitung

Inhaltsverzeichnis

1	Über diese Anleitung	5
1.1	Zielgruppen	5
1.2	Symbolerläuterung	5
1.3	Weitere Unterlagen	5
1.4	Namenskonvention	5
1.5	Feedback zu dieser Anleitung	6
2	Hinweise zum Produkt	7
2.1	Produktidentifizierung	7
2.2	Lieferumfang	7
2.3	Turck-Service	7
3	Zu Ihrer Sicherheit	8
3.1	Bestimmungsgemäße Verwendung	8
3.2	Allgemeine Sicherheitshinweise	9
4	Produktbeschreibung	10
4.1	Geräteübersicht	10
4.1.1	Anzeigeelemente	10
4.2	Eigenschaften und Merkmale	11
4.3	Funktionsprinzip	11
4.4	Funktionen und Betriebsarten	12
4.4.1	Linux-Distribution – Software-Komponenten	12
4.5	Technisches Zubehör	12
5	Montieren	13
6	Anschließen	14
6.1	Geräte an Ethernet anschließen	14
6.2	Versorgungsspannung anschließen	15
6.3	Digitale Sensoren und Aktuatoren anschließen	16
6.4	Externe Antennen anschließen	17
7	In Betrieb nehmen	18
7.1	Reader mit dem Webserver parametrieren	19
7.1.1	Webserver öffnen	19
7.1.2	Einstellungen im Webserver bearbeiten	19
7.1.3	Multiplex-Betrieb	22
7.1.4	Antennenleistung einstellen	25
7.1.5	Antennenpolarisation einstellen	28
7.1.6	Presence Sensing Mode einschalten	31
7.1.7	RSSI-Wert übertragen – Communication	32
7.1.8	RSSI-Filter setzen – Post Read Filter	33
7.2	Reader mit dem Webserver testen	34
7.3	Netzwerk-Einstellungen anpassen	36
7.3.1	Netzwerk-Einstellungen über TAS (Turck Automation Suite) anpassen	36
7.3.2	Netzwerk-Einstellungen über den Webserver anpassen	38
7.4	RFID-Kanäle programmieren	39
7.4.1	RFID-Kanäle mit Python 3 programmieren	39
7.4.2	RFID-Kanäle über C oder C++ programmieren	41

7.5	Digitale Kanäle (DXP) programmieren	43
7.5.1	GPIOs der DXP-Kanäle – Übersicht	43
7.5.2	DXP-Funktionen über Skript einstellen	44
7.5.3	DXP-Kanäle mit Python 3 programmieren	45
7.5.4	DXP-Kanäle über Node.js programmieren	47
7.5.5	DXP-Kanäle über C oder C++ programmieren	49
7.6	LED-Funktionen programmieren	53
7.6.1	LEDs – Übersicht	53
7.6.2	LED-Funktionen über Skript einstellen	53
7.6.3	LED-Funktionen mit Python 3 programmieren	54
7.6.4	LED-Funktionen über Node.js programmieren	56
7.6.5	LED-Funktionen über C oder C++ programmieren	56
7.7	C-Applikation erstellen	58
7.8	Applikation automatisch starten (Autostart)	60
7.8.1	Autostart – Konfigurationsdatei (Unit-Datei) erstellen	60
7.8.2	Beispiel: Unit-Datei nutzen	60
7.8.3	Unit-Datei aktivieren	61
7.9	Zugriffsrechte verwalten	61
7.10	Python-Packages installieren	62
7.10.1	Beispiel: Python-Modul installieren	62
7.11	REST-API nutzen	65
7.11.1	REST-API im Webserver aktivieren	65
7.11.2	Übersicht der Befehle	66
7.11.3	Befehl: Process and inventory	67
7.11.4	Befehl: Read	69
7.11.5	Befehl: Write	71
7.11.6	Befehl: Write and verify	73
7.11.7	Befehl: Process a (universal RFID interface) request	75
7.12	RFID-Kanäle – Übersicht der Befehle	76
7.12.1	Befehl: Leerlauf	78
7.12.2	Befehl: Inventory	79
7.12.3	Befehl: Lesen	82
7.12.4	Befehl: Schreiben	84
7.12.5	Befehl: Schreiben mit Validierung	86
7.12.6	Befehl: Continuous Mode	88
7.12.7	Befehl: Daten aus dem Puffer lesen (Continuous Mode)	90
7.12.8	Befehl: UHF Continuous Presence Sensing Mode	92
7.12.9	Befehl: Continuous (Presence Sensing) Mode beenden	93
7.12.10	Befehl: Schreib-Lese-Kopf-Identifikation	94
7.12.11	Direkter Schreib-Lese-Kopf-Befehl	95
7.12.12	Befehl: Datenträger-Passwort setzen	97
7.12.13	Befehl: Schreib-Lese-Kopf-Passwort setzen	98
7.12.14	Befehl: Schreib-Lese-Kopf-Passwort zurücksetzen	99
7.12.15	Befehl: Datenträger-Schutz setzen	100
7.12.16	Befehl: Datenträger-Info	102
7.12.17	Befehl: UHF-Datenträger unwiderruflich deaktivieren (Kill)	104
7.12.18	Befehl: Einstellungen UHF-Schreib-Lese-Kopf wiederherstellen	105
7.12.19	Befehl: Backup der Einstellungen des UHF-Schreib-Lese-Kopfs	106
7.12.20	Befehl: Fehler/Status UHF-Schreib-Lese-Kopf lesen	107
7.12.21	Befehl: Reset	110
8	Betreiben	111
8.1	LED-Anzeigen	111
9	Störungen beseitigen	112

10 Instand halten 113

10.1 Firmware-Update über den Webserver durchführen 113

11 Reparieren 114

11.1 Geräte zurücksenden 114

12 Entsorgen 115

13 Technische Daten 116

14 EU-Konformitätserklärung 118

15 Turck-Niederlassungen – Kontaktdaten 119

1 Über diese Anleitung

Die Anleitung beschreibt den Aufbau, die Funktionen und den Einsatz des Produkts und hilft Ihnen, das Produkt bestimmungsgemäß zu betreiben. Lesen Sie die Anleitung vor dem Gebrauch des Produkts aufmerksam durch. So vermeiden Sie mögliche Personen-, Sach- und Geräteschäden. Bewahren Sie die Anleitung auf, solange das Produkt genutzt wird. Falls Sie das Produkt weitergeben, geben Sie auch diese Anleitung mit.

1.1 Zielgruppen

Die vorliegende Anleitung richtet sich an fachlich geschultes Personal und muss von jeder Person sorgfältig gelesen werden, die das Gerät montiert, in Betrieb nimmt, betreibt, instand hält, demontiert oder entsorgt.

1.2 Symbolerläuterung

In dieser Anleitung werden folgende Symbole verwendet:



GEFAHR

GEFAHR kennzeichnet eine gefährliche Situation mit hohem Risiko, die zum Tod oder zu schweren Verletzungen führt, wenn sie nicht vermieden wird.



WARNUNG

WARNUNG kennzeichnet eine gefährliche Situation mit mittlerem Risiko, die zum Tod oder zu schweren Verletzungen führen kann, wenn sie nicht vermieden wird.



VORSICHT

VORSICHT kennzeichnet eine gefährliche Situation mit mittlerem Risiko, die zu mittelschweren oder leichten Verletzungen führen kann, wenn sie nicht vermieden wird.



ACHTUNG

ACHTUNG kennzeichnet eine Situation, die zu Sachschäden führen kann, wenn sie nicht vermieden wird.



HINWEIS

Unter HINWEIS finden Sie Tipps, Empfehlungen und nützliche Informationen zu speziellen Handlungsschritten und Sachverhalten. Die Hinweise erleichtern Ihnen die Arbeit und helfen Ihnen, Mehrarbeit zu vermeiden.



HANDLUNGSAUFFORDERUNG

Dieses Zeichen kennzeichnet Handlungsschritte, die der Anwender ausführen muss.



HANDLUNGSRERESULTAT

Dieses Zeichen kennzeichnet relevante Handlungseresultate.

1.3 Weitere Unterlagen

Ergänzend zu diesem Dokument finden Sie im Internet unter www.turck.com folgende Unterlagen:

- Datenblatt
- Zulassungen
- Projektierungshandbuch

1.4 Namenskonvention

Schreib-Lese-Geräte werden im HF-Bereich als „Schreib-Lese-Köpfe“ und im UHF-Bereich als „Reader“ bezeichnet. Geläufige Synonyme für „Datenträger“ sind „Tag“, „Transponder“ und „mobiler Datenspeicher“.

1.5 Feedback zu dieser Anleitung

Wir sind bestrebt, diese Anleitung ständig so informativ und übersichtlich wie möglich zu gestalten. Haben Sie Anregungen für eine bessere Gestaltung oder fehlen Ihnen Angaben in der Anleitung, schicken Sie Ihre Vorschläge an techdoc@turck.com.

2 Hinweise zum Produkt

2.1 Produktidentifizierung

Diese Anleitung gilt für die folgenden UHF-Reader:

T N - UHF - Q300 - EU - LNX

T	N	Schreib-Lese-Gerät	-	UHF	Frequenzbereich	-	Q300	Bauform	-
----------	----------	--------------------	---	------------	-----------------	---	-------------	---------	---

Schreib-Lese-Gerät, Einbaubedingung
N nicht bündig
Turck RFID-System

Frequenzbereich
UHF UHF-Bereich

Bauform
Q180L300 quaderförmig
180 × 300 × 61,7 mm
Q300 quaderförmig
300 × 300 × 61,7 mm

EU	Einsatzregion	-	LNX	Software-Plattform
-----------	---------------	---	------------	--------------------

Einsatzregion
AUS Australien/Neuseeland
BRA Brasilien
CHN China
EU Europa
JPN Japan
KOR Korea
MYS Malaysia
NA Nordamerika
(USA, Kanada, Mexiko)
RUS Russland
SGP Singapur

Software-Plattform
LNX Linux

2.2 Lieferumfang

Im Lieferumfang sind enthalten:

- UHF-Reader
- Wandhalterung (Metallschiene)
- Kurzbetriebsanleitung

2.3 Turck-Service

Turck unterstützt Sie bei Ihren Projekten von der ersten Analyse bis zur Inbetriebnahme Ihrer Applikation. In der Turck-Produktdatenbank unter www.turck.com finden Sie Software-Tools für Programmierung, Konfiguration oder Inbetriebnahme, Datenblätter und CAD-Dateien in vielen Exportformaten.

Die Kontaktdaten der Turck-Niederlassungen weltweit finden Sie auf S. [► 119].

3 Zu Ihrer Sicherheit

Das Produkt ist nach dem Stand der Technik konzipiert. Dennoch gibt es Restgefahren. Um Personen- und Sachschäden zu vermeiden, müssen Sie die Sicherheits- und Warnhinweise beachten. Für Schäden durch Nichtbeachtung von Sicherheits- und Warnhinweisen übernimmt Turck keine Haftung.

3.1 Bestimmungsgemäße Verwendung

Die Reader mit integriertem RFID-Interface dienen zum berührungslosen Datenaustausch mit den RFID-Datenträgern im Turck-UHF-RFID-System. Die Arbeitsfrequenz der Geräte ist in der folgenden Tabelle beschrieben:

Typenbezeichnung	Arbeitsfrequenz	Einsatzbereich
TN-UHF-...-AUS-LNX	920...926 MHz	Australien, Neuseeland
TN-UHF-...-BRA-LNX	915...928 MHz	Brasilien
TN-UHF-...-CHN-LNX	920,5...924,5 MHz	China
TN-UHF-...-EU-LNX	865...868 MHz	Europa, Türkei, Indien
TN-UHF-...-JPN-LNX	916,7...920,9 MHz	Japan
TN-UHF-...-KOR-LNX	917...920,8 MHz	Korea
TN-UHF-...-MYS-LNX	919...923 MHz	Malaysia
TN-UHF-...-NA-LNX	902...928 MHz	Nordamerika (USA, Kanada, Mexiko)
TN-UHF-...-RUS-LNX	866...868 MHz	Russland
TN-UHF-...-SGP-LNX	920...925 MHz	Singapur

Die Geräte dürfen nur in Betrieb genommen werden, wenn die folgenden Bedingungen erfüllt sind:

- Der jeweilige Frequenzbereich ist für die Nutzung von UHF-RFID freigegeben.
- Der Arbeitsfrequenzbereich der Geräte stimmt mit dem regional zur Nutzung von UHF-RFID freigegebenen Bereich überein.
- Für die Einsatzregion liegt eine gültige Zertifizierung und/oder Zulassung vor, sofern gefordert.

Über das integrierte RFID-Interface können die Reader direkt über TCP/IP mit übergeordneten Systemen wie beispielsweise ERP-Systemen kommunizieren. Gelesene Daten werden über das Gerät an das übergeordnete System weitergegeben.

Zum Anschluss von digitalen Sensoren und Aktuatoren stehen vier konfigurierbare digitale Kanäle zur Verfügung.

Das Gerät darf nur wie in dieser Anleitung beschrieben verwendet werden. Jede andere Verwendung gilt als nicht bestimmungsgemäß. Für daraus resultierende Schäden übernimmt Turck keine Haftung.

3.2 Allgemeine Sicherheitshinweise

- Das Gerät erfüllt die EMV-Anforderungen für den industriellen Bereich. Bei Einsatz in Wohnbereichen Maßnahmen treffen, um Funkstörungen zu vermeiden.
- Nur fachlich geschultes Personal darf das Gerät montieren, installieren, betreiben, parametrieren und instand halten.
- Das Gerät nur in Übereinstimmung mit den geltenden nationalen und internationalen Bestimmungen, Normen und Gesetzen einsetzen.
- Ein längerer Aufenthalt im Strahlungsbereich von UHF-Readern kann gesundheitsschädlich sein. Mindestabstand von $> 0,35$ m zur aktiv ausstrahlenden Fläche des UHF-Readers einhalten.
- Die Strahlung der UHF-Reader kann elektrisch gesteuerte medizinische Hilfsmittel beeinflussen. Erhöhten Abstand zu aktiven Strahlungsquellen bis hin zur maximalen Sendereichweite einhalten.
- Default-Passwort des integrierten Webservers nach dem ersten Login ändern. Turck empfiehlt, ein sicheres Passwort zu verwenden.

4 Produktbeschreibung

Die Geräte sind in einem Aluminiumgehäuse in Schutzart IP67 ausgeführt. Die aktive Fläche besteht aus Kunststoff. Zur Verfügung stehen Geräte mit integrierter Antenne (Q300) oder zum Anschluss externer Antennen (Q180). Beide Gerätevarianten eignen sich zum Anschluss von bis zu vier externen, passiven UHF-RFID-Antennen.

Die Anschlüsse für das Ethernet und für digitale I/Os sind als M12-Buchsen ausgeführt. Zum Anschluss an die Spannungsversorgung besitzt das Gerät einen M12-Steckverbinder. Außerdem sind Anschlüsse für bis zu vier externe Antennen verfügbar.

4.1 Geräteübersicht

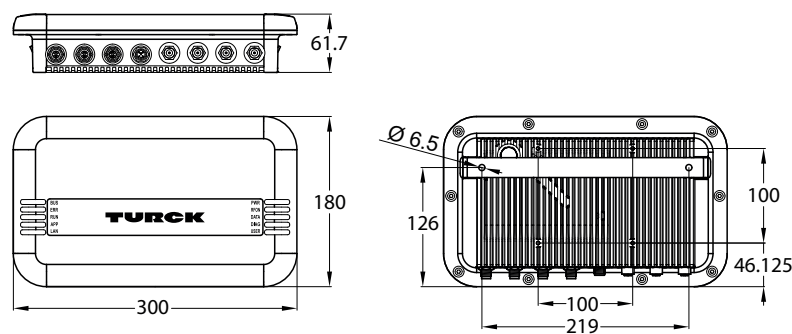


Abb. 1: Abmessungen – TN-UHF-Q180L300...

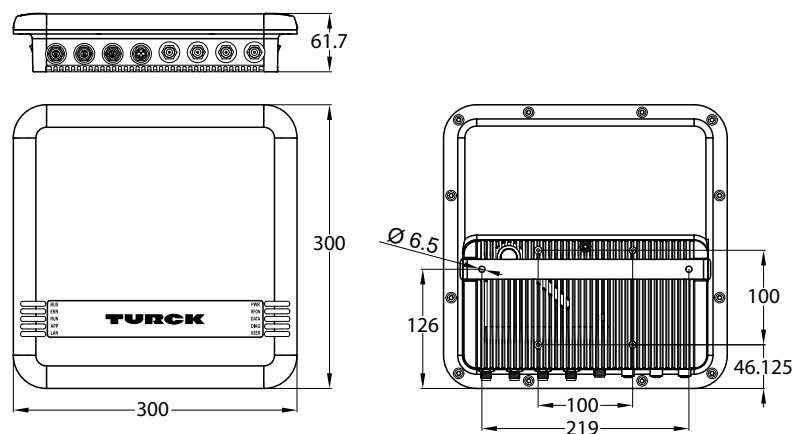


Abb. 2: Abmessungen – TN-UHF-Q300...

4.1.1 Anzeigeelemente

Das Gerät verfügt über folgende LED-Anzeigen:

- Versorgungsspannung
- Sammel- und Busfehler
- Status
- Diagnose

Zusätzlich kann über Software-Tools ein akustisches Signal eingestellt werden.

4.2 Eigenschaften und Merkmale

- TCP/IP
- Frei programmierbarer Ethernet-basierter Reader basierend auf Linux
- Programmiersprachen C, C++, NodeJS, Python
- Software-Komponenten: SSH, SFTP, HTTP, IBTP, MTXP, DHCP, SNTP, Node.js 6.9.5 (LTS), Python 3.x
- Implementierung des Protokolls erforderlich
- 2 W (ERP) maximale Ausgangsleistung
- 4 RP-TNC-Anschlüsse für passive, externe UHF-Antennen
- 4 konfigurierbare digitale Kanäle als PNP-Eingänge und/oder Ausgänge 2 A
- Übertragungsrate 10 Mbps/100 Mbps
- Integrierter Webserver
- LED-Anzeigen und Diagnosen

4.3 Funktionsprinzip

Die Reader dienen zum berührungslosen Datenaustausch mit Datenträgern. Dazu sendet die Steuerung über das Interface Befehle und Daten an den Reader und erhält die entsprechenden Antwortdaten vom Reader zurück. Beispiele für Befehle sind das Auslesen der IDs aller RFID-Datenträger im Lesebereich oder das Beschreiben eines RFID-Datenträgers mit einem bestimmten Produktionsdatum. Zur Kommunikation mit dem Datenträger werden die Daten vom Reader codiert und über ein elektromagnetisches Feld übertragen, das die Datenträger gleichzeitig auch mit Energie versorgt.

Ein Reader enthält einen Sender und einen Empfänger, eine Schnittstelle zum Interface und ein Kopplungselement (Spulen- bzw. Dipol-Antenne) für die Kommunikation mit dem Datenträger. Als Übertragungsverfahren zwischen Reader und Datenträger wird bei Geräten für den UHF-Bereich die elektromagnetische Wellenausbreitung genutzt.

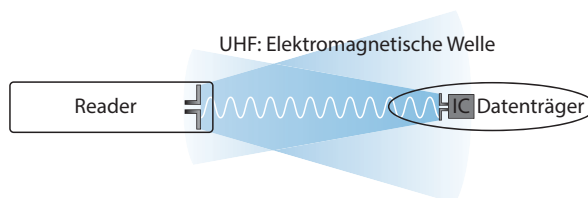


Abb. 3: Funktionsprinzip UHF-RFID

Die Antenne des Readers erzeugt elektromagnetische Wellen. Dadurch entsteht als sogenannte Luftschnittstelle ein Übertragungsfenster, in dem der Datenaustausch mit dem Datenträger stattfindet. Die Größe des Übertragungsfensters ist von den jeweils kombinierten Readern und Datenträgern sowie von den Umgebungsbedingungen abhängig.

Jeder Reader ist in der Lage, mit einer Reihe von Datenträgern zu kommunizieren. Dazu müssen Reader und Datenträger jeweils im gleichen Frequenzbereich arbeiten. Die Reichweiten der Geräte reichen – in Abhängigkeit von Leistung und Frequenz – von wenigen Millimetern bis zu mehreren Metern. Die angegebenen maximalen Schreib-Lese-Abstände stellen Werte unter Laborbedingungen ohne Materialbeeinflussung dar. Durch Bauteiltoleranzen, Einbausituation in der Applikation, Umgebungsbedingungen und die Beeinflussung durch Materialien (insbesondere Metall und Flüssigkeiten) können die erreichbaren Abstände abweichen.

4.4 Funktionen und Betriebsarten

Die Geräte arbeiten mit integrierter oder externer Antenne (TN-UHF-Q300...) bzw. ausschließlich mit externer Antenne (TN-UHF-Q180L300...). Mit den Geräten können passive UHF-Datenträger im Single- und Multitag-Betrieb ausgelesen und beschrieben werden. Dazu bilden die Geräte eine Übertragungszone aus, deren Größe und Ausdehnung u. a. von den verwendeten Datenträgern und den Einsatzbedingungen der Applikation abhängig sind. Die maximalen Schreib-Lese-Abstände sind in den Datenblättern aufgeführt. Die Geräte lassen sich mit Software-Tools über einen PC umfassend testen, konfigurieren und parametrieren.

Die Gerätefunktionen können über das Betriebssystem Linux mit C, C++, NodeJS oder Python programmiert werden. Zudem können Middleware-Funktionalitäten auf dem Gerät integriert werden.

An die konfigurierbaren digitalen Kanäle können Sensoren und Aktuatoren angeschlossen werden. Insgesamt lassen sich bis zu vier 3-Draht-PNP-Sensoren bzw. zwei PNP-DC-Aktuatoren mit einem maximalen Ausgangsstrom von 2 A pro Ausgang anschließen. Um die digitalen Kanäle als Ausgänge nutzen zu können, ist eine externe Spannungsversorgung erforderlich.

4.4.1 Linux-Distribution – Software-Komponenten

Die Linux-Distribution des Geräts enthält die folgenden Software-Komponenten:

- SSH
- SFTP
- HTTP
- IBTP
- MTXP
- DHCP
- SNTP
- Node.js 6.9.5 (LTS)
- Python 3.x

4.5 Technisches Zubehör

Optional erhältliches Zubehör für Montage, Anschluss und Parametrierung finden Sie in der Turck-Produktdatenbank unter www.turck.com. Das Zubehör ist nicht im Lieferumfang enthalten.

5 Montieren

Das Gerät ist zur Montage an einer Halterung nach VESA 100 × 100 vorgesehen. Für die Montage verfügt das Gerät über vier M4-Gewindebohrungen mit einem Abstand von 100 mm (horizontal und vertikal). Die max. Länge der Schrauben beträgt 8 mm zzgl. der Stärke der VESA-Halterung. Die Geräte können in beliebiger Ausrichtung montiert werden.

- Gerät mit vier M4-Schrauben an einer Halterung gemäß VESA 100 × 100 befestigen.

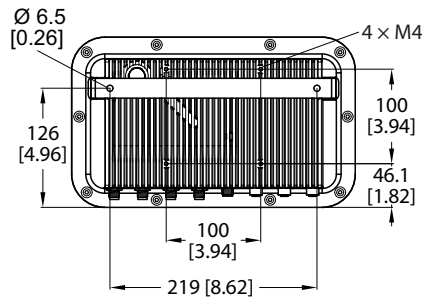


Abb. 4: Rückansicht – TN-UHF-Q180...

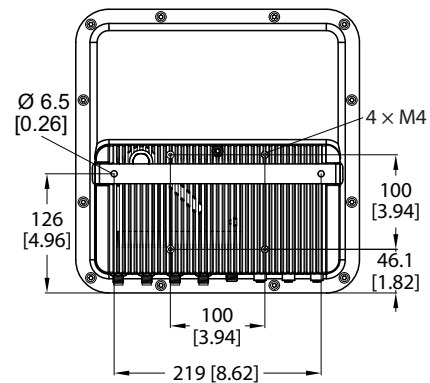


Abb. 5: Rückansicht – TN-UHF-Q300...

6 Anschließen

6.1 Geräte an Ethernet anschließen

Zum Anschluss an ein Ethernet-System verfügt das Gerät über eine 4-polige M12-Buchse.

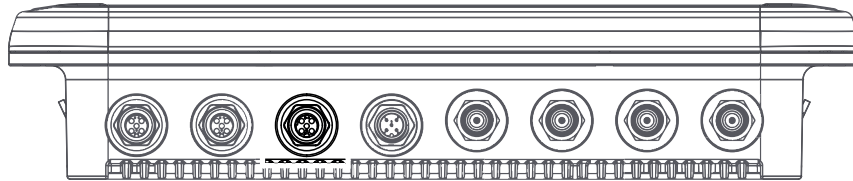


Abb. 6: M12-Ethernet-Steckverbinder

- Gerät gemäß unten stehender Pinbelegung an Ethernet anschließen (max. Anzugsdrehmoment 0,8 Nm).

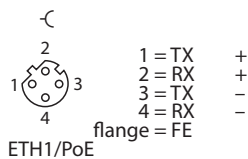


Abb. 7: Pinbelegung Ethernet-Anschlüsse



HINWEIS

Bei PoE wird die Versorgungsspannung über PoE Mode A mit 4-adrigen Leitungen übertragen.

Der Betrieb von PoE und 24 VDC gleichzeitig wird nicht unterstützt.

6.2 Versorgungsspannung anschließen

Zum Anschluss an die Versorgungsspannung verfügt das Gerät über einen 5-poligen M12-Steckverbinder.

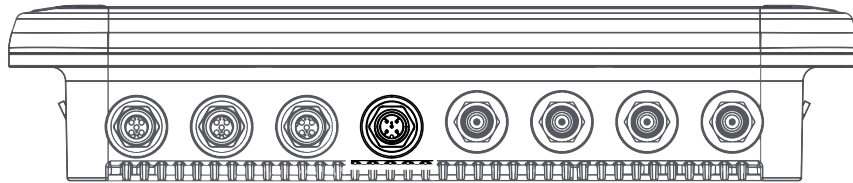


Abb. 8: M12-Steckverbinder zum Anschluss an die Versorgungsspannung

- Gerät gemäß unten stehender Pinbelegung an die Versorgungsspannung anschließen (max. Anzugsdrehmoment 0,8 Nm).

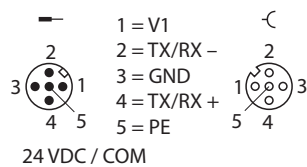


Abb. 9: Pinbelegung Versorgungsspannungs-Anschlüsse

6.3 Digitale Sensoren und Aktuatoren anschließen

Zum Anschluss von digitalen Sensoren und Aktuatoren verfügt das Gerät über zwei 5-polige M12-Steckverbinder.

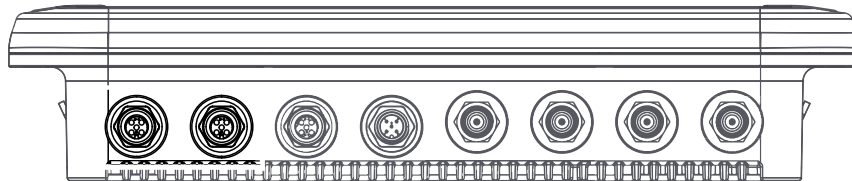


Abb. 10: M12-Steckverbinder zum Anschluss von digitalen Sensoren und Aktuatoren



HINWEIS

Beim Betrieb über PoE (Power over Ethernet) können die digitalen Kanäle nicht als Ausgänge genutzt werden.

- Sensoren und Aktuatoren gemäß unten stehender Pinbelegung an das Gerät anschließen (max. Anzugsdrehmoment 0,8 Nm).

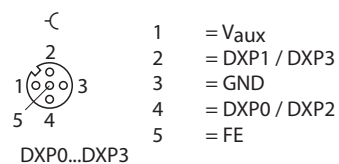


Abb. 11: Anschlüsse für digitale Sensoren und Aktuatoren – Pinbelegung

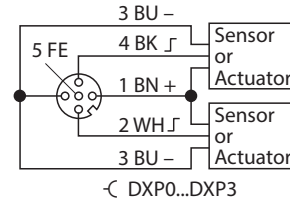


Abb. 12: Anschlüsse für digitale Sensoren und Aktuatoren – Anschlussbild

6.4 Externe Antennen anschließen

Zum Anschluss von bis zu vier externen Antennen verfügt das Gerät über vier RP-TNC-Buchsen. Die Eingangsimpedanz beträgt 50 Ω .

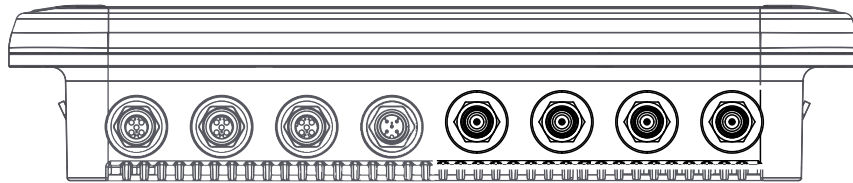


Abb. 13: RP-TNC-Buchsen zum Anschluss externer Antennen

- Externe Antennen mit einem Antennenkabel RP-TNC an das Gerät anschließen (max. Anzugsdrehmoment 0,8 Nm).

7 In Betrieb nehmen

Über das Betriebssystem Linux können mit C, C++, NodeJS und Python die Gerätefunktionen programmiert werden.

Um über die Konsole auf das Gerät zugreifen zu können, sind zusätzliche Software-Tools erforderlich (z. B. PuTTY). Zum Dateiaustausch zwischen dem Gerät und einem PC kann z. B. WinSCP genutzt werden. Per Default sind die folgenden Login-Daten auf dem Gerät hinterlegt:

User: user

Passwort: password



HINWEIS

Im Auslieferungszustand ist das Reader-Protokoll nicht implementiert. Das Protokoll muss durch den Anwender implementiert werden. Ab Firmware Version 1.2.5.0 entfällt die Implementierung des Protokolls durch den Anwender, da eine RFID REST API enthalten ist. Für die Parametrierung des Readers muss die REST API aktiviert sein (siehe Kapitel [► 65]).

7.1 Reader mit dem Webserver parametrieren

Über den integrierten Webserver können die Geräte eingestellt und Befehle an die Geräte geschickt werden. Um den Webserver mit einem PC öffnen zu können, müssen sich das Gerät und der PC im gleichen IP-Netzwerk befinden.

7.1.1 Webserver öffnen

Der Webserver lässt sich über einen Webbrowser oder über die Turck Automation Suite (TAS) öffnen. Der Aufruf des Webserver über TAS ist im Abschnitt „Netzwerk-Einstellungen anpassen“ beschrieben.

7.1.2 Einstellungen im Webserver bearbeiten

Zur Bearbeitung von Einstellungen über den Webserver ist ein Login erforderlich. Im Auslieferungszustand lautet das Passwort „password“.



HINWEIS

Turck empfiehlt, das Passwort aus Sicherheitsgründen nach dem ersten Login zu ändern.

- ▶ Webserver des Geräts öffnen.
- ▶ **Username** und **Password** eingeben.
- ▶ **Login** klicken.

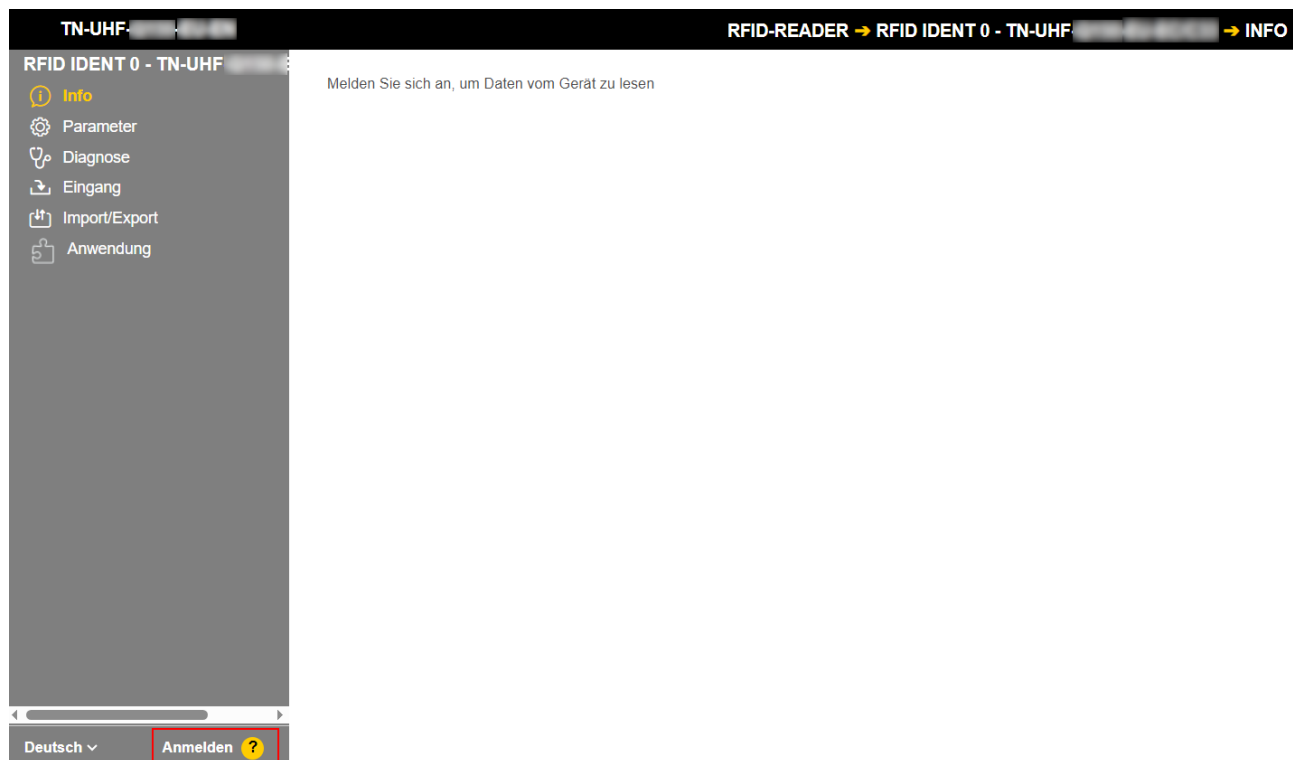


Abb. 14: Webserver – Login

- Nach dem Anmelden das Passwort ändern.

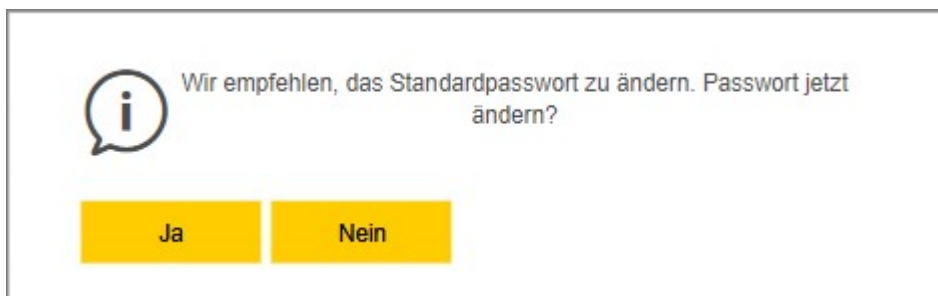


Abb. 15: Webserver – Passwort ändern

Nach dem Anmelden wird die Startseite mit den Geräteinformationen angezeigt.

- **RFID READER** anklicken, um die Geräteparameter anzuzeigen und einzustellen.

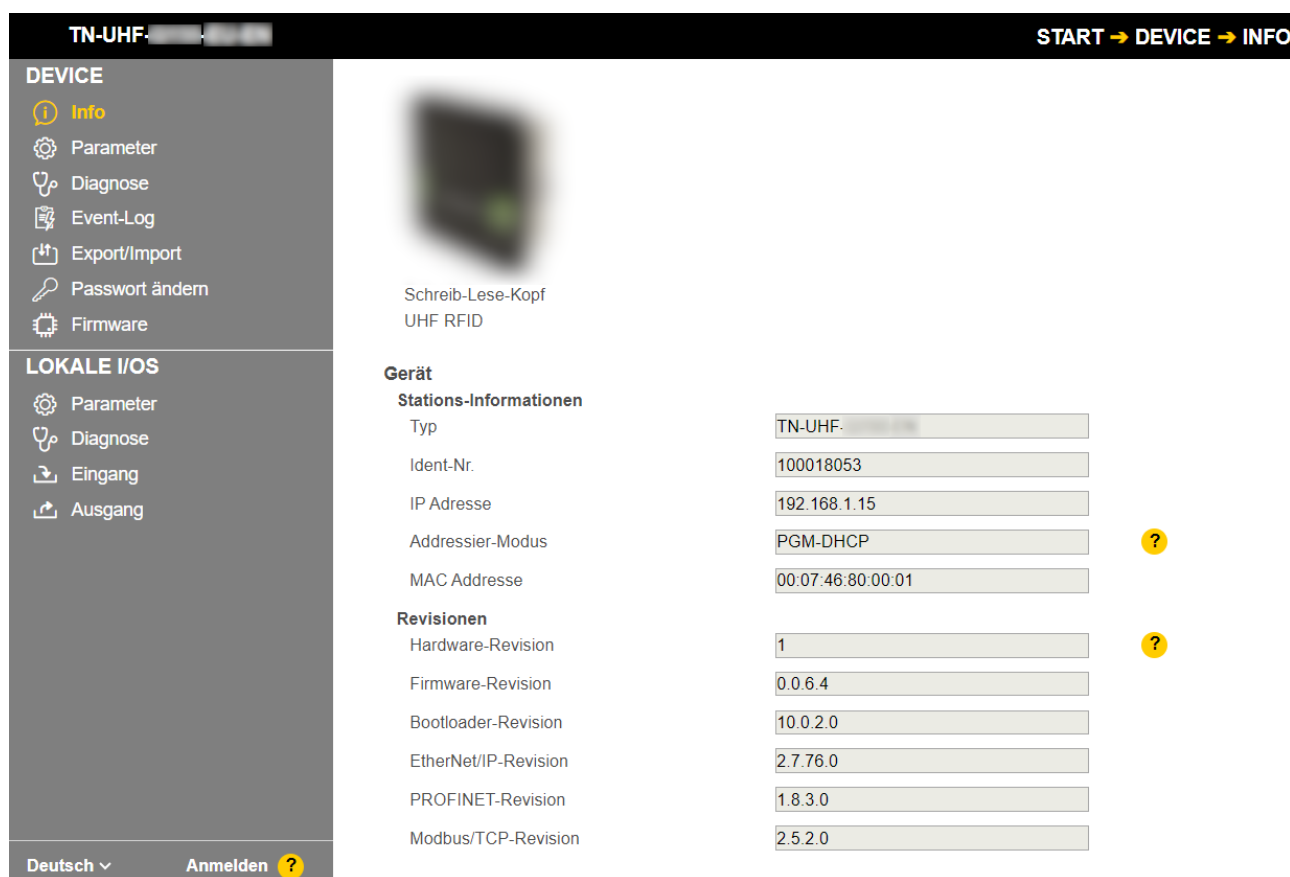


Abb. 16: Webserver – RFID Reader – Info

- ▶ In der Navigationsleiste am linken Bildrand **Parameter** anklicken.
- ⇒ Alle Parameter des Geräts werden angezeigt.

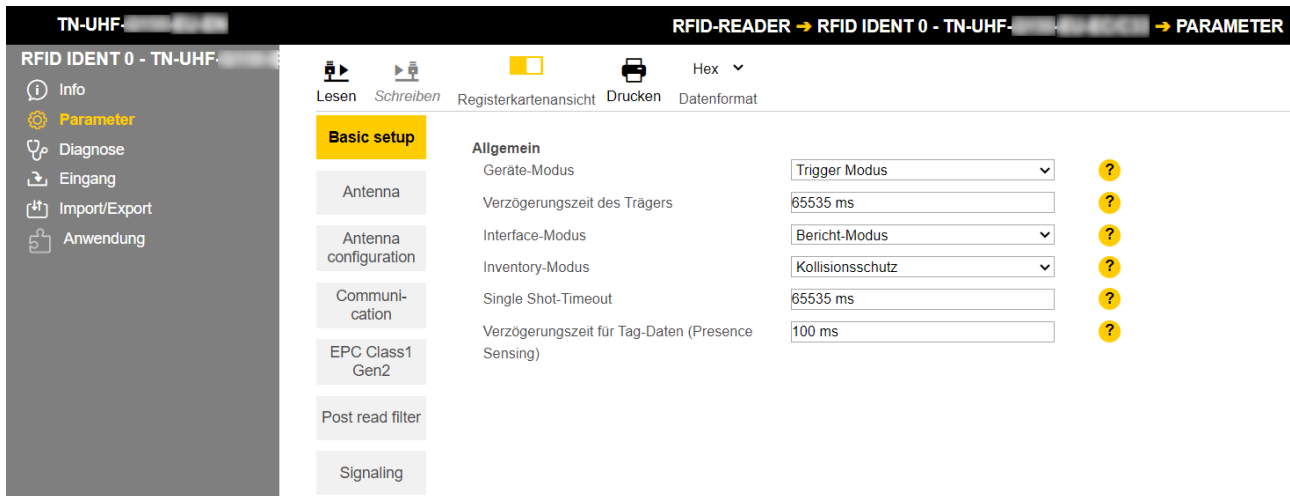


Abb. 17: Webserver – RFID Reader – Parameter

Die folgenden Setup-Fenster können aufgerufen werden:

- Basic setup
- Antenna
- Antenna configuration
- Communication
- EPC Class1 Gen2
- Post read filter
- Signaling

- ▶ Parameter setzen: **Write** klicken.



HINWEIS

Während ein Parameter gesetzt wird, leuchtet die LED ERR rot und wechselt automatisch zu grün.

7.1.3 Multiplex-Betrieb

Im Multiplex-Betrieb können mehrere Antennen sequenziell angesteuert oder eingeschaltet werden. Im unten angeführten Beispiel werden die Antennen nacheinander angesteuert. Der Multiplex-Betrieb kann aus bis zu 16 Abfolgen bestehen und lässt sich z. B. für Gate-Applikationen nutzen.

Zur Bearbeitung von Einstellungen über den Webserver ist ein Login erforderlich. Im Auslieferungszustand lautet das Passwort „password“.



HINWEIS

Turck empfiehlt, das Passwort aus Sicherheitsgründen nach dem ersten Login zu ändern.

- ▶ Webserver des Geräts öffnen.
- ▶ **Username** und **Password** eingeben.
- ▶ **Login** klicken.

Beispiel: Multiplex-Betrieb konfigurieren

- ▶ **RFID-READER** auswählen.
- ▶ **Parameter** auswählen.

RFID-READER → RFID IDENT 0 - TN-UHF → PARAMETER	
Basic setup	Allgemein
Antenna	Geräte-Modus: Trigger Modus
Antenna configuration	Verzögerungszeit des Trägers: 65535 ms
Communication	Interface-Modus: Bericht-Modus
EPC Class1 Gen2	Inventory-Modus: Kollisionsschutz
Post read filter	Single Shot-Timeout: 65535 ms
Signaling	Verzögerungszeit für Tag-Daten (Presence Sensing): 100 ms

Abb. 18: RFID-Reader – Parameter

► Antenna auswählen.

Section	Setting	Value	Help
Antennenmultiplexing	Anzahl der Einträge	1	?
	1. Eintrag	Externe Antenne 1	?
Frequenzeinstellungen	Art der Regelung	adaptive Frequenzagilität (AFA)	?
	Kanal bevorzugt (AFA)	7	?
Link Profil	Ausgewähltes Profil	Link-Profil 4	?
	Link-Profil 4: Kodierungsschema	Müller Stufe 4	?
	Link-Profil 4: TX-Datenrate	40 kHz	?
	Link-Profil 4: RX-Datenrate	320 kHz (Teilungsverhältnis = 64/3)	?
	Link-Profil 4: Teilungsverhältnis 64/3	aktiviert	?
	Link-Profil 4: Pilotton	aktiviert	?
	Link-Profil 4: DRM-Filter	deaktiviert	?
Andere	Link-Profil 4: Modus mit hoher Verstärkung	aktiviert	?
	Auto-Tuning-Funktion	aktiviert	?
	Intervall für die Selbstoptimierung	512 ms	?
	IF AGC-Verstärkung	-12 dB	?
	IF I NA Verstärkung	24 dB	?

Abb. 19: RFID-Reader – Parameter – Antenna

- Unter **Antenna Multiplexing** beim Punkt **Anzahl der Einträge** die Anzahl der Antennen eintragen.

► Antenna configuration auswählen.

RFID-READER -> RFID IDENT 0 - TN-UHF -> PARAMETER

Lesen Schreiben Registerkartenansicht Drucken Hex ▼

Antenna configuration

Allgemein

Stromversorgung externe Spannungsversorgung ?

Antenne RHCP (rechts-drehende zirkuläre Polarisation)

Abgestrahlte Leistung dBm e.r.p. ?

Abgestrahlte Leistung 15 ?

Abgestrahlte Leistung in Milliwatt 31.6 mW ?

Wechsel zur nächsten Antenne, wenn kein Transponder gelesen wurden deaktiviert ?

Maximale Sendezeit 65535 ms ?

Antenne LHCP (links-drehende zirkuläre Polarisation)

Abgestrahlte Leistung dBm e.r.p. ?

Abgestrahlte Leistung 10 ?

Abgestrahlte Leistung in Milliwatt 10.0 mW ?

Wechsel zur nächsten Antenne, wenn kein Transponder gelesen wurden deaktiviert ?

Maximale Sendezeit 65535 ms ?

Externe Antenne 1

Abgestrahlte Leistung dBm e.r.p. ?

Abgestrahlte Leistung 10 ?

Abgestrahlte Leistung in Milliwatt 10.0 mW ?

Dämpfung des Antennenkabels 0.00 dB ?

Deutsch ▼ Abmelden

Abb. 20: RFID-Reader – Parameter – Antenna configuration

- Beim Punkt **Maximale Sendezeit** für jede Antenne die Zeit einstellen, in der die Antenne aktiv bleiben soll.

7.1.4 Antennenleistung einstellen

Die Antennenleistung des Readers lässt sich applikationsspezifisch einstellen. Für die integrierte Antenne kann die abgestrahlte Leistung direkt eingetragen werden. Bei externen Antennen muss die Leistung berechnet werden.

Für die Berechnung der abgestrahlten Leistung (P_{ERP}) sind die folgenden Parameter relevant:

P_{cond}	Leistung, die an der TNC-Buchse des Readers ausgegeben wird
dB	Kabeldämpfung
G_{HW}	Antennengewinn der externen Antenne



HINWEIS

Kabeldämpfung und Antennengewinn entnehmen Sie den Datenblättern der eingesetzten Komponenten.

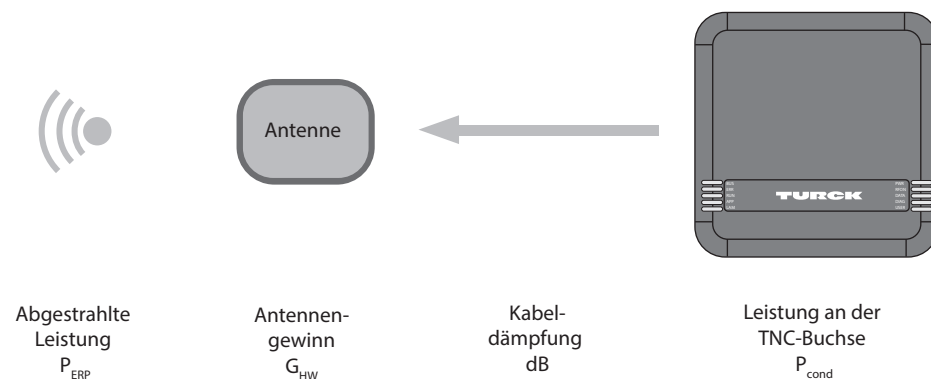


Abb. 21: Leistungsberechnung – Relevante Größen (schematische Darstellung)

Die Leistung kann mit der folgenden Formel berechnet werden:

$$P_{ERP} = G_{HW} - dB + P_{cond}$$

Antennenleistung einstellen – Einschränkungen durch Funkrichtlinien

Einige länderspezifische Richtlinien grenzen den Freiheitsgrad bei der Zusammenstellung eines RFID-Systems ein. Für die Einhaltung der Richtlinien sind Sie als Betreiber verantwortlich.

- ETSI
 - Abgestrahlte Leistung P_{ERP} : max. 33 dBm ERP
- FCC
 - Abgestrahlte Leistung P_{ERP} : max. 36 dBm EIRP
 - P_{cond} : max. 30 dBm bei Antennengewinn $G_{HW} \leq 6$ dBi



HINWEIS

Der Webserver kennzeichnet unzulässige Konfigurationen durch ein Ausrufezeichen. Eine Übertragung zum Gerät wird unterbunden.

Abgestrahlte Leistung berechnen

Die effektiv abgestrahlte Leistung (ERP) ist die Leistung, die von einer Antenne in den freien Raum abgestrahlt wird. Um die technischen Eigenschaften verschiedener Antennen vergleichen zu können, beziehen sich die Leistungsangaben immer auf eine Referenzantenne.

- EIRP = equivalent isotropic radiated power (Referenz: isotropischer Kugelstrahler)
- ERP = effective radiated power (Referenz: mit der Länge von $\lambda/2$)

Die abgestrahlte Leistung kann in Watt oder dBm angegeben werden. Die folgende Tabelle zeigt Näherungswerte zur Orientierung bei der Umrechnung zwischen dBm und mW:

dBm	mW	dBm	mW	dBm	mW	dBm	mW
1	1,25	9	8	17	50	25	316
2	1,6	10	10	18	63	26	400
3	2	11	13	19	80	27	500
4	2,5	12	16	20	100	28	630
5	3	13	20	21	125	29	800
6	4	14	25	22	160	30	1000
7	5	15	32	23	200
8	6	16	40	24	250	33	2000

Die Formel zur Ermittlung der exakten Werte lautet: **dBm = $10 \times \lg(P/1 \text{ mW})$**

Antennengewinn umrechnen

Der Antennengewinn kann in folgenden Einheiten angegeben werden:

- dBd Antennengewinn in Bezug auf einen Dipol
- dBi Antennengewinn in Bezug auf einen isotropischen Strahler (linear)
- dBic Antennengewinn in Bezug auf einen isotropischen Strahler (zirkular)

Die verschiedenen Einheiten lassen sich wie folgt umrechnen:

- $G_{\text{HW}} = \text{dBd}$
- $G_{\text{HW}} = \text{dBi} - 2,15$
- $G_{\text{HW}} = \text{dBic} - 5,15$

Leistung für externe Antennen einstellen

- ▶ Abgestrahlte Leistung unter **External Antenna 1** → **Radiated power** einstellen (hier: 24 dBm e.r.p.).
 - ▶ Kabeldämpfung dem Datenblatt der eingesetzten Leitung entnehmen.
 - ▶ Kabeldämpfung unter **Antenna cable attenuation** eintragen.
 - ▶ Antennengewinn dem Datenblatt der externen Antenne entnehmen.
 - ▶ Einheit für den Antennengewinn unter **Antenna gain unit** einstellen (hier: dBd).
 - ▶ Antennengewinn unter **Antenna gain** einstellen (hier: 5,00).
- ⇒ Die Leistung an der TNC-Buchse (P_{cond}) wird automatisch berechnet und unter **Conducted power** angezeigt.

The screenshot shows the 'Antenna configuration' tab in the TURCK RFID IDENT 0 - TN-UHF web interface. The 'External Antenna 1' section is highlighted with a red box, indicating the settings for the external antenna. The settings are as follows:

Section	Parameter	Value	Unit/Note
General	Power supply	external power supply	
	Antenna RHCP (right-handed circular polarization)		
Antenna RHCP	Radiated power unit	dBm e.r.p.	
	Radiated power	24 dBm e.r.p.	
Antenna LHCP	Radiated power in milliwatt	251.2 mW	
	Switch to next antenna if no transponder was read	disabled	
Antenna LHCP	Maximal transmit time	200 ms	
	Antenna LHCP (left-handed circular polarization)		
Antenna LHCP	Radiated power unit	dBm e.r.p.	
	Radiated power	27 dBm e.r.p.	
Antenna LHCP	Radiated power in milliwatt	501.2 mW	
	Switch to next antenna if no transponder was read	disabled	
Antenna LHCP	Maximal transmit time	200 ms	
	External Antenna 1		
External Antenna 1	Radiated power unit	dBm e.r.p.	
	Radiated power	24 dBm e.r.p.	
External Antenna 1	Radiated power in milliwatt	251.2 mW	
	Antenna cable attenuation	4.00 dB	
External Antenna 1	Antenna gain unit	dBd (dipole)	
	Antenna gain mode	Mid- and Far-field antenna (> 0 dBd)	
External Antenna 1	Antenna gain	5.00 dBd (dipole)	
	Conducted power	23.00 dBm	
External Antenna 1	Switch to next antenna if no transponder was read	disabled	

Abb. 22: Antennenleistung einstellen

- ▶ **Übernehmen** klicken, um die Einstellungen zu speichern.
- ▶ Leistung für jede weitere Antenne separat einstellen.

7.1.5 Antennenpolarisation einstellen

Die Antennenpolarisation kann über den Webserver oder über TAS eingestellt werden. Durch das Umschalten der Polarisation lassen sich durch Interferenzen verursachte Leselöcher verschieben. Die Erfassungsrate kann durch die Polarisationsumschaltung erhöht werden. Die Polarisationsumschaltung eignet sich z. B. für Singletag-Applikationen in besonders metallischen Umgebungen.

Die folgenden Grafiken stellen die Möglichkeiten der Antennenpolarisation schematisch dar.

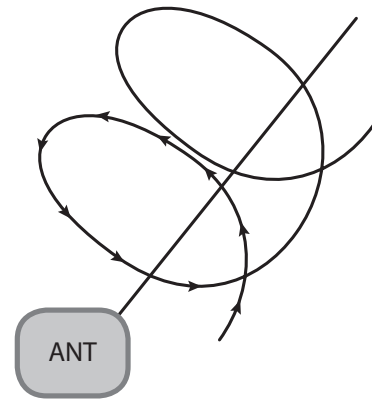
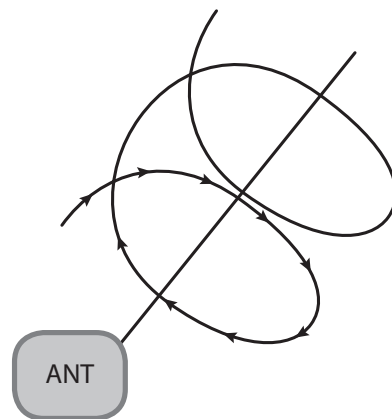


Abb. 23: Antennenpolarisation zirkular (RHCP) Abb. 24: Antennenpolarisation zirkular (LHCP)

Antennenpolarisation umschalten

Die Polarisationsumschaltung wird über die Multiplex-Einstellungen aktiviert.

- ▶ **Antenna** → **Number of entries** auf den Wert 2 einstellen.
- ▶ **Antenna** → **1st entry** auf den Wert **Antenna RHCP** einstellen.
- ▶ **Antenna** → **2nd entry** auf den Wert **Antenna LHCP** einstellen.

The screenshot shows the 'RFID IDENT 0 - TN-UHF' web interface. The top navigation bar includes 'TN-UHF', 'RFID READER → RFID IDENT 0 - TN-UHF', and 'PARAMETERS'. The left sidebar contains a menu with 'Info', 'Parameters' (highlighted), 'Diagnostics', 'Input', 'Import-/Export', 'Application', and 'Firmware'. The main content area is titled 'Antenna multiplexing' and includes a 'Basic setup' section with a sub-tab 'Antenna'. The 'Number of entries' is set to 2. The '1st entry' is set to 'external antenna 1' and the '2nd entry' is set to 'external antenna 2'. Below this, the 'Frequency settings' section includes 'Regulation type' (adaptive frequency agility (AFA)) and 'Channel preferred (AFA)' (04). The 'Link Profile' section includes 'Selected profile' (Link profile 4) and various settings for Link Profile 4: Coding scheme (Miller level 4), TX data rate (40 kHz), RX data rate (320 kHz (divide ratio = 64/3)), Divide ratio 64/3 (enabled), Pilot tone (enabled), DRM filter (disabled), and High gain mode (enabled). The 'Other' section includes 'Auto-tuning function' (enabled), 'Auto-tuning interval' (512 ms), 'IF AGC gain' (-12 dB), and 'IF LNA gain' (24 dB). Each setting has a dropdown arrow and a question mark icon.

Abb. 25: Antennenpolarisation umschalten

- ▶ Unter **Antenna configuration** → **Maximal transmit time** die Zeit bis zur Polarisationsumschaltung einstellen oder die Option **Switch to next antenna if no transponder was read** aktivieren.
- ⇒ Wenn die Option **Switch to next antenna if no transponder was read** aktiviert ist, wechselt der Reader nach einem Inventory-Vorgang ohne Lesung automatisch zur nächsten Multiplex-Sequenz (**Entry**).

TN-UHF- [REDACTED] RFID READER → RFID IDENT 0 - TN-UHF [REDACTED] → PARAMETER

RFID IDENT 0 - TN-UHF-

Info
Parameter
Diagnostics
Input
Import-/Export
Application

Reading Writing Tab view Print Data format

Basic setup

Antenna

Antenna configuration

Communication

EPC Class1 Gen2

Post read filter

Signaling

General

Power supply external power supply ?

Antenna RHCP (right-handed circular polarization)

Radiated power unit dBm e.r.p. ?

Radiated power 24 dBm e.r.p. ?

Radiated power in milliwatt 251.2 mW ?

Switch to next antenna if no transponder was read disabled ?

Maximal transmit time 200 ms ?

Antenna LHCP (left-handed circular polarization)

Radiated power unit dBm e.r.p. ?

Radiated power 27 dBm e.r.p. ?

Radiated power in milliwatt 501.2 mW ?

Switch to next antenna if no transponder was read disabled ?

Maximal transmit time disabled/enabled ?

External Antenna 1

Radiated power unit dBm e.r.p. ?

Radiated power 24 dBm e.r.p. ?

Radiated power in milliwatt 251.2 mW ?

Antenna cable attenuation 4.00 dB ?

Antenna gain unit dBd (dipole) ?

Antenna gain mode Mid- and Far-field antenna (> 0 dBd) ?

Antenna gain 5.00 dBd (dipole) ?

Conducted power 23.00 dBm ?

Switch to next antenna if no transponder was read disabled ?

Maximal transmit time 200 ms ?

Abb. 26: Polarisation automatisch umschalten

7.1.6 Presence Sensing Mode einschalten

Um den Befehl Continuous Presence Sensing Mode nutzen zu können, muss der Presence Sensing Mode im Reader aktiviert werden. Die Reader werden im Presence Sensing Mode automatisch eingeschaltet, sobald sich ein Datenträger im Erfassungsbereich befindet.

- Unter **Basic Setup** → **General** → **Device Mode** die Option **Presence sensing mode** einstellen.

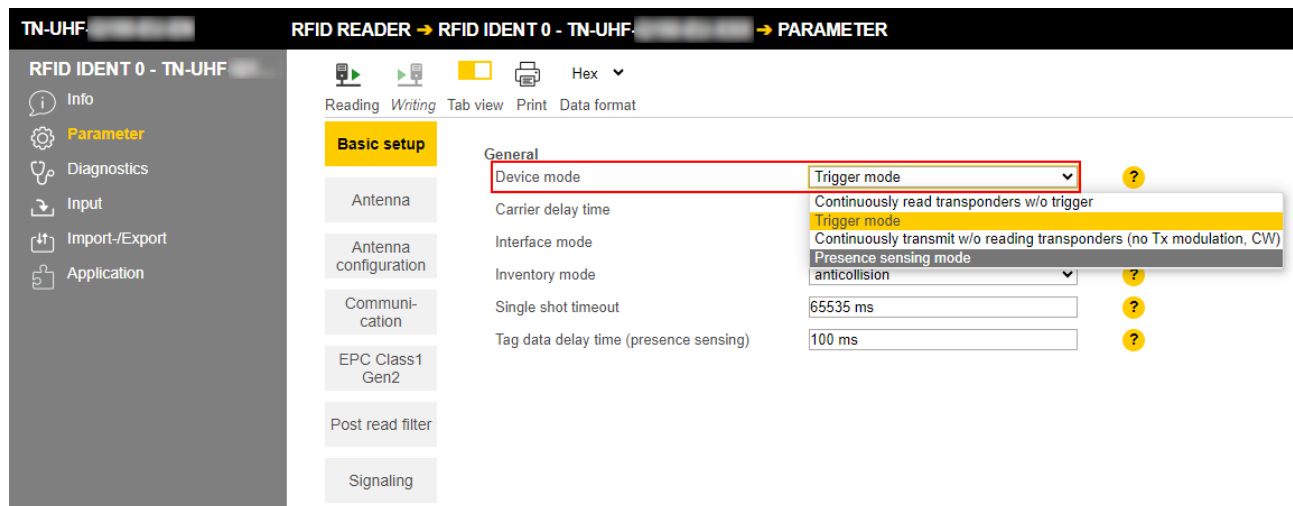


Abb. 27: Presence Sensing Mode einschalten

Im Zugriffslevel Advanced können die Parameter **Tag data delay time** und **Carrier delay time** individuell eingestellt werden.

- **Tag data delay time:** Zeitintervall, in dem der Reader nach einem Datenträger sucht. Wenn ein Datenträger gefunden wird, schaltet sich das Feld ein. Der Parameter ist im Zugriffslevel Basic per Default auf 100 ms eingestellt.
- **Carrier delay time:** Zeit, bis der Reader das Feld nach der letzten Lesung ausschaltet. Der Parameter ist im Zugriffslevel Basic per Default auf 65535 ms eingestellt.



HINWEIS

Für die Verwendung von RFID Test ist der Report Mode sinnvoll, da die gelesenen Datenträger-Informationen im RFID-Test-Fenster erscheinen und nicht einzeln gepollt werden müssen.

7.1.7 RSSI-Wert übertragen – Communication

Im Tab **Communication** können die Parameter für die Konfiguration der deBus-Nachrichten gesetzt werden. Alle Parameter und die einstellbaren Werte sind im Webserver beschrieben.

Beispiel: RSSI-Übertragung einschalten

- ▶ RSSI-Übertragung einschalten: Unter **Communication** → **Message data content** → **Transponder RSSI** die Option **enabled** wählen.

The screenshot shows the web interface of a TURCK RFID Reader. The top navigation bar includes 'TN-UHF', 'RFID READER', 'RFID IDENT 0 - TN-UHF', and 'PARAMETER'. The left sidebar contains navigation links: 'Info', 'Parameter' (highlighted), 'Diagnostics', 'Input', 'Import-/Export', and 'Application'. The main content area is divided into several sections: 'Basic setup', 'Antenna', 'Antenna configuration', 'Communication' (highlighted), 'EPC Class1 Gen2', 'Post read filter', and 'Signaling'. Under the 'Communication' tab, the 'Message data content' section is expanded, showing a list of parameters. The 'Transponder RSSI' parameter is highlighted with a red box, and its value is set to 'enabled'. Other parameters include 'Transponder phase', 'Transponder time', 'Transponder slot', 'Bad CRC', 'Customer ID prefix', 'MSG_IDENT_DATA', and 'Timestamp', all of which are currently set to 'disabled'. The 'Interval' parameter is set to '65535 s', and the 'Message type' is set to 'status (22 Byte)'. The 'Automatic sleep mode' is set to 'disabled'. The 'Message counter' section includes 'Acknowledge timeout' (1000 ms), 'Burst length' (20), and 'Counter' (disabled).

Abb. 28: RSSI-Übertragung einschalten

⇒ Der RSSI-Wert wird beim Inventory in den Lesedaten angezeigt.

7.1.8 RSSI-Filter setzen – Post Read Filter

Im Tab **Post Read Filter** können Parameter gesetzt werden, um Event-Nachrichten zu filtern.

Die eingestellten Filter reduzieren nicht den Datenverkehr auf der Luftschnittstelle und sind nicht für Multitag-Applikationen mit vielen Datenträgern oder hohen Überfahrgeschwindigkeiten geeignet. Alle Parameter und die einstellbaren Werte sind im Webserver beschrieben.

Beispiel: RSSI-Filter einstellen

Mit einem RSSI-Filter lassen sich unerwünschte Lesungen vermeiden. Alle Lesungen mit einem RSSI außerhalb der eingestellten Grenzwerte werden herausgefiltert und nicht angezeigt.

- ▶ Unter **Post read filter** → **RSSI filter** den RSSI-Filter einschalten.
- ▶ Grenzwerte einstellen unter **Post read filter** → **RSSI filter** → **Lower threshold**.

The screenshot shows the web interface of a TURCK RFID reader. The top navigation bar includes 'TN-UHF', 'RFID READER → RFID IDENT 0 - TN-UHF', and 'PARAMETER'. On the left, a sidebar lists menu items: 'Info', 'Parameter' (highlighted), 'Diagnostics', 'Input', 'Import-/Export', and 'Application'. The main content area is titled 'RFID IDENT 0 - TN-UHF' and contains a 'Post read filter' tab, which is active. Within this tab, the 'RSSI filter' section is highlighted with a red box. It shows 'RSSI Filter' set to 'enabled', 'Upper threshold' at '0 dBm', and 'Lower threshold' at '-45 dBm'. Other settings include 'Data match comparison' (disabled), 'Start' (0), 'No. of reads filter' (with a 'RESET' button), 'No. of reads' (0), and 'Timeout' (0 ms). Each setting has a help icon (question mark) to its right.

Abb. 29: RSSI-Filter einschalten

⇒ Beispiel: Alle Lesungen unterhalb eines RSSI-Werts von -45 dBm werden herausgefiltert.

7.2 Reader mit dem Webserver testen

Über die Funktion **Anwendung** können die Geräte mit dem Webserver getestet werden.

- RFID READER → **Anwendung** anklicken

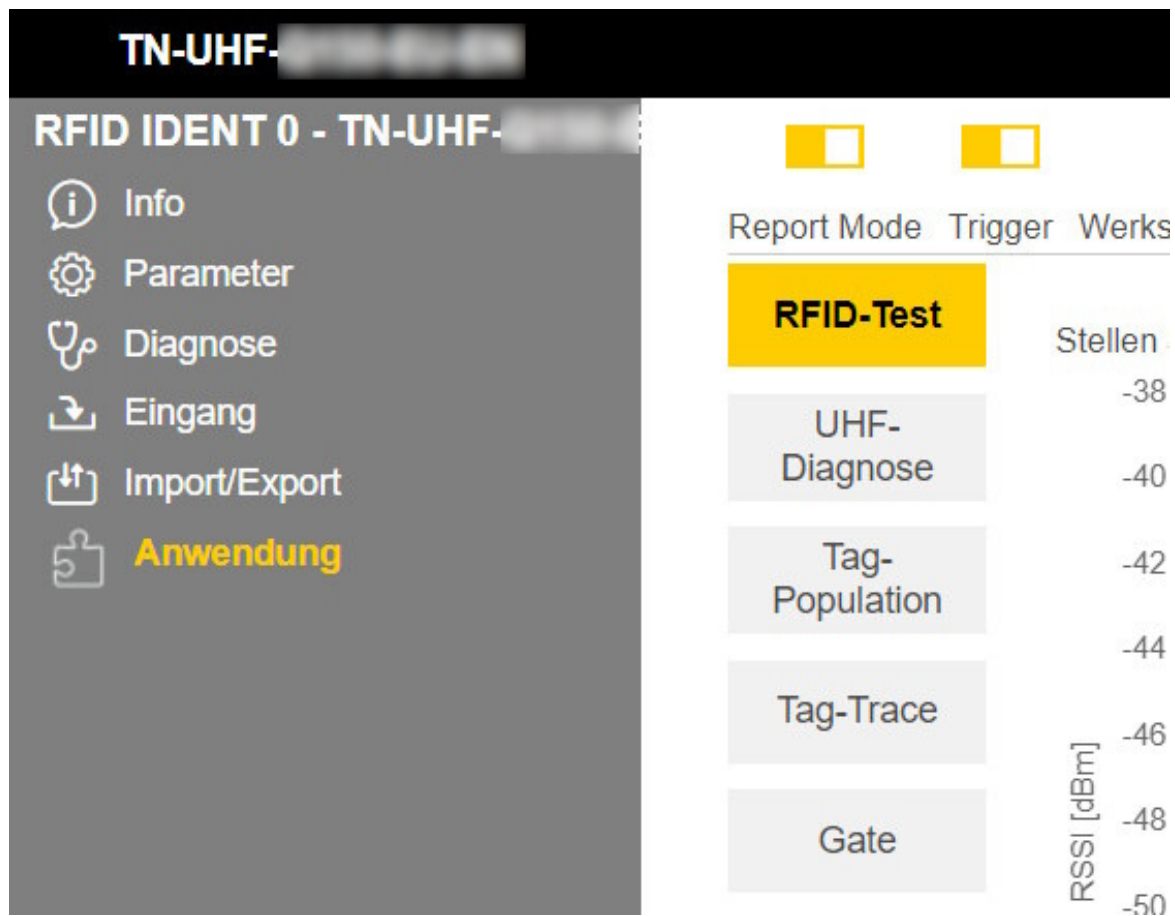


Abb. 30: Webserver – RFID-Reader – Anwendung

Im Bereich **Anwendung** stehen die Punkte **RFID-Test**, **UHF-Diagnose**, **Tag-Population**, **Tag-Trace** und **Gate** zur Verfügung:

- RFID-Test: Wenn der Trigger auf ON steht, wird das RF-Feld aktiviert und Datenträger können gelesen werden.
- UHF-Diagnose: Die Diagramme zeigen Interferenzfrequenzen aller verwendeten Kanäle.
- Tag-Population: Tool zur Ermittlung der abgestrahlten Leistung, ab der alle Datenträger gelesen werden können.
- Tag-Trace: Tool zur Erfassung einzelner Datenträger mit Kurvenverlauf der Signalstärke über die Zeit.
- Gate: Tool zur Erfassung mehrerer Datenträger (Pulkerfassung)

Über den **RFID-Test** können EPC-Informationen von Datenträgern im Singletag- und Multitag-Betrieb angezeigt und ausgelesen werden. Die empfangenen RSSI-Werte werden als Kurve mit zeitlichem Verlauf angezeigt.

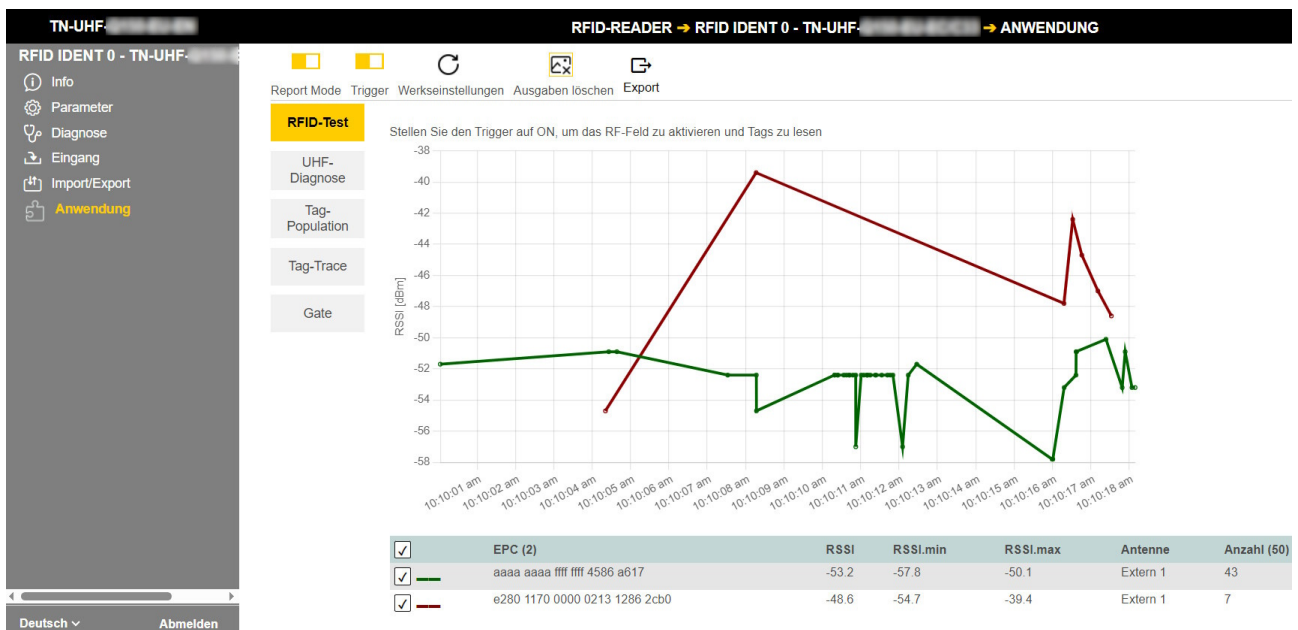


Abb. 31: Beispiel RFID-Test

Die **UHF-Diagnose** zeigt den aktuell empfangenen Leistungspegel des Readers pro Kanal an.

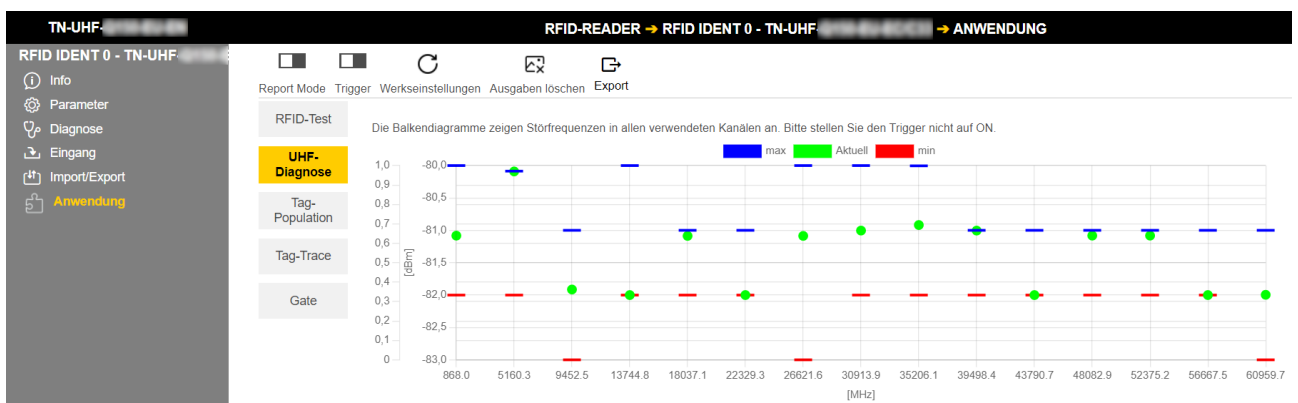


Abb. 32: Beispiel UHF-Diagnose

7.3 Netzwerk-Einstellungen anpassen

7.3.1 Netzwerk-Einstellungen über TAS (Turck Automation Suite) anpassen

Im Auslieferungszustand besitzt das Gerät die IP-Adresse 192.168.1.254. Die IP-Adresse kann über TAS (Turck Automation Suite) eingestellt werden. TAS steht unter www.turck.com kostenlos zur Verfügung.

- ▶ Gerät über die Ethernet-Schnittstelle mit einem PC verbinden.
- ▶ TAS öffnen.
- ▶ **Netzwerk scannen** klicken.

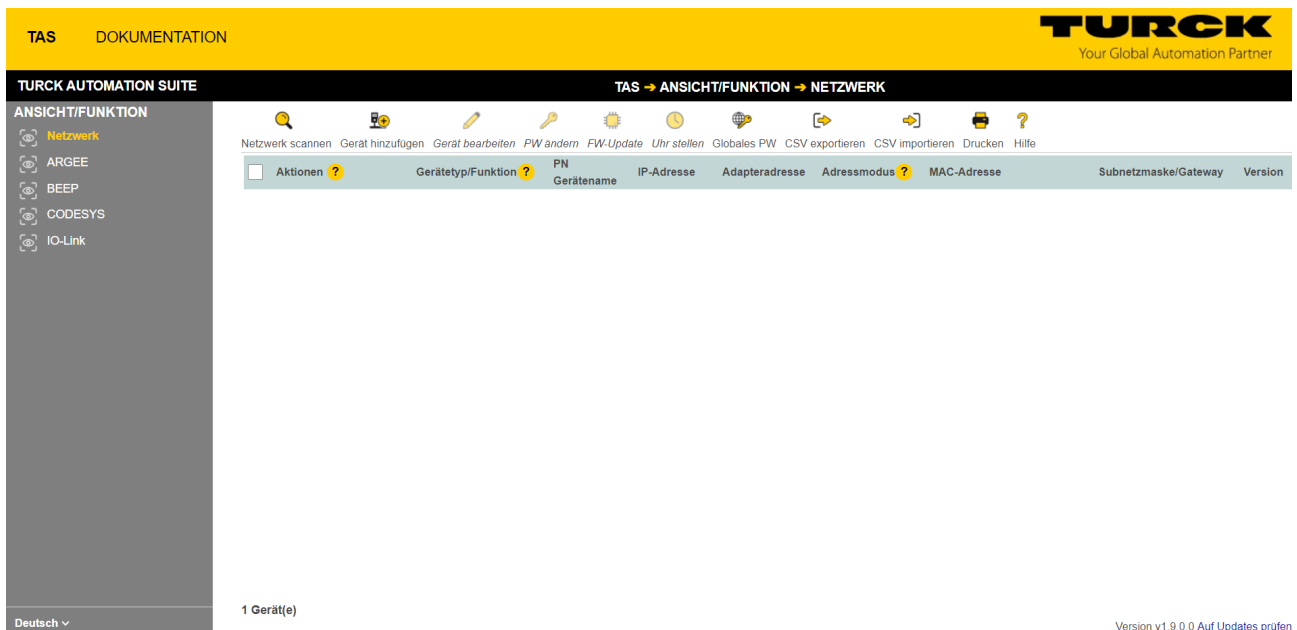


Abb. 33: TAS – Startbildschirm

⇒ TAS zeigt die angeschlossenen Geräte an.

TAS DOKUMENTATION **TURCK**
Your Global Automation Partner

TURCK AUTOMATION SUITE TAS → ANSICHT/FUNKTION → NETZWERK

ANSICHT/FUNKTION

- Netzwerk
- ARGE
- BEEP
- CODESYS
- IO-Link

Deutsch ▾

1 Gerät(e)

Version v1.9.0.0 [Auf Updates prüfen](#)

NETZWERK

Netzwerk scannen Gerät hinzufügen Gerät bearbeiten PW ändern FW-Update Uhr stellen Globales PW CSV exportieren CSV importieren Drucken Hilfe

Aktionen ?	Gerätetyp/Funktion ?	PN Gerätename	IP-Adresse	Adapteradresse	Adressmodus ?	MAC-Adresse	Subnetzmaske/Gateway	Version
<input type="checkbox"/>		TN-UHF-Q150-EN	192.168.1.15	192.168.1.245	PGM-DHCP	00:07:46:80:00:01	255.255.255.0 / 0.0.0.0	0.0.6.4

Abb. 34: TAS – gefundene Geräte

- Gewünschtes Gerät markieren (Checkbox).
- **Gerät bearbeiten** klicken.



HINWEIS

Ein Klick auf die IP-Adresse des Geräts öffnet den Webserver.

- IP-Adresse sowie ggf. Netzmaske und Gateway ändern.
- Änderungen mit einem Klick auf **NETZWERKDATEN EINSTELLEN** übernehmen.

Netzwerkeinstellungen bearbeiten

PN Gerätename	<input type="text"/>
IP-Adresse	<input type="text" value="192.168.1.15"/>
Standard-Gateway	<input type="text" value="0.0.0.0"/>
Subnetzmaske	<input type="text" value="255.255.255.0"/>

Achten Sie darauf, dass die IP-Adresse nicht von anderen Geräten oder Switches verwendet wird.

Abb. 35: TAS – Netzwerkeinstellungen bearbeiten

7.3.2 Netzwerk-Einstellungen über den Webserver anpassen



Um die IP-Adresse über den Webserver einstellen zu können, muss sich das Gerät im PGM-Modus befinden.

- ▶ Webserver öffnen.
- ▶ Als Administrator auf dem Gerät einloggen.
- ▶ **Parameter** → **Network** anklicken.
- ▶ IP-Adresse und ggf. Subnetzmaske sowie Default-Gateway ändern.
- ▶ Neue IP-Adresse, Subnetzmaske und Default-Gateway über **SET NETWORK CONFIGURATION** in das Gerät schreiben.

TN-UHF-

START → DEVICE → PARAMETERS

DEVICE

- Info
- Parameters**
- Diagnostics
- Status
- Event log
- Ex-/Import
- Change password
- Firmware

Hex ▼

Reading Writing Tab view Print Data format

YES	
REST-API activated	no ?
Date and time	
Settings	
Current time (UTC)	13.12.2024 08:51:28 📅
Set time from host	SET TIME FROM HOST ?
Timezone	UTC ▼ ?
Timer server	
SNTP enable	no ▼ ?
NTP server address	pool.ntp.org ?
Network	
Global	
MAC address	00:07:46:a7:4b:22
DNS-Mode	Automatic ▼ ?
DNS Domain	
DNS Name Server 1	0.0.0.0
DNS Name Server 2	0.0.0.0
DNS Name Server 3	0.0.0.0
Ethernet port 1	
Addressing mode	PGM-DHCP ?
Connection mode	Autonegotiation ▼ ?
IP address	192.168.1.254
Netmask	255.255.255.0
Default gateway	192.168.1.1
Set network configuration	SET NETWORK CONFIGURATION ?

Abb. 36: Netzwerk-Einstellungen über den Webserver anpassen

7.4 RFID-Kanäle programmieren

Der RFID-Kanal ist als serielle Schnittstelle (/dev/tty01 oder /dev/COM0) ausgeführt.

7.4.1 RFID-Kanäle mit Python 3 programmieren

Die folgenden Beispiele zeigen die Programmierung der RFID-Schnittstelle mit Python 3.

Beispiel 1: Modul „pySerial“ verwenden

```
import serial # from module pySerial

# open serial interface on port 0 and set a timeout of 8 seconds
seri = serial.Serial("/dev/COM0", timeout=8)

# change settings
seri.baudrate = 115200 # set the baudrate of port COM0 to 115200
seri.parity = 'N' # set no parity for port COM0
seri.bytesize = 7 # set the byte size for a sign to 7 for port
COM0
seri.stopbits = 1 # set stopbits to 1 for port COM0

seri.write(bytearray.fromhex("aa 07 07 49 00 41 23")) # writes a
bytestream

print(seri.readline()) # reads incoming message as ascii
```

Beispiel 2: Modul „periphery“ verwenden

```
from periphery import Serial

# Open /dev/COM0 with baudrate 115200, and defaults of 8N1, no
# flow control
serial = Serial("/dev/COM0", 115200)

# write a bytestream serial.write(bytearray.fromhex("aa 07 07 49
# 00 41 23"))

# Read up to 128 bytes with 500ms timeout
buf = serial.read(128, 0.5)

print(buf)
print("read %d bytes: _%s_" % (len(buf), buf))

serial.close()
```


7.4.2 RFID-Kanäle über C oder C++ programmieren

Die folgenden Beispiele zeigen die Programmierung der RFID-Schnittstelle mit Ansi C/C++.

```
#include <stdio.h>
#include <stdlib.h>
#include <termios.h>
#include <fcntl.h>

// initialize function (use extern for C++)
ssize_t read (int __fd, void *__buf, size_t __nbytes) __wur;
ssize_t write (int __fd, const void *__buf, size_t __n) __wur;
int close (int __fd);

int main(void) {
    //choose Interface for connection
    const char *Path = "/dev/COM0";
    struct termios options;
    int fd, count, i ;
    unsigned char currentBuff[1];
    unsigned char InBuff[255];
    unsigned char *p_InBuff = InBuff;
    unsigned char Message[] = {0x0D}; //deBus-Beispielbefehl für
    „Get status“
    if((fd = open((Path), O_RDWR | O_NOCTTY)) != -1)
    {
        // Set serial Interface
        tcgetattr (fd, &options);

        cfsetspeed(&options, B115200);
        options.c_cflags |= CS8|CSTOPB;
        options.c_flags &= ~(PARENB.PARODD)
        tcsetattr (fd, TCSANOW, &options);

        // write to Interface COM0
        if ((write(fd, Message, sizeof(Message))) == -1)
        {
            printf("not able to write...");
        }
    }
}
```

```
// read from Interface COM0
count = 0;
do
{
    if ((count += read(fd, currentBuff, 1)) == -1)
        printf("can not read...");
    *p_InBuff = currentBuff[0];
    p_InBuff++;
}while(currentBuff[0] != 0xfe);

// print:
p_InBuff -= count;
printf("\nData count: %i",count+1);
printf("\nValues: \n");
for(i = 0; i <= count ; i++)
{
    printf("%.02x ", *p_InBuff ); p_InBuff++;
}

// close the Interface
if((close(fd)) == -1)
{
    printf("\n can not close interface");
}
}
else
    printf("can not open interface\n");
return EXIT_SUCCESS;
}
```

7.5 Digitale Kanäle (DXP) programmieren

7.5.1 GPIOs der DXP-Kanäle – Übersicht

Die digitalen I/O-Kanäle (DXP) können über die GPIOs als Eingänge oder Ausgänge programmiert werden. Die GPIOs sind unter folgendem Pfad verfügbar:
/sys/class/gpio/...

Kanal	Steckplatz	Typ	GPIO	Mögliche Werte	
DXP0	C0	Eingang	59	0: Eingang aus (0V) 1: Eingang ein (24V)	
		Ausgang	88	0: Ausgang aus (0V) 1: Ausgang ein (24V)	
DXP1		Eingang	60	0: Eingang aus (0V) 1: Eingang ein (24V)	
DXP1		Ausgang	89	0: Ausgang aus (0V) 1: Ausgang ein (24V)	
DXP2	C1	Eingang	57	0: Eingang aus (0V) 1: Eingang ein (24V)	
		Ausgang	86	0: Ausgang aus (0V) 1: Ausgang ein (24V)	
DXP3		Eingang	58	0: Eingang aus (0V) 1: Eingang ein (24V)	
DXP3		Ausgang	87	0: Ausgang aus (0V) 1: Ausgang ein (24V)	

Zuschaltbare Spannungsversorgung VAUX einstellen

Steckplatz	Typ	GPIO	Mögliche Werte
C0	Ausgang VAUX0	41	0: VAUX aus
	Ausgang VAUX1	40	1: VAUX ein

Zuschaltbare Spannungsversorgung VAUX – Diagnose

Steckplatz	Typ	GPIO	Mögliche Werte
C0	Eingang VAUX0	56	0: VAUX fehlerfrei
	Eingang VAUX1	55	1: Fehler oder Überspannung an VAUX

7.5.2 DXP-Funktionen über Skript einstellen

Zum Einstellen der DXP-Kanäle ist auf dem Gerät ein Skript installiert. Das Skript ist unter folgendem Pfad verfügbar:

```
/usr/bin/dxp
```

Das Skript kann mit der folgenden Syntax genutzt werden:

```
/usr/bin/dxp channel [value]
```

Das folgende Beispiel setzt den Wert für den Kanal DXP0 auf „EIN“.

```
/usr/bin/dxp 0 1
```

Parameter	Mögliche Werte
DXP0...DXP3	1: Kanal einschalten 0: Kanal ausschalten

7.5.3 DXP-Kanäle mit Python 3 programmieren



HINWEIS

Die Geschwindigkeit der Datenübertragung ist abhängig von der konfigurierten Blockgröße und der eingestellten Übertragungsrate. Für zeitkritische Applikationen ist die Geschwindigkeit ggf. nicht ausreichend. Um eine schnellere Datenverarbeitung zu erreichen, kann der Prozess als Realtime-Prozess eingestellt werden.

Das folgende Beispiel zeigt die Programmierung der digitalen I/O-Kanäle mit Python 3.

```
import sys

#GPIOs-> OUT: IN:
ports = ["88","59"]

# write GPIO:
try:
    # set direction to write DXP
    fo = open("/sys/class/gpio/gpio" + ports[0] + "/direction",
"w")
    fo.write("out")
    fo.close()
    # write DXP:
    f = open("/sys/class/gpio/gpio" + ports[0] + "/value", "w")
    f.write("1")
    f.close()
except:
    # export gpio if not done as yet
    f1 = open("/sys/class/gpio/export", "w")
    f1.write(ports[0]) f1.close()
    # set direction to write DXP
    fo = open("/sys/class/gpio/gpio" + ports[0] + "/direction",
"w")
    fo.write("out")
    fo.close()
    # write DXP:
    fw = open("/sys/class/gpio/gpio" + ports[0] + "/value", "w")
    fw.write("1")
    fw.close()
```

```
# read GPIO:
try:
    # set direction to read DXP
    fo = open("/sys/class/gpio/gpio" + ports[1] + "/direction",
"w")
    fo.write("in")
    fo.close()
    # set active low to get the right value...
    fal = open("/sys/class/gpio/gpio" + ports[1] + "/active_low",
"w")
    fal.write("1")
    fal.close()
    # read DXP: fr = open("/sys/class/gpio/gpio" + ports[1] + "/va-
lue", "r")
    val=fr.read()
    fr.close()
    print(val)
except:
    # export gpio if not done as yet
    f1 = open("/sys/class/gpio/export", "w")
    f1.write(ports[1])
    f1.close()
    # set direction to read DXP
    fo = open("/sys/class/gpio/gpio" + ports[1] + "/direction",
"w")
    fo.write("in")
    fo.close()
    # set active low to get the right value...
    fal = open("/sys/class/gpio/gpio" + ports[1] + "/active_low",
"w")
    fal.write("1")
    fal.close()
    # read DXP:
    fr = open("/sys/class/gpio/gpio" + ports[1] + "/value", "r")
    val=fr.read()
    fr.close()
    print(val)
```

7.5.4 DXP-Kanäle über Node.js programmieren



HINWEIS

Die Geschwindigkeit der Datenübertragung ist abhängig von der konfigurierten Blockgröße und der eingestellten Übertragungsrate. Für zeitkritische Applikationen ist die Geschwindigkeit ggf. nicht ausreichend. Um eine schnellere Datenverarbeitung zu erreichen, kann der Prozess als Realtime-Prozess eingestellt werden.

Die folgenden Beispiele zeigen die Programmierung der digitalen I/O-Kanäle mit Node.js. Weitere Informationen zu Node.js und den Node.js-Packages finden Sie unter:

- <https://nodejs.org>
- <https://www.npmjs.com/>

```
// initialize the onoff box
const Gpio = require('onoff').Gpio;

function setGpioByInt(OUT, val) {
  // switch from DXP to GPIO...
  switch (OUT) {
    case 0:
      res = 88;
      break;
    case 1:
      res = 89;
      break;
    case 2:
      res = 86;
      break;
    case 3:
      res = 87;
      break;
  }

  // initialize the GPIO just to write...
  const DXP_Write = new Gpio(res, "out");
  // write the GPIO / DXP...
  DXP_Write.writeSync(val);
  console.log('set Gpio '+ res + ' to ' + val);
}
```

```
function getGpio(IN) {  
  // switch from DXP to GPIO...  
  switch (IN) {  
    case "0":  
      res = 59;  
      break;  
    case "1":  
      res = 60;  
      break;  
    case "2":  
      res = 57;  
      break;  
    case "3":  
      res = 58;  
      break;  
  }  
  
  // initialize the GPIO just to read...  
  const DXP_Read = new Gpio(res, "in");  
  // set active low to get the right value...  
  DXP_Read.setActiveLow('true');  
  // read the GPIO / DXP...  
  var res = DXP_Read.readSync();  
  console.log('Gpio ' + r_Pin + ' is: ' + res);  
  return res;  
}
```


7.5.5 DXP-Kanäle über C oder C++ programmieren

Das folgende Beispiel zeigt die Programmierung der digitalen I/O-Kanäle mit Ansi C/C++.

```
#include <stdio.h>
#include <stdlib.h>
#include <fcntl.h>

// initialize function (use extern for C++)
int access(const char *pfad, int modus);

int main(void) {
    //choose DXP / GPIO for connection

    char GPIO_IN_FILE[] = "/sys/class/gpio/gpio59";
    char GPIO_OUT_FILE[] = "/sys/class/gpio/gpio88";
    char input[2]; FILE *fh;

    /*
    =====
    =====
    READ:

    =====
    =====*/

    if( access( GPIO_IN_FILE, F_OK ) == -1 )
    {
        // file doesn't exist!
        // export gpio...
        if((fh = fopen("/sys/class/gpio/export", "w")) != 0)
        {
            fputs("59", fh);
            fclose(fh);
        }
        else
        {
            printf("failed on export to read...\n");
            printf("result: %i \n", (int)fh);
            return -1;
        }
    }
```

```
    }
    // set direction to read...
    if((fh =fopen("/sys/class/gpio/gpio59/direction", "w")) != 0)
    {
        fputs("in",fh);
        fclose(fh);
    }
    else
    {
        printf("failed on setting direction to read...\n");
        return -1;
    }

    // set active low to read...
    if((fh = fopen("/sys/class/gpio/gpio59/active_low", "w")) !=
0)
    {
        fputs("1",fh);
        fclose(fh);
    }
    else
    {
        printf("failed on setting active low ...\n");
        return -1;
    }
    // read GPIO...
    if((fh = fopen("/sys/class/gpio/gpio59/value", "r")) != 0)
    {
        fgets(input,2,fh);
        fclose(fh);
        printf("Value: %c\n", input[0]);
    }
    else
    {
        printf("failed on reading ...\n");
        return -1;
    }
}
```

```
/*  
  
=====
```

```
WRITE:  
  
=====
```

```
=====*/  
  
if( access( GPIO_OUT_FILE, F_OK ) == -1 )  
{  
    // file doesn't exist  
    // export gpio...  
    if((fh = fopen("/sys/class/gpio/export", "w")) != 0)  
    {  
        fputs("88",fh);  
        fclose(fh);  
    }  
    else  
    {  
        printf("failed on export to write...\n");  
        printf("result: %i \n", (int)fh);  
        return -1;  
    }  
}  
  
// set direction to read...  
if((fh = fopen("/sys/class/gpio/gpio88/direction", "w")) != 0)  
{  
    fputs("out",fh);  
    fclose(fh);  
}  
else  
{  
    printf("failed on setting direction to write...\n");  
    return -1;  
}  
  
// write GPIO...  
if((fh = fopen("/sys/class/gpio/gpio88/value", "w")) != 0)  
{
```

```
        fputs((const char*)"1",fh);  
        fclose(fh);  
    }  
    else  
    {  
        printf("failed on writing ...\n");  
        return -1;  
    }  
    return EXIT_SUCCESS;  
}
```

7.6 LED-Funktionen programmieren

7.6.1 LEDs – Übersicht

Das Gerät verfügt über drei frei programmierbare LEDs. Über Lese- und Schreibbefehle können die LEDs einzeln programmiert werden. Die LEDs sind auf dem System unter dem folgenden Pfad gemappt: „/sys/class/leds/...“

LED	Farbe	Systemname
APPL	grün	appl_green
	rot	appl_red
ERR	grün	err_green
	rot	err_red
RUN	grün	run_green
	rot	run_red

Wenn die Farben rot und grün einer LED gleichzeitig eingeschaltet sind, leuchtet die LED orange.



HINWEIS

Bei einem laufenden Firmware-Update wird die RUN-LED vom System genutzt.

7.6.2 LED-Funktionen über Skript einstellen

Zum Einstellen der LEDs ist auf dem Gerät ein Skript installiert. Das Skript ist unter folgendem Pfad verfügbar:

/TURCK/scripts/led.sh

Das Skript kann mit der folgenden Syntax genutzt werden:

```
sh led.h led color [value]
```

Das folgende Beispiel schaltet die rote LED „APPL“ ein:

```
sh led.sh appl red 1
```

LED	Mögliche Farbeinstellung	Mögliche Values
ERR	green/red	1: LED einschalten 0: LED ausschalten
RUN	green/red	1: LED einschalten 0: LED ausschalten
APPL	green/red	1: LED einschalten 0: LED ausschalten

7.6.3 LED-Funktionen mit Python 3 programmieren

Das folgende Beispiel zeigt die Programmierung der LED-Funktionen mit Python 3:

```
import sys
import time
# write red LEDs:
fw = open("/sys/class/leds/run_red/brightness", "w")
fw.write("1")
fw.close()

fw = open("/sys/class/leds/appl_red/brightness", "w")
fw.write("1")
fw.close()

fw = open("/sys/class/leds/err_red/brightness", "w")
fw.write("1")
fw.close()

# Wait for 5 seconds
time.sleep(5)

# write green LEDs:
fw = open("/sys/class/leds/appl_green/brightness", "w")
fw.write("1")
fw.close()

fw = open("/sys/class/leds/err_green/brightness", "w")
fw.write("1")
fw.close()

fw = open("/sys/class/leds/run_green/brightness", "w")
fw.write("1")
fw.close()

# Wait for 5 seconds
time.sleep(5)
```

```
# clean red LEDs:
fw = open("/sys/class/leds/run_red/brightness", "w")
fw.write("0")
fw.close()

fw = open("/sys/class/leds/appl_red/brightness", "w")
fw.write("0")
fw.close()

fw = open("/sys/class/leds/err_red/brightness", "w")
fw.write("0")
fw.close()

# Wait for 5 seconds
time.sleep(5)

# clean green LEDs:
fw = open("/sys/class/leds/appl_green/brightness", "w")
fw.write("0")
fw.close()

fw = open("/sys/class/leds/err_green/brightness", "w")
fw.write("0")
fw.close()

fw = open("/sys/class/leds/run_green/brightness", "w")
fw.write("0")
fw.close()
```

7.6.4 LED-Funktionen über Node.js programmieren

Die folgenden Beispiele zeigen die Programmierung der LED-Funktionen mit Node.js. Weitere Informationen zu Node.js und den Node.js-Packages finden Sie unter:

- <https://nodejs.org>
- <https://www.npmjs.com/>

```
// initialize the onoff box
const Gpio = require('onoff').Gpio;

//initialize the leds which are free for the user

appl_green_led = new LED('appl_green');
appl_red_led = new LED('appl_red');
error_green_led = new LED('err_green');
error_red_led = new LED('err_red');
run_green_led = new LED('run_green');
run_red_led = new LED('run_red');
```

7.6.5 LED-Funktionen über C oder C++ programmieren

Das folgende Beispiel zeigt die Programmierung der LED-Funktionen mit Ansi C/C++.

```
#include <stdio.h>
#include <stdlib.h>

// initialize function (use extern for C++)
char* strcpy(char* Ziel, const char* Quelle);
char* strcat(char* s1, const char* s2);

int main(void) {
    // LEDs for the customer:
    char *appl_green_led = "appl_green";
    char *appl_red_led = "appl_red";
    char *error_green_led = "err_green";
    char *error_red_led = "err_red";
    char *run_green_led = "run_green";
    char *run_red_led = "run_red";
    char *LED_FILE = "/sys/class/leds/";
    char *brightness = "/brightness";
    FILE *fh;
```



```
char cur_Str[50] = {0};
strcpy(cur_Str, LED_FILE);
// take LED which will shine:
strcat(cur_Str, run_red_led);
strcat(cur_Str, brightness);

//WRITE:
printf("string to led: %s\n", cur_Str);

// write LED...
if((fh = fopen(cur_Str, "w")) != 0)
{
    // write "1" to switch on and "0" to switch of LED
    fputs((const char*)"0", fh);
    fclose(fh);
}
else
{
    printf("failed on writing ...\n");
    return -1;
}
return EXIT_SUCCESS;
}
```

7.7 C-Applikation erstellen

Voraussetzungen

Um eine C-Applikation erstellen zu können, sind folgende Komponenten erforderlich:

- Toolchain für Cortex A8
- C-Programm

Toolchain herunterladen

Um ein C-Programm cross-kompilieren zu können, benötigen Sie die folgende Toolchain für den Cortex A8-Prozessor:

- OSELAS.toolchain-2014.12.0-arm-cortexa8-linux-gnueabi-hf-gcc-4.9.2-glibc-2.20-binutils-2.24-kernel-3.16-sanitized

Die Toolchain ist unter <http://debian.pengutronix.de/debian> zum Download erhältlich.

Beispiel: C-Programm erstellen

- ▶ Datei „hello.c“ erstellen.
- ▶ Folgenden Text in die Datei kopieren:

```
// hello.c
#include <stdio.h>

int main() {
    printf("Hello World!\n");
    return 0;
}
```

- ▶ Ausführbare Datei über den folgenden Toolchain-Befehl erstellen:
`/opt/OSELAS.Toolchain-2014.12.0/arm-cortexa8-linux-gnueabi-hf-gcc-4.9.2-glibc-2.20-binutils-2.24-kernel-3.16-sanitized/bin/arm-cortexa8-linux-gnueabi-hf-gcc -o helloExample hello.c`

Beispiel: C-Programm über ein Makefile erstellen

Das Dienstprogramm „make“ automatisiert das Erstellen von ausführbaren Dateien aus Quellcode. Über „make“ lassen sich C-Programme kompilieren. Dazu wird ein Makefile verwendet, das Regeln zur Erstellung von ausführbaren Dateien enthält.

Das folgende Beispiel zeigt ein einfaches Makefile:

```
all: helloExample
```

```
helloExample: hello.o
```

```
    /opt/OSELAS.Toolchain-2014.12.0/arm-cortexa8-linux-gnueabi/f/
gcc-4.9.2-glibc-2.20-binutils-2.24-kernel-3.16-sanitized/bin/arm-
cortexa8-linux-gnueabi/f/gcc -o helloExample hello.o
```

```
hello.o: hello.c
```

```
    /opt/OSELAS.Toolchain-2014.12.0/arm-cortexa8-linux-gnueabi/f/
gcc-4.9.2-glibc-2.20-binutils-2.24-kernel-3.16-sanitized/bin/arm-
cortexa8-linux-gnueabi/f/gcc -c hello.c
```

```
clean:
```

```
    rm hello.o helloExample
```

```
}
```

- ▶ Makefile erstellen.
- ▶ Makefile im selben Ordner wie die C-Applikation speichern.
- ▶ Makefile mit dem Befehl „make“ ausführen.
- ⇒ Das C-Programm wird installiert.

7.8 Applikation automatisch starten (Autostart)

Eine Applikation kann mit der Autostart-Funktion nach dem Start des RFID-Interfaces automatisch ausgeführt werden. Dazu muss eine Konfigurationsdatei (Unit-Datei) erstellt, in das Gerät geschrieben und aktiviert werden

7.8.1 Autostart – Konfigurationsdatei (Unit-Datei) erstellen

- ▶ Unit-Datei mit der Endung „.service“ erstellen.

Beispiel: Die Unit-Datei „setdxp.service“ startet eine Node.js-Applikation, mit der die DXP-Kanäle bei jedem Neustart des Geräts getriggert werden.

- ▶ Über „ExecStart“ die Applikation aufrufen, die bei jedem Neustart des Interfaces aufgerufen werden soll:

`ExecStart=path_to_programm app/file`

Optional kann ein Parameter übertragen werden. Beispiel: `ExecStart=path_to_programm app/file parameter`

Weitere Informationen zu Unit-Konfigurationsdateien finden Sie unter:

<https://www.freedesktop.org/software/systemd/man/systemd.service.html>

Beispiel: Autostart einer Applikation mit Parameter-Transfer

`ExecStart=/usr/bin/node /home/user/ hello_GPIO.js webactive`

7.8.2 Beispiel: Unit-Datei nutzen

Im folgenden Beispiel wird die Node.js-Datei „hello_GPIO.js“ heruntergeladen und unter „/home/user“ gespeichert:

[Unit]

Description= trigger the DXPs

#After=Service_that_must_run_before.service

[Service] Type=simpleExecStart=/usr/bin/node /home/user/ hello_GPIO.js

[Install]

WantedBy=multi-user.target

- ▶ Beispiel-Datei in „/etc/systemd/system/“ erstellen:
`sudo touch /etc/systemd/system/setdxp.service`
- ▶ Erstellte Datei öffnen:
`sudo nano /etc/systemd/system/setdxp.service`
- ▶ Oben aufgeführten Quelltext in die geöffnete Datei einfügen.

7.8.3 Unit-Datei aktivieren

Nach dem Erstellen muss die Unit-Datei über den systemctl-Befehl aktiviert werden. Zum Aktivieren sind Zugriffsrechte auf das Root-Verzeichnis erforderlich. Die Angabe der Dateierendung .services ist optional und kann entfallen.

- Unit-Datei über folgenden Befehl aktivieren:
`sudo systemctl enable setdvp.service`

Der erstellte Symlink lautet:

```
/etc/systemd/system/multi-user.target.wants/setdvp.service → /etc/systemd/system/setdvp.service
```

Unit-Datei deaktivieren

- Unit-Datei über folgenden Befehl deaktivieren:
`sudo systemctl disable setdvp.service`

7.9 Zugriffsrechte verwalten

Das Gerät unterstützt die Standard-Linux-Benutzerverwaltung. Die Zugriffsrechte können über die folgenden Standard-Linux-Tools verwaltet werden:

- adduser
- addgroup
- passwd

Benutzer	Rechte	Passwort
root	Systemadministrator (alle Zugriffsrechte)	turck
user	eingeschränkte Zugriffsrechte und Konsolenrechte	password
sftpuser	Zugriffsrechte, SFTP-Rechte im Verzeichnis /home	password

7.10 Python-Packages installieren

Module, Bibliotheken und andere Software können über das BSP (Board Support Package) mit dem Distributionstool PTXdist konfiguriert und auf das Gerät geladen werden. Wenn Pakete in eine bestehende Firmware integriert werden sollen, müssen sie zuvor mit PTXdist erstellt werden. PTXdist ist unter <https://www.pengutronix.de/de/software/ptxdist.html> zum Download erhältlich.

Auf dem Gerät ist zum Integrieren von Software-Paketen der ipkg-Paketmanager (Itsy Package Management System) installiert. Mit dem ipkg-Paketmanager können auch Python-Module nachträglich installiert werden.

7.10.1 Beispiel: Python-Modul installieren

Das folgende Beispiel erläutert das Vorgehen bei der Installation des Python-Moduls sh. Das Python-Modul wird dabei nachträglich in eine bestehende Firmware integriert.

Voraussetzungen

- PTXdist ist auf dem Linux-Hostsystem installiert.
- Das erforderliche Python-Modul wurde heruntergeladen (Beispiel: <https://amoffat.github.io/sh/>).

Beispiel: Python-Modul sh installieren

Um das Python-Modul sh mit PTXdist erstellen zu können, muss zunächst ein Rule File erstellt werden.

- ▶ Rule File mit folgendem Befehl erstellen:
`$ ptxdist newpackage target`
- ▶ Interaktiv-Angaben zum Paket erstellen:
Output:
`ptxdist: creating a new 'target' package:
ptxdist: enter package name.....:
sh ptxdist: enter version number.....:
1.12.13 ptxdist: enter URL of basedir.:https://github.com/
amoffat/sh/archive/
ptxdist: enter suffix.....: tar.gz
ptxdist: enter package author.....: Your Name <E-Mail>
ptxdist: enter package section.....: Python3`

`generating rules/sh.make
generating rules/sh.in`

Die Files sh.make und sh.in werden automatisch erstellt.

- ▶ Wenn bekannt, den MD5-Schlüssel des Pakets als Parameter SH_MD5 im File sh.make eintragen.
- ▶ Parameter SH_CONF_TOOL im File sh.make auf das entsprechende Tool einstellen (hier: Python 3).
`SH_CONF_TOOL :=python3`
- ▶ Wenn ein Python-Modul über einen separaten Unterordner verfügt: Unterordner im Zielverzeichnis erstellen (in diesem Beispiel nicht erforderlich):
`@$(call install_copy, Modul, 0, 0, 0755, $(PYTHON3_SITEPACKAGES)/foldername)`
- ▶ Im Bereich #Target-Install den Installationsort des Python-Moduls im Zielsystem angeben (Beispiel: sh-Modul):

```
@for file in `find $(SH_PKGDIR)/usr/lib/python$(PYTHON3_MAJORMINOR)/site-packages` \
! -type d ! -name "*.py" -printf "%P\n"; do \
$(call install_copy, sh, 0, 0, 0644, -, \
/usr/lib/python$(PYTHON3_MAJORMINOR)/site-packages/$$file); \
done
```

Im File sh.in können Abhängigkeiten eingetragen werden. Im folgenden Beispiel muss Python 3 vorhanden sein, um Python-Module installieren zu können. Auf dem Hostsystem muss das Modul „setuptools“ vorhanden sein.

- ▶ Abhängigkeiten wie folgt eintragen:

```
## SECTION=python3
config PYTHON_SH
tristate
select PYTHON3 # Python 3 must be installed
select HOST_PYTHON3_SETUPTOOLS # Setuptools must be installed on the host
prompt "sh"
help
FIXME
```
- ▶ Kompilieren.

Damit das sh-Modul beim nächsten Build mit erzeugt wird, muss das Modul in „menuconfig“ ausgewählt werden:

- ▶ „menuconfig“ über folgenden Befehl öffnen:
`ptxdist menuconfig`
- ▶ Über „Scripting Languages“ → „python3 Extra Modules“ zu den Python 3-Modulen navigieren.
- ▶ sh-Modul auswählen.
- ▶ Konfiguration speichern.

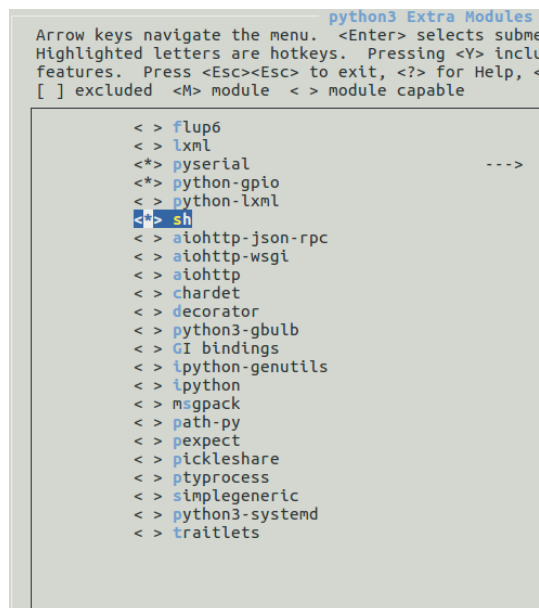


Abb. 37: PTXdist – „Python 3 Extra Modules“

- ▶ ipkg-Pakete über den folgenden Befehl erzeugen:
`ptxdist go`
- ⇒ Wenn keine Fehler aufgetreten sind, ist das Paket mit dem sh-Modul unter „platform-tben-lx-linux/packages/“ zu finden:

```
$ ls platform-tben-lx-linux/packages/  
...  
python3_3.5.0_armhf.ipk  
sh_1.12.14_armhf.ipk  
...
```
- ▶ ipkg-File auf das TBEN-Gerät kopieren (z. B. mit scp):
`scp ~/turck/TBEN-Lx-4RFID-8DXP-LNX/platform-tben-lx-linux/packages/sh_1.12.14_armhf.ipk root@Target-IP:/directory/of/your/choice/`
- ▶ Auf dem TBEN-Gerät einloggen, um das Paket „sh_1.12.14_armhf.ipk“ zu installieren.
- ▶ ipkg-Manager aufrufen, um das Python-Modul zu installieren:
`ipkg -force-depends install sh_1.12.14_armhf.ipk`
- ⇒ Das Modul ist im Python Interpreter verfügbar.

7.11 REST-API nutzen

Der TN-UHF-...-LNX UHF-Reader beinhaltet eine integrierte REST-API, mit der RFID-Funktionen direkt ausgeführt werden können. Die hier gezeigten GET- bzw. POST-Befehle werden sofern nötig mit einem REQUEST-Body im JSON-Datenformat abgebildet.



HINWEIS

Die REST-API ist ab Firmware Version 1.2.5.0 enthalten. Zur einfacheren Integration der REST-API steht eine .YAML-Datei im Downloadbereich unter www.turck.com am Produkt zur Verfügung.

7.11.1 REST-API im Webserver aktivieren

Verwendete Hardware:

- TN-UHF-Q180L300-EU-LNX mit IP Adresse 192.168.1.20



HINWEIS

Standardmäßig ist die REST-API deaktiviert. Nach Aktivierung ist ein Neustart erforderlich, damit die Änderungen aktiv werden.

- ▶ Webserver des Geräts öffnen.
- ▶ **Username** und **Password** eingeben.
- ▶ **Login** klicken.
- ⇒ Nach dem Anmelden wird die Startseite mit den Geräteinformationen angezeigt.
- ▶ In der Navigationsleiste am linken Bildrand **Parameter** anklicken.
- ▶ Im Punkt **Aktiviere REST-API ja** auswählen.
- ▶ **Schreiben** anklicken.
- ▶ Gerät neu starten.

The screenshot shows the web interface of the Turck device. On the left, a sidebar menu under 'DEVICE' has 'Parameter' selected. The main content area is titled 'Gerät Einstellungen'. Under 'Einstellungen', the 'Aktiviere REST-API' option is set to 'ja' (highlighted with a red box). To the right of this are buttons for 'DEAKTIVIERE WEBSERVER', 'RESET AUSFÜHREN', and 'SETZE ZEIT VOM HOST'. Below these are settings for 'Datum und Uhrzeit', including 'Aktuelle Uhrzeit (UTC)' set to '01.01.2021 00:41:22' and 'Zeitzone' set to 'UTC'.

Abb. 38: Webserver – REST-API aktivieren

⇒ Gerät startet neu, REST-API ist aktiviert.

7.11.2 Übersicht der Befehle

Befehl	Funktion
Process an inventory	Führt einen Inventory Befehl zum Lesen einzelner oder auch mehrerer Datenträger (Pulkerfassung) aus.
Read data from transponder	Liest Daten aus einem beliebigen Speicherbereich eines UHF-RFID-Datenträgers aus.
Write data to transponder	Schreibt Daten in einen beliebigen Speicherbereich eines UHF-RFID-Datenträgers.
Write data to transponder and verify	Schreibt Daten in einen beliebigen Speicherbereich eines UHF-RFID-Datenträgers und verifiziert anschließend den Schreibbefehl mit einem automatischen Lesebefehl.
Process a (universal RFID interface) request	Führt einen beliebigen Befehl des RFID-U-Interfaces aus.

7.11.3 Befehl: Process and inventory

URL

GET <http://192.168.24.79:32760/api/v1/rfid/inventory?timeout=0>

Parameter	Bedeutung
Timeout	Zeit in ms, in der Befehl ausgeführt werden soll. Wird ein Befehl nicht innerhalb der angegebenen Zeit ausgeführt, gibt das Gerät eine Fehlermeldung aus. 0: kein Timeout, Befehl bleibt aktiv, bis der erste Datenträger gelesen wurde. 1: Befehl wird einmal ausgeführt (sofern sich bereits ein Datenträger im Erfassungsbereich befindet). >1...65535: Zeit in ms; UHF Inventory: Befehl bleibt für die gesamte angegebene Zeit aktiv.

Übersicht Eingangsdaten

Response	
Schleifenzähler	siehe Beschreibung der Eingangsdaten
Antwortcode	0x0001 (hex.), 1 (dez.)
Länge	Länge der gelesenen Daten
Fehlercode	siehe Beschreibung der Eingangsdaten
Datenträger im Erfassungsbereich	siehe Beschreibung der Eingangsdaten
Daten (Bytes) verfügbar	siehe Beschreibung der Eingangsdaten
Datenträger-Zähler	ansteigend
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Eingangsdaten
Lesedaten, Byte 0...n	siehe Beispiel: UHF-Lesedaten

Datenformat in UHF-Anwendungen

Die UHF-Lesedaten sind durch einen Header formatiert. Der Header ist wie folgt aufgebaut:

Typ	Name	Bedeutung
uint8_t	Größe	Datengröße
uint8_t	Blocktyp	1: EPC etc. andere Werte: reserviert
uint8_t	Daten [Größe]	EPC und Lesedaten

Die Größe von EPC/RSSI etc. ist abhängig von den Reader-Einstellungen.

Beispiel: UHF-Lesedaten (Header und EPC, Gruppierung mit RSSI aktiviert)

Typ	Name	Bedeutung
uint8_t	Größe	16
uint8_t	Blocktyp	1
uint8_t	Daten [20]	uint8_t EPC [12] uint16_t RSSI [2] uint16_t Nummer der Antenne (LSB → MSB) [2] uint16_t Anzahl der Lesevorgänge (LSB → MSB) [2]

Byte	Inhalt	Bedeutung
0	Datengröße (EPC + Anzahl Lesevorgänge)	2 Byte Header
1	UHF-Speicherbereich	
3...13	EPC	12 Byte EPC
14	LSB	2 Byte RSSI
15	MSB	
16	LSB	2 Byte Nummer der Antenne:
17	MSB	<ul style="list-style-type: none"> ■ 0: RHCP ■ 1: LHCP ■ 2: Horizontal ■ 3: Vertikal ■ 4: Extern 1 ■ 5: Extern 2 ■ 6: Extern 3 ■ 7: Extern 4
18	LSB	2 Byte Anzahl Lesevorgänge
19	MSB	

RSSI-Wert auslesen

Der RSSI-Wert wird binär codiert in 2 Bytes ausgegeben und entspricht dem Zweierkomplement des ausgegebenen Binärcodes. Auf ein Signed Integer gemappt ergeben die ausgegebenen 2 Bytes das Zehnfache des aktuellen RSSI-Werts. Ein Beispiel zum Auslesen des RSSI-Werts entnehmen Sie folgender Tabelle:

MSB...LSB (dezimal)	MSB...LSB (binär)	Zweierkomplement	RSSI (dBm)
252 253	11111100 11111101	-771	-77,1

7.11.4 Befehl: Read

URL

POST http://192.168.24.79:32760/api/v1/rfid/read

Request Body

```
{
  "length": "<integer>",
  "memoryArea": "<integer>",
  "address": "<integer>",
  "sizeofUid": "<integer>",
  "timeout": "<integer>",
  "data": [
    "<integer>",
    "<integer>"
  ]
}
```

Übersicht Ausgangsdaten

Request	
Speicherbereich (DOM)	0: Kill-Passwort 1: EPC 2: TID 3: USER-Bereich 4: Access-Passwort 5: PC (definiert die Antwortlänge des EPC)
Länge EPC	Angabe der EPC-Größe in Bytes, wenn ein bestimmter Datenträger gelesen werden soll. Der EPC muss in den Schreibdaten definiert werden (Startbyte: 0). Die Funktion der Länge des EPC ist abhängig vom verwendeten Befehl. 0: Keine Angabe eines EPC zur Ausführung des Befehls. Dabei darf sich nur ein Datenträger im Erfassungsbereich des Schreib-Lese-Kopf befinden. >0: EPC-Länge des Datenträgers, der gelesen werden soll, wenn in den Schreibdaten ein EPC vorhanden ist.
Startadresse	Startadresse des Speicherbereichs auf dem Datenträger, der gelesen werden soll (Angabe in Bytes)
Länge	Länge der zu lesenden Daten in Bytes
Befehls-Time-out	siehe Beschreibung der Ausgangsdaten
Schreibdaten, Byte 0...(Größe des EPC-1)	EPC des Datenträgers, der gelesen werden soll
Schreibdaten, Byte (Größe des EPC)...127	nicht erforderlich

Response

```
{  
  "data": [  
    "<integer>",  
    "<integer>"  
  ]  
}
```

7.11.5 Befehl: Write

URL

POST http://192.168.24.79:32760/api/v1/rfid/write

Request Body

```
{
  "data": [
    "<integer>",
    "<integer>"
  ],
  "memoryArea": "<integer>",
  "address": "<integer>",
  "sizeofUid": "<integer>",
  "timeout": "<integer>"
}
```

Übersicht Ausgangsdaten

Request	
Speicherbereich (DOM)	0: Kill-Passwort 1: EPC 2: TID 3: USER-Bereich 4: Access-Passwort 5: PC (definiert die Antwortlänge des EPC)
Länge EPC	Angabe der EPC-Größe in Bytes, wenn ein bestimmter Datenträger beschrieben werden soll. Der EPC muss in den Schreibdaten definiert werden (Startbyte: 0). Die Funktion der Länge des EPC ist abhängig vom verwendeten Befehl. 0: Keine Angabe eines EPC zur Ausführung des Befehls. Dabei darf sich nur ein Datenträger im Erfassungsbereich des Schreib-Lese-Kopf befinden. >0: EPC-Länge des Datenträgers, der beschrieben werden soll, wenn in den Schreibdaten ein EPC vorhanden ist.
Startadresse	Startadresse des Speicherbereichs auf dem Datenträger, der beschrieben werden soll (Angabe in Bytes)
Länge	Länge der zu schreibenden Daten in Bytes
Befehls-Time-out	siehe Beschreibung der Ausgangsdaten
Schreibdaten, Byte 0...(Größe des EPC-1)	EPC des Datenträgers, der beschrieben werden soll
Schreibdaten, Byte (Größe des EPC)...127	Schreibdaten

Response

```
{  
  "data": [ ],  
  "errorCode": 0,  
  "length": 0,  
  "status": {  
    "code": 4,  
    "tagCounter": 1  
  }  
}
```


7.11.6 Befehl: Write and verify

URL

POST <http://192.168.24.79:32760/api/v1/rfid/writeandverify>

Request Body

```
{
  "data": [
    "<integer>",
    "<integer>"
  ],
  "memoryArea": "<integer>",
  "address": "<integer>",
  "sizeofUid": "<integer>",
  "timeout": "<integer>"
}
```

Übersicht Ausgangsdaten

Request	
Speicherbereich (DOM)	0: Kill-Passwort 1: EPC 2: TID 3: USER-Bereich 4: Access-Passwort 5: PC (definiert die Antwortlänge des EPC)
Länge EPC	Angabe der EPC-Größe in Bytes, wenn ein bestimmter Datenträger beschrieben werden soll. Der EPC muss in den Schreibdaten definiert werden (Startbyte: 0). Die Funktion der Länge des EPC ist abhängig vom verwendeten Befehl. 0: Keine Angabe eines EPC zur Ausführung des Befehls. Dabei darf sich nur ein Datenträger im Erfassungsbereich des Schreib-Lese-Kopf befinden. >0: EPC-Länge des Datenträgers, der beschrieben werden soll, wenn in den Schreibdaten ein EPC vorhanden ist.
Startadresse	Startadresse des Speicherbereichs auf dem Datenträger, der beschrieben werden soll (Angabe in Bytes)
Länge	Länge der zu schreibenden Daten in Bytes
Befehls-Time-out	siehe Beschreibung der Ausgangsdaten
Schreibdaten, Byte 0...(Größe des EPC-1)	EPC des Datenträgers, der beschrieben werden soll
Schreibdaten, Byte (Größe des EPC)...127	Schreibdaten

Response

```
{  
  "data": [ ],  
  "errorCode": 0,  
  "length": 0,  
  "status": {  
    "code": 8,  
    "tagCounter": 1  
  }  
}
```

7.11.7 Befehl: Process a (universal RFID interface) request

URL

POST <http://192.168.24.79:32760/api/v1/rfid/command>

Request Body

```
{  
  "commandCode": "<integer>",  
  "memoryArea": "<integer>",  
  "length": "<integer>",  
  "address": "<integer>",  
  "sizeOfUid": "<integer>",  
  "timeout": "<integer>",  
  "data": [  
    "<integer>",  
    "<integer>"  
  ]  
}
```

7.12 RFID-Kanäle – Übersicht der Befehle

RFID-Befehle werden über den Befehlscode in den Prozess-Ausgangsdaten eines RFID-Kanals angestoßen. Die Befehle lassen sich mit oder ohne Schleifenzähler-Funktion ausführen. Der Schleifenzähler muss für jeden neuen Befehl einzeln gesetzt werden.



HINWEIS

Nach dem Ausführen von Befehlen ohne Schleifenzähler-Funktion muss das Gerät in den Leerlauf-Zustand zurückgesetzt werden, bevor ein neuer Befehl gesendet wird.

- Nach ausgeführtem Befehl einen Leerlauf-Befehl an das Gerät senden.

Befehl	Befehlscode		möglich für	
	hex.	dez.	UHF Kompakt	UHF Erweitert
Leerlauf	0x0000	0	x	x
Inventory	0x0001	1	x	x
Schnelles Inventory	0x2001	8193	x	x
Lesen	0x0002	2	x	x
Schnelles Lesen	0x2002	8194	x	x
Schreiben	0x0004	4	x	x
Schnelles Schreiben	0x2004	8196	x	x
Schreiben mit Validierung	0x0008	8	x	x
Continuous Mode	0x0010	16	–	x
Daten aus dem Puffer lesen (Continuous Mode)	0x0011	17	max. 128 Byte	x
Daten aus dem Puffer lesen mit schneller Befehlsverarbeitung (Continuous Mode)	0x2011	8209	max. 128 Byte	x
Continuous Presence Sensing Mode	0x0020	32	–	x
Continuous (Presence Sensing) Mode beenden	0x0012	18	–	x
Schreib-Lese-Kopf-Identifikation	0x0041	65	x	x
Direkter Schreib-Lese-Kopf-Befehl	0x0060	96	x	x
Direkter Schreib-Lese-Kopf-Befehl mit schneller Befehlsverarbeitung	0x2060	8288	x	x
Datenträger-Passwort setzen	0x0102	258	x	x
Datenträger-Passwort setzen mit schneller Befehlsverarbeitung	0x2102	8450	x	x
Schreib-Lese-Kopf-Passwort setzen	0x0100	256	x	x
Schreib-Lese-Kopf-Passwort zurücksetzen	0x0101	257	x	x
Datenträger-Schutz setzen	0x0103	259	x	x
Datenträger-Schutz setzen mit schneller Befehlsverarbeitung	0x2103	8451	x	x
Permanente Sperre setzen (Lock)	0x0105	261	x	x
Permanente Sperre setzen (Lock) mit schneller Befehlsverarbeitung	0x2105	8453	x	x
Datenträger-Info	0x0050	80	x	x

Befehl	Befehlscode		möglich für	
	hex.	dez.	UHF Kompakt	UHF Erweitert
Datenträger-Info mit schneller Befehlsverarbeitung	0x2050	8272	x	x
UHF-Datenträger unwiderruflich deaktivieren (Kill)	0x0200	512	x	x
UHF-Datenträger unwiderruflich deaktivieren (Kill) mit schneller Befehlsverarbeitung	0x2200	8704	x	x
Einstellungen UHF-Schreib-Lese-Kopf wiederherstellen	0x1000	4096	x	x
Backup der Einstellungen des UHF-Schreib-Lese-Kopfs	0x1001	4097	x	x
Fehler/Status UHF-Schreib-Lese-Kopf lesen	0x0042	66	x	x
Reset	0x8000	32768	x	x

7.12.1 Befehl: Leerlauf

Über den Befehl **Leerlauf** wird das Interface in den Leerlauf versetzt. Die Befehlsausführung wird abgebrochen. Der EPC wird angezeigt, wenn der Reader im Presence Sensing Mode über TAS oder den Webserver parametrisiert ist.

Übersicht Ausgangsdaten

Request	
Schleifenzähler	nicht erforderlich
Befehlscode	0x0000 (hex.), 0 (dez.)
Schreib-Lese-Kopf-Adresse	nicht erforderlich
Länge EPC	nicht erforderlich
Startadresse	nicht erforderlich
Länge	nicht erforderlich
Befehls-Time-out	nicht erforderlich
Schreib-Fragment-Nr.	nicht erforderlich
Lese-Fragment-Nr.	nicht erforderlich
Schreibdaten	nicht erforderlich

Übersicht Eingangsdaten

Response	
Schleifenzähler	siehe Beschreibung der Eingangsdaten
Antwortcode	0x0000 (hex.), 0 (dez.)
Länge	EPC-Länge des Datenträgers im Erfassungsbereich
Fehlercode	siehe Beschreibung der Eingangsdaten
Datenträger im Erfassungsbereich	siehe Beschreibung der Eingangsdaten
Daten (Bytes) verfügbar	siehe Beschreibung der Eingangsdaten
Datenträger-Zähler	siehe Beschreibung der Eingangsdaten
Schreib-Fragment-Nr.	Größe der Fragmente
Lese-Fragment-Nr.	Größe der Fragmente
Lesedaten, Byte 0...n	EPC des Datenträgers im Erfassungsbereich

7.12.2 Befehl: Inventory

Über den Befehl **Inventory** sucht der Reader nach Datenträgern im Erfassungsbereich und liest den EPC oder sofern im UHF-Reader aktiviert den RSSI der Datenträger aus. Der Inventory-Befehl kann im Single-Tag-Modus und im Multitag-Modus ausgeführt werden.



HINWEIS

Der Befehlscode für die schnelle Bearbeitung mit dem Schleifenzähler ist 0x2001 (hex.) bzw. 8193 (dez.).

Übersicht Ausgangsdaten

Request	
Schleifenzähler	siehe Beschreibung der Ausgangsdaten
Befehlscode	0x0001 (hex.), 1 (dez.)
Schreib-Lese-Kopf-Adresse	siehe Beschreibung der Ausgangsdaten
Länge EPC	nicht erforderlich
Startadresse	1: Gruppierung der EPCs aktiv 0: Gruppierung der EPCs inaktiv
Länge	0: Die tatsächliche Länge (Bytes) des übertragenen EPC bei einem Inventory übertragen. > 0 : EPC wird vollständig ausgegeben.
Befehls-Time-out	siehe Beschreibung der Ausgangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Ausgangsdaten
Schreibdaten	nicht erforderlich

Übersicht Eingangsdaten

Response	
Schleifenzähler	siehe Beschreibung der Eingangsdaten
Antwortcode	0x0001 (hex.), 1 (dez.)
Länge	Länge der gelesenen Daten
Fehlercode	siehe Beschreibung der Eingangsdaten
Datenträger im Erfassungsbereich	siehe Beschreibung der Eingangsdaten
Daten (Bytes) verfügbar	siehe Beschreibung der Eingangsdaten
Datenträger-Zähler	ansteigend
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Eingangsdaten
Lesedaten, Byte 0...n	siehe Beispiel: UHF-Lesedaten

Datenformat in UHF-Anwendungen

Die UHF-Lesedaten sind durch einen Header formatiert. Der Header ist wie folgt aufgebaut:

Typ	Name	Bedeutung
uint8_t	Größe	Datengröße
uint8_t	Blocktyp	1: EPC etc. andere Werte: reserviert
uint8_t	Daten [Größe]	EPC und Lesedaten

Die Größe von EPC/RSSI etc. ist abhängig von den Reader-Einstellungen.

RSSI-Wert auslesen

Der RSSI-Wert wird binär codiert in 2 Bytes ausgegeben und entspricht dem Zweierkomplement des ausgegebenen Binärcodes. Auf ein Signed Integer gemappt ergeben die ausgegebenen 2 Bytes das Zehnfache des aktuellen RSSI-Werts. Ein Beispiel zum Auslesen des RSSI-Werts entnehmen Sie folgender Tabelle:

MSB...LSB (dezimal)	MSB...LSB (binär)	Zweierkomplement	RSSI (dBm)
252 253	11111100 11111101	-771	-77,1

Beispiel: UHF-Lesedaten (Header und EPC, Gruppierung mit RSSI aktiviert)

Typ	Name	Bedeutung
uint8_t	Größe	16
uint8_t	Blocktyp	1
uint8_t	Daten [20]	uint8_t EPC [12] uint16_t RSSI [2] uint16_t Nummer der Antenne (LSB → MSB) [2] uint16_t Anzahl der Lesevorgänge (LSB → MSB) [2]

Byte	Inhalt	Bedeutung
0	Datengröße (EPC + Anzahl Lesevorgänge)	2 Byte Header
1	UHF-Speicherbereich	
3...13	EPC	12 Byte EPC
14	LSB	2 Byte RSSI
15	MSB	
16	LSB	2 Byte Nummer der Antenne:
17	MSB	<ul style="list-style-type: none"> ■ 0: RHCP ■ 1: LHCP ■ 2: Horizontal ■ 3: Vertikal ■ 4: Extern 1 ■ 5: Extern 2 ■ 6: Extern 3 ■ 7: Extern 4
18	LSB	2 Byte Anzahl Lesevorgänge
19	MSB	

7.12.3 Befehl: Lesen

Über den Befehl **Lesen** liest der Reader Daten von Datenträgern im Erfassungsbereich. Standardmäßig werden bei einem Lesevorgang 128 Bytes übertragen. Größere Datenmengen können in Fragmenten übertragen werden. Wird ein bestimmter EPC angegeben, liest der Reader ausschließlich die entsprechenden Datenträger. Alle anderen Datenträger im Erfassungsbereich werden in diesem Fall ignoriert.



HINWEIS

Der Befehlscode für die schnelle Bearbeitung mit dem Schleifenzähler ist 0x2002 (hex.) bzw. 8194 (dez.).

Übersicht Ausgangsdaten

Request	
Schleifenzähler	siehe Beschreibung der Ausgangsdaten
Befehlscode	0x0002 (hex.), 2 (dez.)
Speicherbereich	siehe Beschreibung der Ausgangsdaten
Schreib-Lese-Kopf-Adresse	siehe Beschreibung der Ausgangsdaten
Länge EPC	Angabe der EPC-Größe in Bytes, wenn ein bestimmter Datenträger gelesen werden soll. Der EPC muss in den Schreibdaten definiert werden (Startbyte: 0). Die Funktion der EPC-Länge ist abhängig vom verwendeten Befehl. 0: Keine Angabe eines EPC zur Ausführung des Befehls. Dabei darf sich nur ein Datenträger im Erfassungsbereich des Schreib-Lese-Kopfs befinden. > 0: EPC-Länge des Datenträgers, der gelesen werden soll, wenn in den Schreibdaten ein EPC vorhanden ist.
Startadresse	Startadresse des Speicherbereichs auf dem Datenträger, der gelesen werden soll (Angabe in Bytes)
Länge	Länge der zu lesenden Daten in Bytes
Befehls-Time-out	siehe Beschreibung der Ausgangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Ausgangsdaten
Schreibdaten, Byte 0...(EPC-Größe-1)	EPC des Datenträgers, der gelesen werden soll
Schreibdaten, Byte (EPC-Größe)...127	nicht erforderlich

Übersicht Eingangsdaten

Response	
Schleifenzähler	siehe Beschreibung der Eingangsdaten
Antwortcode	0x0002 (hex.), 2 (dez.)
Länge	Länge der gelesenen Daten
Fehlercode	siehe Beschreibung der Eingangsdaten
Datenträger im Erfassungsbereich	siehe Beschreibung der Eingangsdaten
Daten (Bytes) verfügbar	steigt während der Befehlsausführung an
Datenträger-Zähler	siehe Beschreibung der Eingangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Eingangsdaten
Lesedaten, Byte 0...n	gelesene Daten

7.12.4 Befehl: Schreiben

Über den Befehl **Schreiben** schreibt der Reader Daten auf Datenträger im Erfassungsbereich. Standardmäßig werden bei einem Schreibvorgang 128 Bytes übertragen. Größere Datenmengen können in Fragmenten übertragen werden. Wird ein bestimmter EPC angegeben, schreibt der Reader ausschließlich die entsprechenden Datenträger. Alle anderen Datenträger im Erfassungsbereich werden in diesem Fall ignoriert.



HINWEIS

► Bei Multitag-Anwendungen EPC des zu beschreibenden Datenträgers angeben.



HINWEIS

Der Befehlscode für die schnelle Bearbeitung mit dem Schleifenzähler ist 0x2004 (hex.) bzw. 8196 (dez.).

Übersicht Ausgangsdaten

Request	
Schleifenzähler	siehe Beschreibung der Ausgangsdaten
Befehlscode	0x0004 (hex.), 4 (dez.)
Speicherbereich	siehe Beschreibung der Ausgangsdaten
Schreib-Lese-Kopf-Adresse	siehe Beschreibung der Ausgangsdaten
Länge EPC	Angabe der EPC-Größe in Bytes, wenn ein bestimmter Datenträger beschrieben werden soll. Der EPC muss in den Schreibdaten definiert werden (Startbyte: 0). Die Funktion der EPC-Länge ist abhängig vom verwendeten Befehl. 0: Keine Angabe eines EPC zur Ausführung des Befehls. Dabei darf sich nur ein Datenträger im Erfassungsbereich des Schreib-Lese-Kopfs befinden. > 0: EPC-Länge des Datenträgers, der beschrieben werden soll, wenn in den Schreibdaten ein EPC vorhanden ist.
Startadresse	Startadresse des Speicherbereichs auf dem Datenträger, der beschrieben werden soll (Angabe in Bytes)
Länge	Länge der zu schreibenden Daten in Bytes
Befehls-Time-out	siehe Beschreibung der Ausgangsdaten
Schreib-Fragment-Nr.	1: Fragmentierung nutzen 0: Fragmentierung nicht nutzen
Lese-Fragment-Nr.	0
Schreibdaten, Byte 0...(Größe des EPC-1)	EPC des Datenträgers, der beschrieben werden soll
Schreibdaten, Byte (Größe des EPC)... 127	Schreibdaten

Übersicht Eingangsdaten

Response	
Schleifenzähler	siehe Beschreibung der Eingangsdaten
Antwortcode	0x0004 (hex.), 4 (dez.)
Länge	Länge der gelesenen Daten
Fehlercode	siehe Beschreibung der Eingangsdaten
Datenträger im Erfassungsbereich	siehe Beschreibung der Eingangsdaten
Daten (Bytes) verfügbar	steigt während der Befehlsausführung an
Datenträger-Zähler	siehe Beschreibung der Eingangsdaten
Schreib-Fragment-Nr.	siehe Beschreibung der Eingangsdaten
Lese-Fragment-Nr.	0
Lesedaten, Byte 0...127	nicht erforderlich

7.12.5 Befehl: Schreiben mit Validierung

Über den Befehl **Schreiben mit Validierung** wird eine vom Anwender definierte Anzahl Bytes geschrieben. Die geschriebenen Daten werden zusätzlich zurück an das Interface geschickt und validiert. Beim Schreiben werden standardmäßig bis zu 128 Bytes übertragen. Größere Datenmengen können in Fragmenten übertragen werden. Die geschriebenen Daten werden ausschließlich im Interface validiert und nicht an die Steuerung zurückgeschickt. Schlägt die Validierung fehl, wird eine Fehlermeldung ausgegeben. Wird der Befehl ohne Fehlermeldung abgearbeitet, wurden die Daten erfolgreich validiert.



HINWEIS

► Bei Multitag-Anwendungen EPC des zu beschreibenden Datenträgers angeben.



HINWEIS

Der Befehlscode für die schnelle Bearbeitung mit dem Schleifenzähler ist 0x2008 (hex.) bzw. 8200 (dez.).

Übersicht Ausgangsdaten

Request	
Schleifenzähler	siehe Beschreibung der Ausgangsdaten
Befehlscode	0x0008 (hex.), 8 (dez.)
Speicherbereich	siehe Beschreibung der Ausgangsdaten
Schreib-Lese-Kopf-Adresse	siehe Beschreibung der Ausgangsdaten
Länge EPC	Angabe der EPC-Größe in Bytes, wenn ein bestimmter Datenträger beschrieben werden soll. Der EPC muss in den Schreibdaten definiert werden (Startbyte: 0). Die Funktion der Länge des EPC ist abhängig vom verwendeten Befehl. 0: Keine Angabe einer EPC zur Ausführung des Befehls. Dabei darf sich nur ein Datenträger im Erfassungsbereich des Readers befinden. > 0: EPC-Länge des Datenträgers, der beschrieben werden soll, wenn in den Schreibdaten ein EPC vorhanden ist.
Startadresse	Startadresse des Speicherbereichs auf dem Datenträger, der beschrieben werden soll (Angabe in Bytes)
Länge	Länge der zu schreibenden Daten in Bytes
Befehls-Time-out	siehe Beschreibung der Ausgangsdaten
Schreib-Fragment-Nr.	1: Fragmentierung nutzen 0: Fragmentierung nicht nutzen
Lese-Fragment-Nr.	0
Schreibdaten, Byte 0...(EPC-Größe-1)	optional: EPC des Datenträgers, der beschrieben werden soll
Schreibdaten, Byte (EPC-Größe)...127	Schreibdaten

Übersicht Eingangsdaten

Response	
Schleifenzähler	siehe Beschreibung der Eingangsdaten
Antwortcode	0x0008 (hex.), 8 (dez.)
Länge	Länge der gelesenen Daten
Fehlercode	siehe Beschreibung der Eingangsdaten
Datenträger im Erfassungsbereich	siehe Beschreibung der Eingangsdaten
Daten (Bytes) verfügbar	steigt während der Befehlsausführung an
Datenträger-Zähler	siehe Beschreibung der Eingangsdaten
Schreib-Fragment-Nr.	siehe Beschreibung der Eingangsdaten
Lese-Fragment-Nr.	0
Lesedaten, Byte 0...MIN(127, eingestellte Länge-1)	nicht erforderlich

7.12.6 Befehl: Continuous Mode

Im Continuous Mode wird ein benutzerdefinierter Befehl an den Reader gesendet und im Reader gespeichert. Die Befehle Schreiben, Lesen und Inventory sind im Continuous Mode ausführbar. Die Parameter für den Continuous Mode müssen direkt im Reader eingestellt werden.

Der Befehl wird so lange kontinuierlich ausgeführt, bis der Anwender den Continuous Mode beendet. Der Continuous Mode lässt sich durch das Ausführen eines Reset-Befehls beenden.



HINWEIS

Der Reset-Befehl setzt alle gelesenen Daten zurück.

Reader im Continuous Mode senden alle befehlspezifischen Daten an das Interface. Die Daten werden im FIFO-Speicher des Interfaces hinterlegt und können über den Befehl **Get Data from FIFO** durch die Steuerung abgefragt werden.

Befehle im Continuous Mode werden ausgelöst, wenn der Reader einen Datenträger erkennt. Befindet sich beim Starten des Continuous Mode ein Datenträger im Erfassungsbereich des Readers, wird der im Continuous Mode gesendete Befehl erst für den nächsten Datenträger ausgeführt.



HINWEIS

Im Continuous Mode wird das Signal **Datenträger im Erfassungsbereich** nicht aktualisiert.

Startadresse und Länge können während der Ausführung des Continuous Mode nicht geändert werden.

Nach einem Neustart des Continuous Mode werden alle Daten des bereits laufenden Continuous Mode gelöscht.

Übersicht Ausgangsdaten

Request	
Schleifenzähler	siehe Beschreibung der Ausgangsdaten
Befehlscode	0x0010 (hex.), 16 (dez.)
Schreib-Lese-Kopf-Adresse	siehe Beschreibung der Ausgangsdaten
Länge EPC	nicht erforderlich
Startadresse	1: Gruppierung der EPCs aktiv (nur UHF Inventory) 0: Gruppierung der EPCs inaktiv (nur UHF Inventory) >1: nicht definiert
Länge	nicht erforderlich
Befehls-Time-out	nicht erforderlich
Schreib-Fragment-Nr.	0
Lesen-Fragment-Nr.	siehe Beschreibung der Ausgangsdaten
Schreibdaten	nicht erforderlich

Übersicht Eingangsdaten

Response	
Schleifenzähler	siehe Beschreibung der Eingangsdaten
Antwortcode	0x0010 (hex.), 16 (dez.)
Länge	0
Fehlercode	siehe Beschreibung der Eingangsdaten
Datenträger im Erfassungsbereich	siehe Beschreibung der Eingangsdaten
Daten (Bytes) verfügbar	steigt während der Befehlsausführung an
Datenträger-Zähler	steigt mit jedem gelesenen oder geschriebenen EPC
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Eingangsdaten
Lesedaten	siehe Beschreibung der Eingangsdaten

7.12.7 Befehl: Daten aus dem Puffer lesen (Continuous Mode)



HINWEIS

Der Befehlscode für die schnelle Bearbeitung mit dem Schleifenzähler ist 0x2011 (hex.) bzw. 8209 (dez.).

Über den Befehl **Daten aus dem Puffer lesen (Continuous Mode)** können im Interface gespeicherte Daten an die Steuerung weitergegeben werden. Der Befehl ist erforderlich, um im Continuous Mode oder im Continuous Presence Sensing Mode gelesene Daten an die Steuerung zu übertragen. Die Daten werden in Fragmenten von bis zu 128 Bytes an die Steuerung übertragen. Die Größe der Fragmente lässt sich vom Anwender einstellen. Ein EPC wird nicht durch Fragmentgrenzen geteilt. Passt ein EPC nicht vollständig in ein Fragment, wird er automatisch in das nächste Fragment geschoben.



HINWEIS

Der Befehl **Daten aus dem Puffer lesen** beendet nicht den Continuous Mode.

Übersicht Ausgangsdaten

Request	
Schleifenzähler	siehe Beschreibung der Ausgangsdaten
Befehlscode	0x0011 (hex.), 17 (dez.)
Schreib-Lese-Kopf-Adresse	siehe Beschreibung der Ausgangsdaten
Länge EPC	nicht erforderlich
Startadresse	nicht erforderlich
Länge	max. Länge der Daten, die vom Gerät gelesen werden sollen (\leq Größe der Daten, die das Gerät tatsächlich gespeichert hat), Angabe in Bytes
Befehls-Time-out	siehe Beschreibung der Ausgangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Ausgangsdaten
Schreibdaten	nicht erforderlich

Übersicht Eingangsdaten

Response	
Schleifenzähler	siehe Beschreibung der Eingangsdaten
Antwortcode	0x0011 (hex.), 17 (dez.)
Länge	Länge der gelesenen Daten. Die Daten werden in vollständigen Blöcken angegeben.
Fehlercode	siehe Beschreibung der Eingangsdaten
Datenträger im Erfassungsbereich	siehe Beschreibung der Eingangsdaten
Daten (Bytes) verfügbar	wird nach der Befehlsausführung automatisch verringert
Datenträger-Zähler	siehe Beschreibung der Eingangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Eingangsdaten
Lesedaten	Lesedaten

Datenformat in UHF-Anwendungen

Die UHF-Lesedaten sind durch einen Header formatiert. Der Header ist wie folgt aufgebaut:

Typ	Name	Bedeutung
uint8_t	Größe	Datengröße
uint8_t	Blocktyp	1: EPC etc. andere Werte: reserviert
uint8_t	Daten [Größe]	EPC und Lesedaten

Die Größe von EPC/RSSI etc. ist abhängig von den Reader-Einstellungen.

7.12.8 Befehl: UHF Continuous Presence Sensing Mode

Im Continuous Presence Sensing Mode wird ein benutzerdefinierter Befehl (Schreiben, Lesen, Inventory) an den UHF-Reader gesendet und im Reader gespeichert. Die Reader werden im Continuous Presence Sensing Mode automatisch eingeschaltet, sobald sich ein Datenträger im Erfassungsbereich befindet. Die Dauer des Abfrageintervalls und die Einschaltdauer können in den Einstellungen des UHF-Readers angepasst werden. Der Befehl wird so lange kontinuierlich ausgeführt, bis der Anwender den Continuous Presence Sensing Mode durch das Ausführen eines Reset-Befehls beendet.



HINWEIS

Der Reset-Befehl setzt alle gelesenen Daten zurück.

Reader im Continuous Presence Sensing Mode senden alle befehlspezifischen Daten an das Interface. Die Daten werden im Puffer des Interfaces hinterlegt und können über den Befehl **Daten aus dem Puffer lesen** durch die Steuerung abgefragt werden. Im Continuous Presence Sensing Mode wird das Signal **Datenträger im Erfassungsbereich** nicht dauerhaft aktualisiert.

Übersicht Ausgangsdaten

Request	
Schleifenzähler	siehe Beschreibung der Ausgangsdaten
Befehlscode	0x0020 (hex.), 32 (dez.)
Schreib-Lese-Kopf-Adresse	siehe Beschreibung der Ausgangsdaten
Länge EPC	nicht erforderlich
Startadresse	0: Gruppierung inaktiv 1: Gruppierung aktiv >1: nicht definiert
Länge	nicht erforderlich
Befehls-Time-out	nicht erforderlich
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Ausgangsdaten
Schreibdaten	nicht erforderlich

Übersicht Eingangsdaten

Response	
Schleifenzähler	siehe Beschreibung der Eingangsdaten
Antwortcode	0x0020 (hex.), 32 (dez.)
Länge	nicht erforderlich
Fehlercode	siehe Beschreibung der Eingangsdaten
Datenträger im Erfassungsbereich	siehe Beschreibung der Eingangsdaten
Daten (Bytes) verfügbar	steigt während der Befehlsausführung an
Datenträger-Zähler	steigt mit jedem gelesenen oder geschriebenen EPC
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Eingangsdaten
Lesedaten	siehe Beschreibung der Eingangsdaten

7.12.9 Befehl: Continuous (Presence Sensing) Mode beenden

Über den Befehl **Continuous (Presence Sensing) Mode beenden** können Continuous Mode und Presence Sensing Mode gestoppt werden. Die Daten im Puffer des Interfaces werden nach der Befehlsausführung nicht gelöscht und können über den Befehl **Daten aus dem Puffer lesen** weiterhin abgerufen werden.

Übersicht Ausgangsdaten

Request	
Schleifenzähler	siehe Beschreibung der Ausgangsdaten
Befehlscode	0x0012 (hex.), 18 (dez.)
Schreib-Lese-Kopf-Adresse	nicht erforderlich
Länge EPC	nicht erforderlich
Startadresse	nicht erforderlich
Länge	nicht erforderlich
Befehls-Time-out	siehe Beschreibung der Ausgangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Ausgangsdaten
Schreibdaten	nicht erforderlich

Übersicht Eingangsdaten

Response	
Schleifenzähler	siehe Beschreibung der Eingangsdaten
Antwortcode	0x0012 (hex.), 18 (dez.)
Länge	nicht erforderlich
Fehlercode	siehe Beschreibung der Eingangsdaten
Datenträger im Erfassungsbereich	siehe Beschreibung der Eingangsdaten
Daten (Bytes) verfügbar	siehe Beschreibung der Eingangsdaten
Datenträger-Zähler	siehe Beschreibung der Eingangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Eingangsdaten
Lesedaten	nicht erforderlich

7.12.10 Befehl: Schreib-Lese-Kopf-Identifikation

Der Befehl **Schreib-Lese-Kopf-Identifikation** fragt die folgenden Parameter des angeschlossenen Readers ab:

- ID
- Seriennummer
- Hardware-Version
- Firmware-Stand

Die Parameter sind im Reader im Identification Record zusammengefasst.

Übersicht Ausgangsdaten

Request	
Schleifenzähler	siehe Beschreibung der Ausgangsdaten
Befehlscode	0x0041 (hex.), 65(dez.)
Schreib-Lese-Kopf-Adresse	siehe Beschreibung der Ausgangsdaten
Länge EPC	nicht erforderlich
Startadresse	Startadresse im Identification Record, Angabe in Bytes
Länge	Länge der abzufragenden Daten 0: Lese vollständigen Parametersatz
Befehls-Time-out	nicht erforderlich
Schreib-Fragment-Nr.	nicht erforderlich
Lese-Fragment-Nr.	siehe Beschreibung der Ausgangsdaten
Schreibdaten	nicht erforderlich

Übersicht Eingangsdaten

Response	
Schleifenzähler	siehe Beschreibung der Eingangsdaten
Antwortcode	0x0041 (hex.), 65(dez.)
Länge	siehe Beschreibung der Eingangsdaten
Fehlercode	siehe Beschreibung der Eingangsdaten
Datenträger im Erfassungsbereich	siehe Beschreibung der Eingangsdaten
Daten (Bytes) verfügbar	siehe Beschreibung der Eingangsdaten
Datenträger-Zähler	steigt mit jedem gelesenen oder geschriebenen EPC
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Eingangsdaten
Lesedaten, Byte 0...19	ID: ARRAY [0...19] of BYTE
Lesedaten, Byte 20...35	Seriennummer: ARRAY [0...15] of BYTE
Lesedaten, Byte 36...37	Hardware-Version: INT16 (Little Endian)
Lesedaten, Byte 38...41	Firmware-Stand: ARRAY [0...] of BYTE: V (0x56), x, y, z (Vx.y.z)
Lesedaten, Byte 42...119	nicht erforderlich

7.12.11 Direkter Schreib-Lese-Kopf-Befehl



HINWEIS

Der Befehlscode für die schnelle Bearbeitung mit dem Schleifenzähler ist 0x2060 (hex.) bzw. 8288 (dez.).

Über einen direkten Befehl können Kommandos aus dem Reader-Protokoll direkt an den Reader gesendet werden. Die Kommandos werden über Angaben in den Schreib- und Lesedaten definiert und interpretiert.



HINWEIS

Das Reader-Protokoll ist nicht Bestandteil dieser Dokumentation und muss bei Turck angefragt und speziell freigegeben werden. Bei Fragen zum Reader-Protokoll wenden Sie sich an Turck.

Übersicht Ausgangsdaten

Request	
Schleifenzähler	siehe Beschreibung der Ausgangsdaten
Befehlscode	0x0060 (hex.), 96 (dez.)
Schreib-Lese-Kopf-Adresse	siehe Beschreibung der Ausgangsdaten
Länge EPC	0
Startadresse	nicht erforderlich
Länge	Länge der Beschreibung des direkten Befehls in den Schreibdaten, Angabe in Bytes
Befehls-Time-out	siehe Beschreibung der Ausgangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Ausgangsdaten
Schreibdaten	Beschreibung des direkten Befehls

Übersicht Eingangsdaten

Response	
Schleifenzähler	siehe Beschreibung der Eingangsdaten
Antwortcode	0x0060 (hex.), 96 (dez.)
Länge	Länge der Beschreibung des direkten Befehls in den Schreibdaten
Fehlercode	siehe Beschreibung der Eingangsdaten
Datenträger im Erfassungsbereich	siehe Beschreibung der Eingangsdaten
Daten (Bytes) verfügbar	siehe Beschreibung der Eingangsdaten
Datenträger-Zähler	siehe Beschreibung der Eingangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Eingangsdaten
Lesedaten	Antwort auf den direkten Befehl

Beispiel: Direkter Befehl in UHF-Anwendungen (Reader-Version abfragen)

Request	
Schleifenzähler	0
Befehlscode	0x0060
Schreib-Lese-Kopf-Adresse	0
Länge EPC	0
Startadresse	0
Länge	2
Befehls-Time-out	200
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	0
Schreibdaten	0x02 (CMD), 0x00 (application) – siehe debus-Protokoll
Response	
Schleifenzähler	0
Antwortcode	0x0060
Länge	12
Fehlercode	0
Datenträger im Erfassungsbereich	0
Daten (Bytes) verfügbar	0
Datenträger-Zähler	0
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	0
Lesedaten	0x02, 0x00, 0x01, 0x02, 0x03, 0x04, 0x8B, 0x20, 0x00, 0x01, 0x00, 0x01

Über das debus-Protokoll können die Lesedaten wie folgt interpretiert werden:

MSG	ERR	SNR0	SNR1	SNR2	SNR3	GTYP	VERS	HW
0x02	0x00	0x01	0x02	0x03	0x04	0x8B 0x20	0x00 0x01	0x00 0x01

- Seriennummer: 0x01020304
- Gerätetyp: 0x208B
- Software-Version: v1.00
- Hardware-Version: v1.00

7.12.12 Befehl: Datenträger-Passwort setzen



HINWEIS

Der Befehlscode für die schnelle Bearbeitung mit dem Schleifenzähler ist 0x2102 (hex.) bzw. 8450 (dez.).

Über den Befehl **Datenträger-Passwort setzen** wird ein Passwort in den Datenträger gesetzt. Beim Senden des Befehls darf sich nur ein Datenträger im Erfassungsbereich des Readers befinden. Nach dem Senden des Passworts können weitere Befehle (z. B. **Datenträger-Schutz setzen**) an den Datenträger gesendet werden. Über den Befehl **Datenträger-Passwort setzen** kann kein Kill-Passwort in den Datenträger gesetzt werden.

Übersicht Ausgangsdaten

Request	
Schleifenzähler	siehe Beschreibung der Ausgangsdaten
Befehlscode	0x0102 (hex.), 258 (dez.)
Schreib-Lese-Kopf-Adresse	siehe Beschreibung der Ausgangsdaten
Länge EPC	Angabe der EPC-Größe in Bytes, wenn ein bestimmter Datenträger geschützt werden soll. Der EPC muss in den Schreibdaten definiert werden (Startbyte: 0). Die Funktion der EPC-Länge ist abhängig vom verwendeten Befehl. 0: Keine Angabe eines EPC zur Ausführung des Befehls. Dabei darf sich nur ein Datenträger im Erfassungsbereich des Schreib-Lese-Kopfs befinden. > 0: EPC-Länge des Datenträgers, der geschützt werden soll, wenn in den Schreibdaten ein EPC vorhanden ist.
Startadresse	nicht erforderlich
Länge	4 Byte
Befehls-Time-out	siehe Beschreibung der Ausgangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Ausgangsdaten
Schreibdaten, Byte 0...3	Passwort: ARRAY [0...3] OF BYTE
Schreibdaten, Byte 4...127	nicht erforderlich

Übersicht Eingangsdaten

Response	
Schleifenzähler	siehe Beschreibung der Eingangsdaten
Antwortcode	0x0102 (hex.), 258 (dez.)
Länge	nicht erforderlich
Fehlercode	siehe Beschreibung der Eingangsdaten
Datenträger im Erfassungsbereich	siehe Beschreibung der Eingangsdaten
Daten (Bytes) verfügbar	siehe Beschreibung der Eingangsdaten
Datenträger-Zähler	siehe Beschreibung der Eingangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Eingangsdaten
Lesedaten	nicht erforderlich

7.12.13 Befehl: Schreib-Lese-Kopf-Passwort setzen

Über den Befehl **Schreib-Lese-Kopf-Passwort setzen** wird mit einem direkten Befehl ein Passwort für Schreibzugriff, Lesezugriff oder einen Kill-Befehl in den Datenträger gesetzt. Das Passwort wird flüchtig im Speicher des Readers hinterlegt. Nach einem Spannungs-Reset des Readers muss das Passwort erneut in den Reader gesetzt werden. Bei UHF-Anwendungen wird das Passwort im Speicher des Interfaces gespeichert.

Übersicht Ausgangsdaten

Request	
Schleifenzähler	siehe Beschreibung der Ausgangsdaten
Befehlscode	0x0100 (hex.), 256 (dez.)
Schreib-Lese-Kopf-Adresse	siehe Beschreibung der Ausgangsdaten
Länge EPC	nicht erforderlich
Startadresse	nicht erforderlich
Länge	nicht erforderlich
Befehls-Time-out	siehe Beschreibung der Ausgangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Ausgangsdaten
Schreibdaten, Byte 0...3	Passwort: ARRAY [0...3] OF BYTE
Schreibdaten, Byte 4...127	nicht erforderlich

Übersicht Eingangsdaten

Response	
Schleifenzähler	siehe Beschreibung der Eingangsdaten
Antwortcode	0x0100 (hex.), 256 (dez.)
Länge	nicht erforderlich
Fehlercode	siehe Beschreibung der Eingangsdaten
Datenträger im Erfassungsbereich	siehe Beschreibung der Eingangsdaten
Daten (Bytes) verfügbar	siehe Beschreibung der Eingangsdaten
Datenträger-Zähler	siehe Beschreibung der Eingangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Eingangsdaten
Lesedaten	nicht erforderlich

7.12.14 Befehl: Schreib-Lese-Kopf-Passwort zurücksetzen

Über den Befehl **Schreib-Lese-Kopf-Passwort zurücksetzen** wird mit einem direkten Befehl das Passwort für Schreibzugriff, Lesezugriff oder einen Kill-Befehl im Reader zurückgesetzt. Die Passwort-Funktion wird ausgeschaltet, zwischen Reader und Passwort findet kein Passwort-Austausch mehr statt.

Übersicht Ausgangsdaten

Request	
Schleifenzähler	siehe Beschreibung der Ausgangsdaten
Befehlscode	0x0101 (hex.), 257 (dez.)
Schreib-Lese-Kopf-Adresse	siehe Beschreibung der Ausgangsdaten
Länge EPC	nicht erforderlich
Startadresse	nicht erforderlich
Länge	nicht erforderlich
Befehls-Time-out	siehe Beschreibung der Ausgangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Ausgangsdaten
Schreibdaten	nicht erforderlich

Übersicht Eingangsdaten

Response	
Schleifenzähler	siehe Beschreibung der Eingangsdaten
Antwortcode	0x0101 (hex.), 257 (dez.)
Länge	nicht erforderlich
Fehlercode	siehe Beschreibung der Eingangsdaten
Datenträger im Erfassungsbereich	siehe Beschreibung der Eingangsdaten
Daten (Bytes) verfügbar	siehe Beschreibung der Eingangsdaten
Datenträger-Zähler	siehe Beschreibung der Eingangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Eingangsdaten
Lesedaten	nicht erforderlich

7.12.15 Befehl: Datenträger-Schutz setzen



HINWEIS

Der Befehlscode für die schnelle Bearbeitung mit dem Schleifenzähler ist 0x2103 (hex.) bzw. 8451 (dez.).

Über den Befehl **Datenträger-Schutz setzen** wird mit einem direkten Befehl der Passwort-Schutz für den Datenträger definiert. Dazu muss festgelegt werden, ob ein Schreibschutz und/oder ein Leseschutz gesetzt werden soll und für welchen Bereich des Datenträgers das Passwort gilt. Der Schutz für alle Bereiche wird mit einem Befehl definiert. Beim Senden des Befehls darf sich nur ein Datenträger im Erfassungsbereich des Readers befinden.

In einem Leseschutz ist immer auch ein Schreibschutz enthalten.



HINWEIS

Ein Schreibschutz für UHF-Datenträger kann nicht rückgängig gemacht werden.

Übersicht Ausgangsdaten

Request	
Schleifenzähler	siehe Beschreibung der Ausgangsdaten
Befehlscode	0x0103 (hex.), 259 (dez.)
Schreib-Lese-Kopf-Adresse	siehe Beschreibung der Ausgangsdaten
Länge EPC	Angabe der EPC-Größe in Bytes, wenn ein bestimmter Datenträger geschützt werden soll. Der EPC muss in den Schreibdaten definiert werden (Startbyte: 0). Die Funktion der Länge des EPC ist abhängig vom verwendeten Befehl. 0: Der Befehl wird für den Datenträger ausgeführt, der sich im Erfassungsbereich des Schreib-Lese-Kopfs befindet. > 0: EPC-Länge des Datenträgers, der geschützt werden soll, wenn in den Schreibdaten ein EPC vorhanden ist.
Startadresse	nicht erforderlich
Speicherbereich	mögliche Werte: ■ PC und EPC (Speicherbereich 1) ■ USER memory (Speicherbereich 3) Der gesamte ausgewählte Speicherbereich wird mit einem Passwort geschützt.
Länge	0 Byte
Befehls-Time-out	siehe Beschreibung der Ausgangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Ausgangsdaten
Schreibdaten, Byte 0	nicht erforderlich
Schreibdaten, Byte 1	0
Schreibdaten, Byte 2	0
Schreibdaten, Byte 3	0
Schreibdaten, Byte 4	nicht erforderlich
Schreibdaten, Byte 5	0
Schreibdaten, Byte 6	0
Schreibdaten, Byte 7	0
Schreibdaten, Byte 8...127	nicht erforderlich

Übersicht Eingangsdaten

Response	
Schleifenzähler	siehe Beschreibung der Eingangsdaten
Antwortcode	0x0103 (hex.), 259 (dez.)
Länge	nicht erforderlich
Fehlercode	siehe Beschreibung der Eingangsdaten
Datenträger im Erfassungsbereich	siehe Beschreibung der Eingangsdaten
Daten (Bytes) verfügbar	siehe Beschreibung der Eingangsdaten
Datenträger-Zähler	siehe Beschreibung der Eingangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Eingangsdaten
Lesedaten	nicht erforderlich

7.12.16 Befehl: Datenträger-Info



HINWEIS

Der Befehlscode für die schnelle Bearbeitung mit dem Schleifenzähler ist 0x2050 (hex.) bzw. 8272 (dez.).

Über den Befehl **Datenträger-Info** können die folgenden Chip-Informationen eines Datenträgers abgefragt werden:

- Allocation Class Identifier
- Tag Mask Designer Identifier
- Tag Model Number

Die Daten werden aus dem GSI-Record des Datenträgers abgefragt.

Übersicht Ausgangsdaten

Request	
Schleifenzähler	siehe Beschreibung der Ausgangsdaten
Befehlscode	0x0050 (hex.), 80 (dez.)
Schreib-Lese-Kopf-Adresse	siehe Beschreibung der Ausgangsdaten
Länge EPC	nicht erforderlich
Startadresse	Startadresse im GSI-Record
Länge	Länge der Systemdaten, die gelesen werden (Byte) 0: Alle Systemdaten werden gelesen
Befehls-Time-out	nicht erforderlich
Schreib-Fragment-Nr.	nicht erforderlich
Lese-Fragment-Nr.	siehe Beschreibung der Ausgangsdaten
Schreibdaten	nicht erforderlich

Übersicht Eingangsdaten

Response	
Schleifenzähler	siehe Beschreibung der Eingangsdaten
Antwortcode	0x0050 (hex.), 80(dez.)
Länge	siehe Beschreibung der Eingangsdaten
Fehlercode	siehe Beschreibung der Eingangsdaten
Datenträger im Erfassungsbereich	siehe Beschreibung der Eingangsdaten
Daten (Bytes) verfügbar	siehe Beschreibung der Eingangsdaten
Datenträger-Zähler	siehe Beschreibung der Eingangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Eingangsdaten
Lesedaten, Byte 0...3	Erste 32 Bytes der TID (Datenträger-Klasse, Hersteller und Chip-Typ)
Lesedaten, Byte 4...n	EPC (Länge variabel)

Chip-Informationen zu den UHF-Datenträgern

Name	TID-Speicher		Tag Model Number	Größe (Bits)		
	Allocation Class Identifier	Tag Mask Designer		EPC	TID	USER
Alien Higgs-3	0xE2	0x003	0x412	96...480	96	512
Alien Higgs-4	0xE2	0x003	0x414	16...128	96	128
NXP U-Code G2XM	0xE2	0x006	0x003	240	64	512
NXP U-Code G2XL	0xE2	0x006	0x004	240	64	–
NXP U-Code G2iM	0xE2	0x006	0x80A	256	96	512
NXP U-Code G2iM+	0xE2	0x006	0x80B	128...448	96	640...320
NXP U-Code G2iL	0xE2	0x006	0x806, 0x906, 0xB06	128	64	–
NXP U-Code G2iL+	0xE2	0x006	0x807, 0x907, 0xB07	128	64	–
NXP U-Code 7	0xE2	0x806	0x890	128	96	–
NXP U-Code 7xm (2k)	0xE2	0x806	0xF12	448	96	2048
Impinj Monza 4E	0xE2	0x001	0x10C	496	96	128
Impinj Monza 4D	0xE2	0x001	0x100	128	96	32
Impinj Monza 4QT	0xE2	0x001	0x105	128	96	512
Impinj Monza 5	0xE2	0x001	0x130	128	96	–
Impinj Monza R6	0xE2	0x001	0x160	96	96	–
Impinj Monza R6-P	0xE2	0x001	0x170	128	96	64

7.12.17 Befehl: UHF-Datenträger unwiderruflich deaktivieren (Kill)



HINWEIS

Der Befehlscode für die schnelle Bearbeitung mit dem Schleifenzähler ist 0x2200 (hex.) bzw. 8704 (dez.).

Über den Befehl **UHF-Datenträger unwiderruflich deaktivieren (Kill)** wird der Datenträger-Speicher unbenutzbar gemacht. Nach einem Kill-Befehl kann der Datenträger weder gelesen noch beschrieben werden. Ein Kill-Befehl kann nicht rückgängig gemacht werden.

Übersicht Ausgangsdaten

Request	
Schleifenzähler	siehe Beschreibung der Ausgangsdaten
Befehlscode	0x0200 (hex.), 512 (dez.)
Schreib-Lese-Kopf-Adresse	siehe Beschreibung der Ausgangsdaten
Länge EPC	Angabe der EPC-Größe in Bytes, wenn ein bestimmter Datenträger gelöscht werden soll. Der EPC muss in den Schreibdaten definiert werden (Startbyte: 0). Die Funktion der Länge des EPC ist abhängig vom verwendeten Befehl. 0: Keine Angabe einer EPC zur Ausführung des Befehls. Dabei darf sich nur ein Datenträger im Erfassungsbereich des Readers befinden. > 0: EPC-Länge des Datenträgers, der gelöscht werden soll, wenn in den Schreibdaten ein EPC vorhanden ist.
Startadresse	nicht erforderlich
Länge	1 Byte
Befehls-Time-out	siehe Beschreibung der Ausgangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Ausgangsdaten
Schreibdaten, Byte 0...3	Passwort: ARRAY [0...3] OF BYTE
Schreibdaten, Byte 4...127	nicht erforderlich

Übersicht Eingangsdaten

Response	
Schleifenzähler	siehe Beschreibung der Eingangsdaten
Antwortcode	0x0200 (hex.), 512 (dez.)
Länge	nicht erforderlich
Fehlercode	siehe Beschreibung der Eingangsdaten
Datenträger im Erfassungsbereich	siehe Beschreibung der Eingangsdaten
Daten (Bytes) verfügbar	siehe Beschreibung der Eingangsdaten
Datenträger-Zähler	siehe Beschreibung der Eingangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Eingangsdaten
Lesedaten	nicht erforderlich

7.12.18 Befehl: Einstellungen UHF-Schreib-Lese-Kopf wiederherstellen

Über den Befehl **Einstellungen UHF-Schreib-Lese-Kopf wiederherstellen** werden die Parameter des UHF-Readers aus einem Backup wiederhergestellt. Um den Befehl ausführen zu können, muss zuvor über den Befehl **Backup der Einstellungen des UHF-Schreib-Lese-Kopfs** ein Backup erstellt werden.

Übersicht Ausgangsdaten

Request	
Schleifenzähler	siehe Beschreibung der Ausgangsdaten
Befehlscode	0x1000 (hex.), 4096 (dez.)
Schreib-Lese-Kopf-Adresse	siehe Beschreibung der Ausgangsdaten
Länge EPC	nicht erforderlich
Startadresse	nicht erforderlich
Länge	nicht erforderlich
Befehls-Time-out	siehe Beschreibung der Ausgangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Ausgangsdaten
Schreibdaten	nicht erforderlich

Übersicht Eingangsdaten

Response	
Schleifenzähler	siehe Beschreibung der Eingangsdaten
Antwortcode	0x1000 (hex.), 4096 (dez.)
Länge	nicht erforderlich
Fehlercode	siehe Beschreibung der Eingangsdaten
Datenträger im Erfassungsbereich	siehe Beschreibung der Eingangsdaten
Daten (Bytes) verfügbar	siehe Beschreibung der Eingangsdaten
Datenträger-Zähler	siehe Beschreibung der Eingangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Eingangsdaten
Lesedaten	nicht erforderlich

7.12.19 Befehl: Backup der Einstellungen des UHF-Schreib-Lese-Kopfs

Der Befehl **Backup der Einstellungen des UHF-Schreib-Lese-Kopfs** speichert die aktuellen Einstellungen des Readers im Speicher des Interfaces. Das Backup bleibt auch nach einem Spannungsreset erhalten. Die Backup-Daten können über den Befehl **Einstellungen UHF-Schreib-Lese-Kopf wiederherstellen** wiederhergestellt werden.

Übersicht Ausgangsdaten

Request	
Schleifenzähler	siehe Beschreibung der Ausgangsdaten
Befehlscode	0x1001 (hex.), 4097 (dez.)
Schreib-Lese-Kopf-Adresse	siehe Beschreibung der Ausgangsdaten
Länge EPC	nicht erforderlich
Startadresse	nicht erforderlich
Länge	nicht erforderlich
Befehls-Time-out	siehe Beschreibung der Ausgangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Ausgangsdaten
Schreibdaten	nicht erforderlich

Übersicht Eingangsdaten

Response	
Schleifenzähler	siehe Beschreibung der Eingangsdaten
Antwortcode	0x1001 (hex.), 4097 (dez.)
Länge	nicht erforderlich
Fehlercode	siehe Beschreibung der Eingangsdaten
Datenträger im Erfassungsbereich	siehe Beschreibung der Eingangsdaten
Daten (Bytes) verfügbar	siehe Beschreibung der Eingangsdaten
Datenträger-Zähler	siehe Beschreibung der Eingangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Eingangsdaten
Lesedaten	nicht erforderlich

7.12.20 Befehl: Fehler/Status UHF-Schreib-Lese-Kopf lesen

Über den Befehl **Fehler/Status UHF-Schreib-Lese-Kopf lesen** können Fehler- und Statusmeldungen des UHF-Readers ausgelesen werden.

Übersicht Ausgangsdaten

Request	
Schleifenzähler	siehe Beschreibung der Ausgangsdaten
Befehlscode	0x0042 (hex.), 66 (dez.)
Schreib-Lese-Kopf-Adresse	nicht erforderlich
Länge EPC	nicht erforderlich
Startadresse	Adresse im Get Status response -Record
Länge	Länge der Daten, die aus dem Get Status response -Record ausgelesen werden sollen 0: Gesamten Get Status response -Record lesen
Befehls-Time-out	siehe Beschreibung der Ausgangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Ausgangsdaten
Schreibdaten	nicht erforderlich

Übersicht Eingangsdaten

Response	
Schleifenzähler	siehe Beschreibung der Eingangsdaten
Antwortcode	0x042 (hex.), 66 (dez.)
Länge	siehe Beschreibung der Eingangsdaten
Fehlercode	siehe Beschreibung der Eingangsdaten
Datenträger im Erfassungsbereich	siehe Beschreibung der Eingangsdaten
Daten (Bytes) verfügbar	siehe Beschreibung der Eingangsdaten
Datenträger-Zähler	siehe Beschreibung der Eingangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Eingangsdaten
Lesedaten, Byte 0...(Länge-1)	<ul style="list-style-type: none"> ■ Status allgemein: 1 Byte allgemeiner Status ■ RF-Status: 1 Byte Status des RF-Moduls ■ Gerätestatus: 1 Byte gerätespezifischer Status-Informationen ■ RF-Modus: 1 Byte, definiert den Grund für den Start eines Lesevorgangs ■ Trigger-Status: 1 Byte, Trigger-Nummer des RF-Modus ■ I/O-Status: 1 Byte, Status der Ein- und Ausgänge (0 = low, 1 = high) ■ Umgebungstemperatur: 1 Byte, Umgebungstemperatur in °C (Datenformat: 8 bit, Zweierkomplement) ■ PA-Temperatur: 1 Byte, PA-Temperatur in °C (Datenformat: 8 bit, Zweierkomplement) ■ RF-Antennenemperatur: 1 Byte, Antennentemperatur in °C (Datenformat: 8 bit, Zweierkomplement) ■ Transmit Power: 2 Bytes, Ausgangsleistung des Readers in 1/10 dBm-Schritten, LSB...MSB (Datenformat: 16 bit, Zweierkomplement) ■ Reverse Power: 2 Byte zurückgestrahlte Leistung in 1/10 dBm-Schritten, LSB...MSB (Datenformat: 16 bit, Zweierkomplement) ■ Antenna DC Resistance: 4 Bytes Widerstand am Antennenport in Ω, LSB...MSB ■ Jammer Power: 2 Bytes, Eingangsleistung am RX-Port in 1/10 dBm-Schritten, LSB...MSB (Datenformat: 16 bit, Zweierkomplement) ■ Kanal: Nummer des aktuell genutzten Kanals (Offset zum nächsten verfügbaren Kanal)
Lesedaten, Byte (Länge)...127	nicht erforderlich

Lesedaten auswerten – allgemeiner Status

Bit	Bedeutung
7	Reader wurde zurückgesetzt (nach Reset)
6	Reader-Konfiguration beschädigt, Default-Einstellungen werden genutzt
5	Testmodus aktiv
1	Datenträger vorhanden

Lesedaten auswerten – RF-Status

Bit	Bedeutung
4	Grenzwert für abgestrahlte Leistung überschritten
3	kein freier Kanal vorhanden
2	Antennenwiderstand zu hoch oder zu niedrig
1	Rückleistung zu hoch
0	PLL nicht gesperrt

Lesedaten auswerten – Gerätestatus

Bit	Bedeutung
4	Fehler bei der Nachrichtengenerierung (im Polling-Modus außerhalb des Speicherbereichs)
3	Temperaturwarnung
2	Temperatur zu hoch
1	Kommunikationsfehler
0	Konfiguration ungültig. Ausführung des Kommandos nicht möglich.

Lesedaten auswerten – RF-Modus

Wert	Bedeutung
0x00	keine (Träger aus)
0x01	Modus 1: Trigger ist digitales Signal (Flanke), Time-out
0x02	Modus 2: Trigger ist digitales Signal (Flanke), Time-out
0x03	Modus 3: Trigger ist digitales Signal (Level), kein Time-out
0x04	Trigger ist ein Kommando
0x08	reserviert
0x10	DCU-gesteuerter Lesevorgang
0x20	Continuous Mode
0x80	automatischer Trigger (Presence Sensing Mode)

Lesedaten auswerten – I/O-Status

Wert	Bedeutung
7	Ausgang 4
6	Ausgang 3
5	Ausgang 2
4	Ausgang 1
3	Eingang 4
2	Eingang 3
1	Eingang 2
0	Eingang 1

7.12.21 Befehl: Reset

Über den Befehl **Reset** werden Reader und Interface zurückgesetzt.

Übersicht Ausgangsdaten

Request	
Schleifenzähler	siehe Beschreibung der Ausgangsdaten
Befehlscode	0x8000 (hex.), 32768 (dez.)
Schreib-Lese-Kopf-Adresse	siehe Beschreibung der Ausgangsdaten
Länge EPC	nicht erforderlich
Startadresse	0: Software-Reset 1: Spannungs-Reset
Länge	nicht erforderlich
Befehls-Time-out	siehe Beschreibung der Ausgangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Ausgangsdaten
Schreibdaten	nicht erforderlich

Übersicht Eingangsdaten

Response	
Schleifenzähler	siehe Beschreibung der Eingangsdaten
Antwortcode	0x8000 (hex.), 32768 (dez.)
Länge	nicht erforderlich
Fehlercode	siehe Beschreibung der Eingangsdaten
Datenträger im Erfassungsbereich	siehe Beschreibung der Eingangsdaten
Daten (Bytes) verfügbar	siehe Beschreibung der Eingangsdaten
Datenträger-Zähler	siehe Beschreibung der Eingangsdaten
Schreib-Fragment-Nr.	0
Lese-Fragment-Nr.	siehe Beschreibung der Eingangsdaten
Lesedaten	nicht erforderlich

8 Betreiben

8.1 LED-Anzeigen

Das Gerät verfügt über folgende LED-Anzeigen:

- Versorgungsspannung
- Sammel- und Busfehler
- Status
- Diagnose

LED PWR	Bedeutung
aus	keine Spannungsversorgung
grün	Spannungsversorgung fehlerfrei
gelb	Unterspannung innerhalb der Toleranz
rot	Unterspannung außerhalb der Toleranz

LED RFON	Bedeutung
aus	Funkfeld ausgeschaltet
grün	Funkfeld eingeschaltet

LED DATA	Bedeutung
aus	kein Datenträger im Feld, kein Datentransfer
blinkt gelb	Datenträger im Feld, Datentransfer über die Luftschnittstelle

LED DIAG	Bedeutung
aus	kein Fehler
rot	Fehler

Die folgenden Mehrfarben-LEDs sind frei programmierbar. Die untenstehenden Tabellen beschreiben die Default-Anzeigefunktionen.

DXP-LEDs (Digitale Kanäle, LEDs DXP0...3)		
LED grün	LED rot	Bedeutung
aus	aus	kein I/O-Signal vorhanden
leuchtet	aus	I/O-Signal vorhanden
aus	leuchtet	Überlast am Ausgang
blinkt	blinkt	Überlast der Hilfsversorgung

LED APPL	Bedeutung
blinkt weiß	Wink-Kommando aktiv

9 Störungen beseitigen

Wenn das Gerät nicht wie erwartet funktioniert, gehen Sie wie folgt vor:

- ▶ Umgebungsstörungen ausschließen.
- ▶ Anschlüsse des Geräts auf Fehler untersuchen.
- ▶ Gerät auf Parametrierfehler überprüfen.

Wenn die Fehlfunktion weiterhin besteht, liegt eine Gerätestörung vor. In diesem Fall nehmen Sie das Gerät außer Betrieb und ersetzen Sie es durch ein neues Gerät des gleichen Typs.

10 Instand halten

Der ordnungsgemäße Zustand der Verbindungen und Kabel muss regelmäßig überprüft werden.

Die Geräte sind wartungsfrei, bei Bedarf trocken reinigen.

10.1 Firmware-Update über den Webserver durchführen



ACHTUNG

Unterbrechung der Spannungsversorgung und Ethernet-Verbindung während des Firmware-Updates

Geräteschäden durch fehlerhaftes Firmware-Update

- ▶ Spannungsversorgung des Geräts während des Firmware-Updates nicht unterbrechen.
- ▶ Während des Firmware-Updates keinen Spannungsreset durchführen.
- ▶ Ethernet-Verbindung des Geräts während des Firmware-Updates nicht unterbrechen.

- ▶ Webserver öffnen und auf dem Gerät einloggen.
- ▶ **Firmware** → **FIRMWARE-DATEI AUSWÄHLEN** klicken.

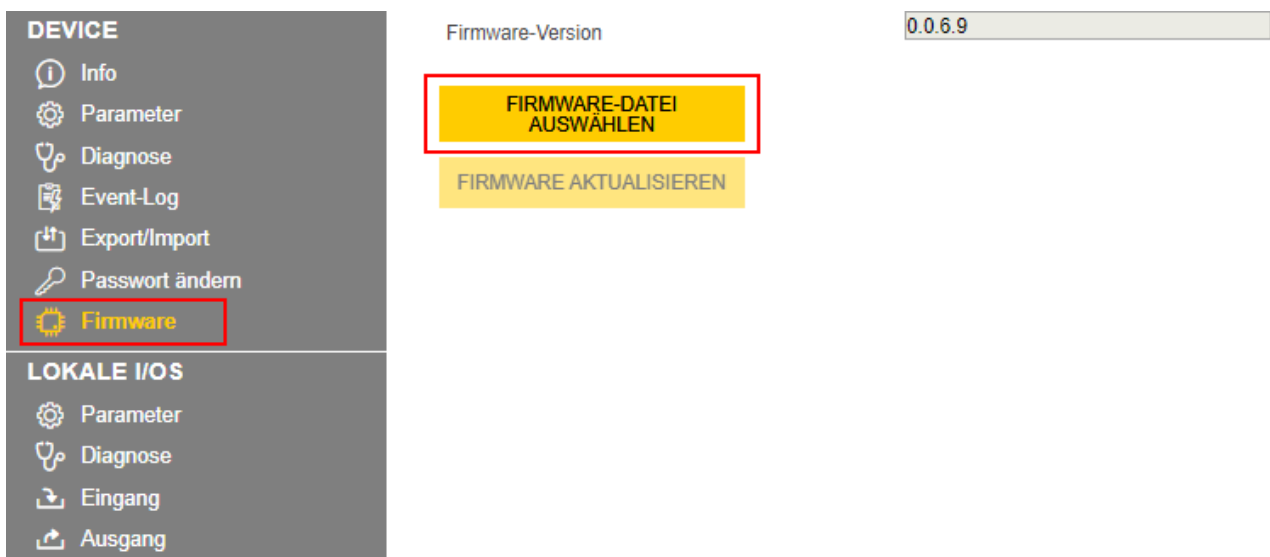


Abb. 39: Neue Firmware-Datei auswählen

- ▶ Ablageort der Datei wählen und Datei auswählen.
- ▶ Firmware-Update über die Schaltfläche **FIRMWARE AKTUALISIEREN** starten.
 - ⇒ Der Fortschritt des Firmware-Updates wird angezeigt.
- ▶ Nach erfolgreichem Update-Vorgang das Gerät mit einem Klick auf **OK** neu starten.

11 Reparieren

Das Gerät ist nicht zur Reparatur durch den Benutzer vorgesehen. Sollte das Gerät defekt sein, nehmen Sie es außer Betrieb. Bei Rücksendung an Turck beachten Sie unsere Rücknahmebedingungen.

11.1 Geräte zurücksenden

Rücksendungen an Turck können nur entgegengenommen werden, wenn dem Gerät eine Dekontaminationserklärung beiliegt. Die Erklärung steht unter <http://www.turck.de/de/produkt-retoure-6079.php> zur Verfügung und muss vollständig ausgefüllt, wetter- und transportsicher an der Außenseite der Verpackung angebracht sein.

12 Entsorgen



Die Geräte müssen fachgerecht entsorgt werden und gehören nicht in den normalen Hausmüll.

13 Technische Daten

Technische Daten	
Elektrische Daten	
Betriebsspannung	18...30 VDC
DC Bemessungsbetriebsstrom	≤ 1000 mA
Datenübertragung	elektromagnetisches Wechselfeld
Funk- und Protokollstandards	ISO 18000-6C EN 302208 EPCglobal Gen 2
Antennenpolarisation	zirkular/linear, einstellbar
Antennenhalbwidthsbreite	65°
Ausgangsfunktion	lesen/schreiben
Mechanische Daten	
Einbaubedingung	nicht bündig
Umgebungstemperatur	-20...+50 °C
Abmessungen	300 × 300 × 61,7 mm
Gehäusewerkstoff	Aluminium, AL, silber
Material aktive Fläche	Glasfaser verstärktes Polyamid, PA6-GF30, schwarz
Vibrationsfestigkeit	55 Hz (1 mm)
Schockfestigkeit	30 g (11 ms)
Schutzart	IP67
Kanalanzahl	4
Elektrischer Anschluss	RP-TNC
Eingangsimpedanz	50 Ω
Systembeschreibung	
Prozessor	ARM Cortex A8, 32 Bit, 800 MHz
ROM-Speicher	512 MB Flash
RAM-Speicher	512 MB DDR3
Systemdaten	
Übertragungsrate Ethernet	10 Mbit/s / 100 Mbit/s
Anschlussstechnik Ethernet	1 × M12, 4-polig, D-codiert
Digitale Eingänge	
Kanalanzahl	4
Anschlussstechnik Eingänge	M12, 5-polig
Eingangstyp	PNP
Schaltsschwelle	EN 61131-2 Typ 3, PNP
Signalspannung Low-Pegel	< 5 V
Signalspannung High-Pegel	> 11 V
Signalstrom Low-Pegel	< 1,5 mA
Signalstrom High-Pegel	> 2 mA
Art der Eingangsdiagnose	Kanaldiagnose

Technische Daten	
Digitale Ausgänge	
Kanalanzahl	4
Anschluss technik Ausgänge	M12, 5-polig
Ausgangstyp	PNP
Art der Ausgangsdiagnose	Kanaldiagnose

14 EU-Konformitätserklärung

Hiermit erklärt die Hans Turck GmbH & Co. KG, dass der Funkanlagentyp TN-UHF-Q...L...-EU... der Richtlinie 2014/53/EU entspricht. Der vollständige Text der EU-Konformitätserklärung ist unter der folgenden Internetadresse verfügbar: www.turck.com

15 Turck-Niederlassungen – Kontaktdaten

Deutschland	Hans Turck GmbH & Co. KG Witzlebenstraße 7, 45472 Mülheim an der Ruhr www.turck.de
Australien	Turck Australia Pty Ltd Building 4, 19-25 Duerdin Street, Notting Hill, 3168 Victoria www.turck.com.au
Belgien	TURCK MULTIPROX Lion d'Orweg 12, B-9300 Aalst www.multiprox.be
Brasilien	Turck do Brasil Automação Ltda. Rua Anjo Custódio Nr. 42, Jardim Anália Franco, CEP 03358-040 São Paulo www.turck.com.br
China	Turck (Tianjin) Sensor Co. Ltd. 18,4th Xinghuazhi Road, Xiqing Economic Development Area, 300381 Tianjin www.turck.com.cn
Frankreich	TURCK BANNER S.A.S. 11 rue de Courtalin Bat C, Magny Le Hongre, F-77703 MARNE LA VALLEE Cedex 4 www.turckbanner.fr
Großbritannien	TURCK BANNER LIMITED Blenheim House, Hurricane Way, GB-SS11 8YT Wickford, Essex www.turckbanner.co.uk
Indien	TURCK India Automation Pvt. Ltd. 401-403 Aurum Avenue, Survey. No 109 /4, Near Cummins Complex, Baner-Balewadi Link Rd., 411045 Pune - Maharashtra www.turck.co.in
Italien	TURCK BANNER S.R.L. Via San Domenico 5, IT-20008 Bareggio (MI) www.turckbanner.it
Japan	TURCK Japan Corporation ISM Akihabara 1F, 1-24-2, Taito, Taito-ku, 110-0016 Tokyo www.turck.jp
Kanada	Turck Canada Inc. 140 Duffield Drive, CDN-Markham, Ontario L6G 1B5 www.turck.ca
Korea	Turck Korea Co, Ltd. A605, 43, Iljik-ro, Gwangmyeong-si 14353 Gyeonggi-do www.turck.kr
Malaysia	Turck Banner Malaysia Sdn Bhd Unit A-23A-08, Tower A, Pinnacle Petaling Jaya, Jalan Utara C, 46200 Petaling Jaya Selangor www.turckbanner.my

Mexiko	Turck Comercial, S. de RL de CV Blvd. Campestre No. 100, Parque Industrial SERVER, C.P. 25350 Arteaga, Coahuila www.turck.com.mx
Niederlande	Turck B. V. Ruiterlaan 7, NL-8019 BN Zwolle www.turck.nl
Österreich	Turck GmbH Graumanngasse 7/A5-1, A-1150 Wien www.turck.at
Polen	TURCK sp.z.o.o. Wroclawska 115, PL-45-836 Opole www.turck.pl
Rumänien	Turck Automation Romania SRL Str. Siriului nr. 6-8, Sector 1, RO-014354 Bucuresti www.turck.ro
Schweden	Turck AB Fabriksstråket 9, 433 76 Jonsered www.turck.se
Singapur	TURCK BANNER Singapore Pte. Ltd. 25 International Business Park, #04-75/77 (West Wing) German Centre, 609916 Singapore www.turckbanner.sg
Südafrika	Turck Banner (Pty) Ltd Boeing Road East, Bedfordview, ZA-2007 Johannesburg www.turckbanner.co.za
Tschechien	TURCK s.r.o. Na Brne 2065, CZ-500 06 Hradec Králové www.turck.cz
Türkei	Turck Otomasyon Ticaret Limited Sirketi Inönü mah. Kayisdagi c., Yesil Konak Evleri No: 178, A Blok D:4, 34755 Kadiköy/ Istanbul www.turck.com.tr
Ungarn	TURCK Hungary kft. Árpád fejedelem útja 26-28., Óbuda Gate, 2. em., H-1023 Budapest www.turck.hu
USA	Turck Inc. 3000 Campus Drive, USA-MN 55441 Minneapolis www.turck.us

TURCK

Your Global Automation Partner

Over 30 subsidiaries and
60 representations worldwide!

100003066 | 2024/09



www.turck.com