

# Benutzerhandbuch

---

## Langstrecken-UHF-RFID- Leser

### **HD-RD06-UHF**

Inhaltsverzeichnis	
Technische Daten: .....	5
Inhalt des Kits:.....	6
Beschreibung des Protokolls .....	6
Datenrahmenformate .....	7
Command Frame Format (von HOST).....	7
Antwortrahmenformat (von READER).....	9
Zusammenfassung der Kommandos.....	11
Befehle 18000-6B.....	12
Reader Custom Befehl.....	13
RFID-Tag-Fehlercodes.....	16
Tag-Speicher und Zugriffsbefehle .....	18
Detaillierte Beschreibung des Kommandos .....	19
EPC C1G2-Befehle.....	20
Inventar-Tags.....	20
Lesedaten .....	29
Datenaufzeichnung.....	32
Eingabe der EPC-Nummer .....	36
Lese-/Schreibschutzeinstellung für einen bestimmten Speicher .....	37
Blöcke löschen .....	41
Leseschutzkonfiguration (gemäß EPC-Nummer) .....	44
Leseschutzkonfiguration (ohne EPC-Nummer) .....	46
Entsperren Leseschutz .....	47
Lesen Sie die Schutzstatusprüfung .....	48
EAS-Konfiguration .....	49
EAS-Alarmerkennung .....	52
Einzel-Tag-Inventar.....	52
Blocktypisierungsbefehl.....	53
Ermittlung der Betriebsparameter von Monza4QT .....	56
Modifiziere die Betriebsparameter von Monza4QT .....	59
Erweiterte Datenlesung mit zugewiesener Maske .....	62
Erweiterte Datenlesung mit zugewiesener Maske .....	65
Speicherpuffer-Inventar .....	68
Mischinventar.....	3

Inventar mit EPC-Nummer.....	11
QT Wrestling.....	14
18000-6B Kommandos.....	19
Fordern Sie Etiketten entsprechend einer bestimmten Bedingung an.....	20
Leserdaten.....	22
Datenaufzeichnung.....	23
Erhalten Sie einen Lock-Zustand.....	24
Lock-Byte.....	25
Individuelle Lesebefehle.....	25
Ändern Sie die Betriebsfrequenz.....	28
Ändere die Adresse des Lesers.....	31
Ändern Sie die Zählzeit des Lesers.....	32
Modifiziere die serielle Baudrate.....	32
RF-Leistung modifizieren.....	34
Summer-/LED-Steuerung.....	35
Antennenmultiplex-Konfiguration.....	35
Schalte den Summer an/aus.....	37
GPIO-Regelung.....	37
Erhalten Sie GPIO-Status.....	38
Holen Sie sich eine eindeutige Seriennummer für Ihren Leser.....	38
Modifiziere die benutzerdefinierte Funktion eines Tags.....	39
Antennenprüfung aktivieren.....	40
Änderung der Kommunikationsschnittstelle.....	41
Konfiguration der Antennen-Reflexionsdämpfungsschwellenwerte modifizieren oder belasten.....	42
Änderung der maximalen EPC/TID-Längenkonfiguration für den Speicherpuffer.....	43
EPC/TID-Konfiguration mit maximaler Länge.....	44
Daten aus einem Speicherpuffer abholen.....	44
Speicherpuffer löschen.....	46
Holen Sie die Gesamtmenge der Tags aus dem Speicherpuffer.....	47
Echtzeit-Inventarmodus-Parameteränderung.....	47
Ändern Sie den Betriebsmodus des Lesers.....	50
Hol den Lesemodus.....	54
Laden oder verändern Sie die Pause von Herzschlagpaketen im Echtzeit-Inventar.....	56
Anpassung der HF-Leistungskonfiguration separat für Schreibvorgänge.....	57

Last-RF-Leistungskonfiguration für den Schreibbetrieb.....	58
Ändern oder laden Sie die maximale Schreibzeit-Konfiguration .....	59
Ändern Sie das Passwort der benutzerdefinierten Tag-Funktion .....	61
Holen Sie sich das Passwort für die benutzerdefinierte Tag-Funktion.....	61
Leseprofil laden/verändern .....	62
Synchronisations-EM4325-Zeitstempel .....	63
Erhalten Sie EM4325 Temperaturdaten .....	64
Externe Daten über EM4325 SPI erhalten.....	68
Reset EM4325-Warnung .....	72
Ändern oder laden Sie DRM-Konfigurationen .....	75
Messung der Antennenreflexionsdämpfung .....	76
Messen Sie die Stromtemperatur des Lesers .....	77
Einstellung erweiterter Parameter.....	78
Einstellung erweiterter Parameter .....	79
Starte eine schnelle Inventur.....	79
Stopp Quick Inventory .....	81
Individuelle Lesebefehle.....	81
Tabelle der Leserkonfigurationsparameter.....	81
Beschreibung der Funktion des Leserparameters .....	82

## Technische Daten:

- **Garantie:** 1 Jahr
- **Farbe:** Weiß
- **Material:** ABS, Gusseisen
- **Betriebsfrequenz:** 865–868 MHz, 902–928 MHz
- **RFID-Ausgang:** 0 dBm bis 26 dBm\* mit +/-1 dBm
- **Lesedistanz:** bis zu 2 Meter
- **Integrierte Antenne:** 5,5 dBm
- **Höchste Inventargeschwindigkeit:**  
>50 Tags pro Sekunde
- **RSSI-Tag:** unterstützt
- **Antennendetektor:** Unterstützt
- **Kommunikationsschnittstelle:** 10M/100M adaptives Ethernet, RS232
- **Buchsen:** DC, GPIO, RS232, RJ45
- **Stromversorgung:** 220 V Wechselstrom, +12 V/5A Gleichstromwandlerausgang
- **Betriebssystem:** Linux
- **Abmessungen des Geräts:** 12,9 x 12,9 x 4,9 cm
- **Produktmaße:** 25,5 x 20 x 7 cm
- **Gerätegewicht:** 450 g
- **Gewicht mit Verpackung:** 1250 g
- **Betriebstemperatur:** -20°C bis 55°C
- **Lagertemperatur:** -20°C bis +85°C
- **Betriebsfeuchtigkeit:** 5 %~95 % nicht kondensierend
- **Speicherfeuchtigkeit:** 5 %~95 % nicht kondensierend

## Inhalt des Kits:

- RFID-Tag-Leser
- Stromversorgung
- Befestigungsschrauben
- Metallplatte

## Beschreibung des Protokolls

In der typischen Kommunikation liefert der Host den Befehl und die Parameter an den Leser, der Leser soll den Befehl ausführen und das Ergebnis oder die Antwort der Daten liefern. Der Leser erhält und führt in jedem Kommunikationszyklus EINEN Befehl aus. Der Leser kann den neuen Befehl nicht empfangen, bevor der letzte die Ausführung abgeschlossen hat und eine Antwort liefert. Jeder Befehl, der während der Ausführung des Befehls gesendet wird, wird vom Leser ignoriert.

### Um einen Befehl vom Wirt zu übermitteln:

Gast	Regie	Leser
Befehlsrahmen	->	

### Beschreibung:

Wenn der Leser den Befehl erhält, generiert das Gerät innerhalb einer vordefinierten Zählzeit eine Antwort an den Host. Die Zeit der Datenübertragung ist hier irrelevant.

Während der Kommunikation sollte die Übertragungsunterbrechung zwischen jeweils zwei aufeinanderfolgenden Bytes weniger als 15 ms betragen. Wenn die Lücke 15 ms übersteigt, sollten alle

empfangenen Daten verworfen werden und der normale Empfangsprozess beginnt mit dem nächsten Byte.

**Leserantwort:**

Leser	Regie	Gast
Answer Framework	->	

Beschreibung:

Während der Kommunikation sollte das Übertragungsintervall zwischen zwei aufeinanderfolgenden Bytes weniger als 15 ms betragen.

Im Folgenden ist ein typischer Kommunikationsprozess:

1. Der Host sendet den Befehl an den Leser und wartet auf eine Antwort;
2. Der Leser empfängt den Befehl, startet die angegebene Operation gemäß dem empfangenen Rahmen und erzeugt eine Antwort (auf die Operation und/oder die Daten) an den Host;
3. Der Moderator erhält eine Antwort vom Leser

## Datenrahmenformate

Command Frame Format (von HOST)

Nur	Adr	Cmd	Datum[]	LSB-CRC16	MSB-CRC16
-----	-----	-----	---------	-----------	-----------

**Beschreibung der Formate:**

	Länge (in Bytes)	Beschreibung
Nur	1	Die Länge des Befehlsrahmens,



		bestimmt durch die Anzahl der Bytes zwischen Adr und MSB_CRC16, also der Länge der Data[] + 4. Die Minimal- und Maximumswerte von Len sind 4 und 255.
Adr	1	Die Adresse des Lesegeräts, der korrekte Geräteadressbereich ist 0x00 ~ 0xFE. 0xFF ist die Sendeadresse. Der Leser reagiert nur auf das mit der Adresse bereitgestellte Befehlsrahmen Identisch oder 0xFF. Standardgeräteadresse: 0x
Cmd	1	Vermittlungscode
Datum[]	Variable	Ein Parameterfeld kann der Host Parameter angeben oder auch nicht. je nach verschiedenen Befehlsformaten.
LSB-CRC16	1	Niedriges Byte CRC16. Diese 2 Bytes sind die CRC16-Berechnung von Len zu Data[]

MSB-CRC16	1	CRC16 High Byte
-----------	---	-----------------

Antwortrahmenformat (von READER)

<b>Nur</b>	<b>Adr</b>	<b>reCmd</b>	<b>Status</b>	<b>Datum[]</b>	<b>LSB-CRC16</b>	<b>MSB-CRC16</b>
------------	------------	--------------	---------------	----------------	------------------	------------------

### Beschreibung der Formate:

	<b>Länge (in Bytes)</b>	<b>Beschreibung</b>
Nur	1	Die Länge des Antwortrahmens, bestimmt durch die Anzahl der Bytes zwischen Adr und MSB_CRC16), d. h. die Länge der Date[] + 5.
Adr	1	Leseradresse
reCmd	1	reCmd gibt an, auf welchen Befehl dieses Frame reagiert . Für einen nicht erkannten Befehl sollte reCmd in 0x00 sein.
Status	1	Status/Ergebnis der Befehlsausführung
Datum[]	Variable	In einem Datenfeld kann der Leser Parameter liefern oder auch nicht

		nach tatsächlichen Befehlen
LSB-CRC16	1	Niedriges Byte CRC16. Diese 2 Bytes sind die CRC16-Berechnung von Len zu Daten[ ].
MSB-CRC16	1	CRC16 High Byte

### CRC16-Algorithmus in C:

```

#define PRESET_VALUE 0xFFFF #define POLYNOMIELLE 0x8408
unsigned int uiCrc16Cal (unsigned char const * pucY, unsigned
char ucX)
{
unsigned char ucl,ucJ;
unsigned short int uiCrcValue = PRESET_VALUE; for(ucl = 0; ucl <
ucX; ucl++)
{
uiCrcValue = uiCrcValue ^ *(pucY + ucl); for(ucJ = 0, ucJ < 8,
ucJ++)
{
if(uiCrcValue & 0x0001) {
uiCrcValue = (uiCrcValue >> 1) ^ POLYNOMIAL;
}else {
uiCrcValue = (uiCrcValue >> 1); }

```

```
} }r eturn uiCrcValue; }
```

**pucY** ist der Indikator für den Array-Eintrag der CRC16-Berechnung, ucX ist die Anzahl der in dieser Berechnung enthaltenen Bytes. Der Rückgabewert von 0x0000 zeigt an, dass die Eingabe die CRC16-Verifikation bestanden hat. Wann immer ein Host einen Dataframe von einem Leser empfängt, sollte der Host den Rahmen immer durch den Aufruf dieses CRC16-Algorithmus validieren.

## Zusammenfassung der Kommandos

LP.	Vermittlungscode	Ausstattung
1	0x01	Liste der Tags
2	0x02	Datenlesebefehl
3	0x03	Write Data Befehl
4	0x04	EPC-Nummer
5	0x05	Kill-Tag
6	0x06	Lese- /Schreibschutzeinstellung für einen bestimmten Speicher
7	0x07	Einen Block löschen
8	0x08	Leseschutzkonfiguration (gemäß EPC-Nummer)
9	0x09	Leseschutzkonfiguration (keine EPC-Nummer)
10	0x0a	Entblocken-Leseschutz
11	0x0b	Überprüfen Sie den Read-Protect-Status
12	0x0c	EAS-Konfiguration
13	0x0d	EAS-Alarmerkennung
14	0x0f	Einzel-Tag-Inventar
15	0x10	Schreibblöcke

16	0x11	Ermittlung der Arbeitsparameter von Monza4QT
17	0x12	Änderung der Monza4QT-Betriebsparameter
18	0x15	Erweiterte Datenlesung mit zugewiesener Maske
19	0x16	Erweiterte Datenaufzeichnung mit zugewiesener Maske
20	0x18	Speicherpuffer-Inventar
21	0x19	Bestandsmischung
22	0x1a	Inventar mit EPC-Nummer
23	0x1b	QT-Inventar

#### Befehle 18000-6B

LP.	Vermittlungscode	Ausstattung
1	0x50	Inventarbefehl für ein EINZELNES Tag. Dieser Befehl verlangt nur einen Tag pro Zyklus, ohne Inventarbedingung.
2	0x51	Inventarbefehl für MEHRERE Tags. Abfragen Sie nach Tags gemäß einer vordefinierten Bedingung und geben Sie nur die UIDs der entsprechenden Tags zurück.
3	0x52	Ein Befehl zum Auslesen von Daten.

		Lesen Sie Daten aus dem Tag, bis zu 32 Byte pro Befehlszyklus.
4	0x53	Befehl Write Data. Schreibe Daten auf das Tag, bis zu 32 Bytes pro Befehlszyklus.
5	0x54	Befehl, um den Lock-Status zu erhalten. Überprüft den Schlosstatus einer angegebenen Lagereinheit.
6	0x55	Byte-Lock-Befehl. Sperrt ein bestimmtes (freigeschaltetes) Byte im Tag.

## Reader Custom Befehl

LP.	Vermittlungscode	Ausstattung
1	0x21	Erhalten Sie Informationen über Ihren Leser
2	0x22	Betriebsfrequenzänderung
3	0x24	Änderung der Leseradresse
4	0x25	Änderung der Leseinventarzeit
5	0x28	Modifikation der seriellen Übertragungsrates
6	0x2f	RF-Leistungsmodifikation
7	0x33	LED/Buzzer-Steuerung
8	0x3f	Antennenmultiplexing-Konfiguration

9	0x40	Schalte den Summer an/aus
10	0x46	GPIO-Regelung
11	0x47	GPIO-Status erhalten
12	0x4c	Holen Sie sich eine eindeutige Seriennummer für Ihren Leser
13	0x3a	Modifiziere eine benutzerdefinierte Tag-Funktion
14	0x66	Antennenprüfung aktivieren
15	0x6a	Modifikation der Kommunikationsschnittstelle
16	0x6e	Änderung oder Belastung der Antennenrücklaufschwollen-Konfiguration
17	0x70	Änderung der EPC/TID-Konfiguration der maximalen Länge für den Speicherpuffer
18	0x71	EPC/TID-Konfiguration mit maximaler Länge
19	0x72	Daten aus einem Speicherpuffer abholen
20	0x73	Speicherpuffer löschen
21	0x74	Holen Sie die Gesamtmenge der Tags aus dem Speicherpuffer
22	0x75	Echtzeit-Inventarmodus-Parameteränderung
23	0x76	Änderung des Betriebsmodus
24	0x77	Echtzeit-Laden von Inventarmodusparametern
25	0x78	Lade/ändere den Timeout des Echtzeit-Inventar-Herzschlagpakets

26	0x79	Modifikation der HF-Leistungskonfiguration separat für Schreiboperationen
27	0x7a	Last-RF-Leistungskonfiguration für den Schreibbetrieb
28	0x7b	Modifiziere oder lade maximale Schreib-Wiederholungszeit-Konfigurationen
29	0x7d	Passwortänderung der benutzerdefinierten Tag-Funktion
30	0x7e	Hol dir das Passwort für benutzerdefinierte Tag-Funktionen
31	0x7f	Leseprofil laden/verändern
32	0x85	EM4325 Zeitstempel-Synchronisation
33	0x86	Ermittlung von EM4325-Temperaturdaten
34	0x87	Externe Daten über EM4325 SPI erhalten
35	0x88	Zurücksetzen des EM4325-Alarms
36	0x90	Ändern oder laden Sie DRM-Konfigurationen
37	0x91	Messung des Antennenspiegelverlusts
38	0x92	Messung der Stromtemperatur des Lesers
39	0x50	Schneller Start des Inventars (nur Ex10-Serie)
40	0x51	Stop Quick Inventory (nur Ex10-Serie)

41	0xEA	Erweiterte Parameter setzen (unterstützt nur die Ex10-Serie)
42	0xEB	Erweiterte Parameter herunterladen (unterstützt nur die Ex10-Serie)

## RFID-Tag-Fehlercodes

Fehlercode-Typ	Fehlercode	Fehlercode-Definition	Fehlerbeschreibung
Spezifischer Fehlercode	0x00	Weitere Fehler	Alle anderen vom Tag unterstützten Fehler
0x03	Voll- oder illegaler PC-Wertspeicher	Speicherplatz nicht gefunden oder PC-Wert nicht unterstützt.	
0x04	Gesperrter Speicher	Der Speicher ist vorübergehend/dauerhaft gesperrt, keine Schreiboperation kann durchgeführt werden.	

0x0b	Unzureichende Stromversorgung	Schreiboperationen können aufgrund unzureichender Stromversorgung des Tags nicht durchgeführt werden.	
------	-------------------------------	-------------------------------------------------------------------------------------------------------	--

Fehlercode-Typ	Fehlercode	Fehlercode-Definition	Fehlerbeschreibung
Spezifischer Fehlercode	0x00	Weitere Fehler	Alle anderen Fehler werden vom Tag behandelt.
0x03	Vollspeicher oder illegaler Computerwert	Speicherplatz nicht gefunden oder PC-Wert nicht unterstützt.	
0x04	Gedächtnissperrung	Der Speicher ist vorübergehend/dauerhaft gesperrt, keine Schreiboperation kann durchgeführt werden.	
0x0b	Unzureichende	Schreiboperationen können	

	Stromversorgung	aufgrund unzureichender Stromversorgung für das Tagging nicht durchgeführt werden.	
Unangegebener Fehlercode	0x0f	Undefinierter Fehler	Markiere nicht unterstützte Fehler.

## Tag-Speicher und Zugriffsbefehle

In G2-Tags gibt es vier Speicherbereiche: reservierter Speicher (auch Passwortspeicher genannt), EPC-Speicher, EID-Speicher und Benutzerspeicher.

**Reservierter Speicher:** 4 Bytes, die ersten 2 Bytes sind das Tag-Kill-Passwort und die restlichen 2 Bytes das Zugriffspasswort. Dieser Speicher ist lesbar und wird standardmäßig geschrieben. Die Lese-/Schreibschutzkonfigurationen der beiden Passwortblöcke sind isoliert und können separat modifiziert werden.

**EPC-Speicher:** Die EPC-Nummer des Tags wird in diesem Speicher gespeichert. Das 0-Wort ist der CRC16 PC-Wert und die EPC-Nummer. Word1 ist der PC-Wert und gibt die Länge der EPC-Nummer an. Die tatsächliche EPC-Nummer stammt von Word2. Dieser Speicher ist lesbar und wird standardmäßig geschrieben.

**TID-Speicher:** Dieser Speicher speichert eine vom Hersteller vordefinierte Tag-Kennung. Sie ist standardmäßig lesbar, aber nicht beschreibbar.

**Benutzerspeicher:** Dieser Speicher speichert Benutzerdaten. Standardmäßig kann es gelesen und geschrieben werden.

**HINWEIS:**

- Die Datenlänge ist in vielen G2-Befehlen erforderlich, die Einheit der Datenlänge ist ein Wort oder Byte. Die Länge von 1 Wort entspricht der Länge von 2 Bytes.
- Einige Befehle erfordern möglicherweise ein Zugangspasswort; wenn der Passcode im Ziel-Tag nicht gesetzt ist, füllen Sie das Passwortfeld mit dem Wert 0 aus. Das Passwortfeld darf nicht leer sein.

## Detaillierte Beschreibung des Kommandos

Es gibt zwei Arten von Befehlen:

- protokollbezogene Befehle;
- Befehle, die mit dem Leser zusammenhängen;

Wenn ein ungültiger Befehlsrahmen vom Host geliefert wird, wie zum Beispiel ein nicht erkannter Befehlscode oder ein Befehlsrahmen mit einem CRC-Fehler (die CRC16-Prüfung fehlgeschlagen ist). Der Leser antwortet mit einem ähnlichen Rahmen wie unten gezeigt.

Len	Adr	reCmd	Status	CRC-16	
0x05	0xXX	0x00	0xFE	LSB	0x05

Wenn der Befehlsrahmen falsch gelängt ist, ähnelt die Antwort der folgenden

Len	Adr	reCmd	Status	CRC-16	
0x05	0xXX	0xXX	0xFD	LSB	MSB

Es gibt zwei Situationen, in denen Befehle vom Leser ignoriert werden:

- Ungültige Adresse (weder die Sendeadresse des 0xff noch die entsprechende Leseradresse)
- Unvollständiger Befehlsrahmen, d. h. die tatsächliche Länge des Befehlsrahmens ist kleiner als der im Feld "Len" angegebene Wert. Der Leser wird in den oben beschriebenen Situationen keine Antwort geben.

## EPC C1G2-Befehle

### Inventar-Tags

Der Inventarbefehl wird verwendet, um RFID-Tags mit dem entsprechenden Protokoll abzufragen und Tag-Informationen in einem effektiven Feld zu erhalten. Um weitere Operationen an einem neuen Tag mit einem unbekanntem EPC durchzuführen, sollte der Benutzer zunächst die EPC-Nummer des Tags mit dem Inventarbefehl erhalten.

Je nach Umständen kann der Nutzer die maximale Arbeitszeiteinstellung (Inventarzeit) anpassen. Der Leser gibt das Inventarergebnis innerhalb der angegebenen Zeit zurück. Wenn der Leser den Zählprozess aller Tags im effektiven Feld nicht innerhalb der definierten Zählzeit abschließt, führt er nach Überschreiten der Zählzeit keine weiteren Zählungen mehr durch. Der Leser übermittelt

dem Host Informationen über alle bereits überprüften Tags sowie den aktuellen (unvollständigen) Inventarstatus. Dann wartet der Leser auf den nächsten Befehl vom Host.

Die Standard-Inventarzeit beträgt 2 Sekunden, du kannst sie mit dem benutzerdefinierten Lesebefehl "Inventarzeit des Lesers ändern" ändern. Der korrekte Inventarzeitbereich liegt zwischen 3\*100 ms und 255\*100 ms (die tatsächliche Reaktionszeit kann etwas länger als der eingestellte Wert sein, und der Unterschied

Reichen von 0 bis 75 ms).

**Hinweis:** Wenn die Zählzeit zu kurz ist, kann der Tag das Zählen nicht zählen.

Befehlsrahmen:

Len	Adr	Cmd	Data[]	CRC-16	
0xXX	0xXX	0x01	—	LSB	MSB

Daten[]-Parameter:

Data[]							
QValue	Session	MaskMem	MaskAdr	MaskLen	MaskData	AdrTID	LenTID
0xXX	0xXX	0xXX	2 bytes	0xXX	Variable	0xXX	0xXX

Data[]		
Target	Ant	Scantime
0xXX	0xXX	0xXX

**Parameterdefinition**



QValue: 1 Byte.

bit7: Flagge für statistische Datenpakete.

0 – nach dem Inventar, NICHT das statistische Datenpaket des Inventarprozesses bereitstellen;

1 – Nach dem Inventur stellt das statistische Datenpaket des Inventarprozesses bereit

Bit6: Strategieindikator.

0 – allgemeine Strategie;

1 – Spezialstrategie.

bit5: FastID-Inventarindikator.

0 – Deaktivieren Sie die Impinj FastID-Funktion;

1 – Aktivieren der Impinj FastID-Funktion.

Bit4: Phaseninformation.

0 – aus;

1 – Schalten Sie ein.

Bit3 ~ bit0: der ursprüngliche Q-Wert des EPC-Tag-Inventars, die Q-Werteinstellung sollte der Regel folgen:  $2Q \approx$  die Gesamtzahl der Tags im effektiven Feld. Der Bereich der Q-Werte ist 0~15; wenn in diesem Feld ein weiterer Wert angegeben wird, gibt der Leser den Fehlerstatus des Parameters im Antwortrahmen zurück.

Sitzung: 1 Byte, der Wert der EPC-Tag-Inventarsitzung.

0x00 – S0 als Sitzungswert verwenden;

0x01 – S1 als Sitzungswert verwenden;

0x02 – S2 als Sitzungswert anwenden;

0x03 – S3 als Sitzungswert verwenden;

0xff – Verwenden Sie die Smart-Reader-Konfiguration (gültig nur im EPC-Inventar).

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn ein anderer Wert in diesem Feld angegeben wird.

**MaskMem:** 1 Byte, Anzeige des Maskenbereichs.

0x01 – EPC-Speicher;

0x02 – TID-Speicher;

0x03 – Benutzergedächtnis.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn ein anderer Wert in diesem Feld angegeben wird.

**MaskAdr:** 2 Bytes, die bitweise Eingabeadresse der Maske, der gültige Bereich von MaskAdr ist 0~16383.

**MaskLen:** 1 Byte, Maskenbitlänge (Einheit: Bits).

**MaskData:** Maskendaten. Die Länge von MaskData entspricht MaskLen/8. Ist MaskLen kein Vielfaches einer ganzen Zahl von 8, so ist die Länge von MaskData gleich  $\text{int}[\text{MaskLen}/8]+1$ . Unbestimmte untere signifikante Ziffern sollten mit 0 gefüllt werden.

**AdrTID:** Die Eingabeadresse der TID des Inventarspeichers. Wenn der Befehl diesen Parameter und LenTID im selben Rahmen passierte, sollte der Zählmodus TID und nicht EPC sein.

**LenTID:** Die Datenlänge für TID-Inventaroperationen, der gültige Bereich von LenTID beträgt 0~15. Alle anderen Werte sind reserviert, und der Leser gibt den Fehlerstatus des Parameters im

Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird. Wenn der Befehl diesen Parameter und AdrTID bereitstellt, sollte der Zählmodus TID und nicht EPC sein.

**Ziel (optional):** 1 Byte, EPC-Tag Inventarziel.

0x00 – A als Zielwert verwenden;

0x01 – verwenden Sie B als Zielwert.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn ein anderer Wert in diesem Feld angegeben wird.

**Ameise (optional):** 1 Byte, Antennenauswahl für das aktuelle Inventar.

0x80 – Antenne 1;

0x81 – Antenne 2;

0x82 – Antenne 3;

0x83 – Antenne 4.

Alle anderen Werte sind reserviert, und der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**ScanTime (optional):** 1 Byte, Counting-Time-Konfiguration für die aktuelle Zählung. Der Leser stellt die maximale Laufzeit auf  $\text{ScanTime} * 100$  ms ein.

**HINWEIS:**

1. Wenn die Parameter MaskMem, MaskAdr, MaskLen und MaskData nicht angegeben sind, wird kein Maskenmuster im Inventar enthalten sein

2. Ziel, Ameise und ScanTime sind optionale Parameter, alle drei Parameter müssen in einem Befehlsrahmen bereitgestellt werden. Der Leser wendet die Standardeinstellungen für diese Parameter an, bis alle drei Parameter korrekt vom Host übermittelt wurden, und die neuen Einstellungen gelten nach der Zählung.

3. Die FastID-Funktion gilt nur für den FastID-fähigen Zylinder von Impinj.

(1) Der Antwortstatus ist 0x26, die Antwort sollte wie folgt lauten.

Len	Adr	reCmd	Status	Data[]			CRC-16	
				Ant	ReadRate	TotalCount		
0xXX	0xXX	0x01	0x26	0xXX	2 bytes	4 bytes	LSB	MSB

## Parameterdefinition

Status: Antwortstatus, detaillierte Beschreibung ist in der untenstehenden Tabelle dargestellt

Status	Beschreibung
0x26	Sobald die Zählung abgeschlossen ist, liefern Sie statistische Daten aus dem letzten Zählprozess

**Ameise:** Zeigt an, welche Antenne nach dem Tag gefragt hat. Zum Beispiel, 0x04 im Binärcode 0000 0100 ist, zeigt es an, dass Antenna 3 nach diesem bestimmten Tag gefragt hat. Für 0x08 ist es im Binärcode 0000 1000, was darauf hindeutet, dass Antenna 4 nach Tag-Informationen gefragt hat.

**ReadRate:** Die Identifikationsrate des Tags für das aktuelle Inventar (erfolgreiche Identifikationszeit/Sekunde), mehrfach identifizierte Tag-Authentifizierung erhöht auch die Zeit der erfolgreichen Identifikation.

**TotalCount: Die Gesamtzahl der im aktuellen Inventar entdeckten Tags, ein Tag, der mehrfach abgerufen wurde,** erhöht ebenfalls die Gesamtanzahl.

(2) Der Antwortstatus ist nicht 0x26, sollte die Antwort wie folgt lauten:

Len	Adr	reCmd	Status	Data[]			CRC-16	
				Ant	Num	EPC ID	LSB	MSB
0xXX	0xXX	0x01	0xXX	0xXX	0xXX	EPC-1, EPC-2, EPC-3...	LSB	MSB

## Parameterdefinition

Status: Antwortstatus, detaillierte Beschreibung ist in der untenstehenden Tabelle dargestellt

Status	Beschreibung
0x01	Nach Abschluss der Operation meldet der Leser alle erforderlichen Tag-Informationen an den Host
0x02	Inventarzeit, der Betrieb wird abgebrochen. Der Leser meldet alle bereits angeforderten Tags an den Gastgeber.

0x03	Für Status = 0x03 kann der Leser nicht auf alle Daten in einem einzigen Frame antworten, weitere Daten werden in folgenden Frames übertragen.
0x04	Der Leser hat nur einen Teil des Inventars gemacht, aber der Speicherplatz ging wegen der Anzahl der Tags aus. Der Leser meldet alle bereits geprüften Tags als sowie diesen Statuswert an den Host.
0xF8	Ein Antennenfehler wurde erkannt, die aktuelle Antenne kann getrennt sein.

**Ameise:** Zeigt an, welche Antenne nach dem Tag gefragt hat. Zum Beispiel, 0x04 im Binärcode 0000 0100 ist, zeigt es an, dass Antena 3 nach diesem bestimmten Tag gefragt hat. Für 0x08 ist es im Binärcode 0000 1000, was darauf hindeutet, dass Antena 4 nach Tag-Informationen gefragt hat.

**Num:** Die Menge an EPC/TID-RFID, die in dieser Antwort enthalten ist.  
**EPC-ID:** EPC/TID-Daten abgefragt. Zum Beispiel ist EPC-1 die EPC/TID-Länge + EPC/TID-Nummer, Daten + RSSI-Wert des ersten Tags usw. Die RSSI- und EPC-Längenwerte sind Einzelbyte-Parameter. Die EPC C1 G2 Dateneinheit ist das Wort. Beim Kommunizieren werden zuerst das bedeutungsvollste Wort und das bedeutendste Byte im Wort übertragen.

Wenn FastID aktiviert ist, folgt jeder EPC-ID-Datenblock diesem Format:

Gesamtlänge EPC plus TID (1 Byte) + Tag EPC-Nummer + 12 Bytes  
TID-Daten + Tag RSSI-Wert

Das bedeutendste Bit der Gesamtlänge von EPC und TID gibt an, ob ein TID in diesem Block enthalten ist.

### Detaillierte EPC-ID-Struktur

EPC-ID				
EPC-1				
Datenlänge	Daten	RSSI	Phase	Frequenz (kHz)
1 Bit	N Bits	1 Bit	4 Bit	3 Bit
Bit7: 0 – der Datenblock enthält EPC oder TID; 1 – Datenblock enthält EPC plus TID (FastID aktiviert). Bit6: 0 – Phase Off und Frequenz; 1 – Phase und Frequenz aktiviert. Bit 5 ~ Bit 0: Datenlänge N				

**HINWEIS:** Die FastID-Funktion funktioniert nur für Impinj-Einsätze, die FastID unterstützen.

### Lesedaten

Die Aufgabe dieses Befehls ist es, Daten aus dem reservierten Speicher, EPC-Speicher, TID-Speicher und Tag-Benutzerspeicher auszulesen. Die Leseoperation beginnt an der definierten Adresse. Die Dateneinheit ist das Wort.

### Befehlsrahmen:

Len	Adr	Cmd	Data[]	CRC-16	
0xXX	0xXX	0x02	—	LSB	MSB

### Daten[]-Parameter:

Data[]									
ENum	EPC	Mem	WordPtr	Num	Pwd	MaskMem	MaskAdr	MaskLen	MaskData
0xXX	variable	0xXX	0xXX	0xXX	4 bytes	0xXX	2 bytes	0xXX	variable

## Parameterdefinition

### ENum:

0x00 ~ 0x0f – ENum steht für die Länge der EPC-Nummer. Die ENum-Einheit ist das Wort, und die EPC-Länge sollte weniger als 15 Wörter betragen. In dieser Situation sollten die Parameter MaskMem, MaskAdr, MaskLen und MaskData leer gelassen werden.

0xff – anstelle von EPC-Parametern sollten die Parameter MaskMem, MaskAdr, MaskLen und MaskData angegeben werden. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn ein anderer Wert im Feld angegeben ist.

**EPC:** Die EPC-Nummer des Ziel-Tags. Die Länge des EPC hängt von der angegebenen EPC-Nummer ab, und der Längenwert muss eine ganze Zahl sein. Die Einheit EPC-Nummer ist das Wort. Während der Kommunikation werden zuerst das bedeutendste Wort und das bedeutendste Byte im Wort eingereicht. Es ist notwendig, in diesem Befehl das VOLLE EPC anzugeben.

**Meme:** 1 Byte, Zielspeichertyp.

0x00 – reservierter Speicher

0x01 – EPC-Speicher;

0x02 – TID-Speicher;

0x03 – Benutzergedächtnis.

Alle anderen Werte sind reserviert. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben ist.

**WordPtr:** 1 Byte, die Anfangsadresse des Zielworts. Zum Beispiel bedeutet 0x00, dass die Leseoperation mit dem ersten Wort (dem ersten 16-Bit-Speicherbereich) beginnt, 0x01 bedeutet, dass die Operation mit 2 beginnt. Worte usw.

**Num:** 1 Byte, die Anzahl der Wörter, die in dieser Operation gelesen werden. Daher ist 0x00 kein gültiger Wert für diesen Parameter. Außerdem sollte der Num-Wert 120 nicht überschreiten, d. h. die maximale Lesezahl beträgt 120 Wörter. Der Leser gibt den

Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein Wert von 0 oder größer als 120 angegeben ist.

**Pwd:** 4 Bytes, dies ist ein 4-Byte-Tag-Zugriffspasswort. Von links nach rechts sind die ersten zwei Bytes die hohen Bytes des Zugriffspassworts, und die letzten zwei Bytes sind die niedrigen Bytes. Das bedeutendste Bit des 32-Bit-Tag-Zugriffspassworts ist das bedeutendste Bit im 1. Byte von Pwd (von links nach rechts).

Ebenso ist das am wenigsten signifikante Bit eines 32-Bit-Tag-Zugriffspassworts das am wenigsten signifikante Bit im vierten (letzten) Pwd-Byte. Der Passcode wird zur Verifizierung während der Leseoperationen des reservierten Speichers verwendet, wenn der zugehörige Zielspeicher passwortgeschützt ist und das Tag-Zugriffspasswort nicht null ist. Andernfalls sollte das PSW auf Null oder auf ein gültiges Zugangspasswort gesetzt werden.

**MaskMem:** 1 Byte, Anzeige des Maskenbereichs.

0x01 – EPC-Speicher;

0x02 – TID-Speicher;

0x03 – Benutzergedächtnis.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn ein anderer Wert in diesem Feld angegeben wird.

**MaskAdr:** 2 Bytes, die bitweise Eingabeadresse der Maske, der gültige Bereich von MaskAdr ist 0~16383.

**MaskLen:** 1 Byte, Maskenbitlänge (Einheit: Bits).

**MaskData:** Maskendaten. Die Länge von MaskData entspricht MaskLen/8. Ist MaskLen kein Vielfaches einer ganzen Zahl von 8, so ist

die Länge von MaskData gleich  $\text{int}[\text{MaskLen}/8]+1$ . Unbestimmte untere signifikante Ziffern sollten mit 0 gefüllt werden.

**HINWEIS:**

Wenn keine MaskMem, MaskAdr, MaskLen und MaskData bereitgestellt werden, wird die vollständige EPC-Nummer als Maskenvorlage im Katalog verwendet.

**Antwortrahmen:**

Len	Adr	reCmd	Status	Data[]	CRC-16	
0xXX	0xXX	0x02	0x00	Word1, Word2,...	LSB	0xXX

**Parameterdefinition**

Word1, Word2...: Die Dateneinheit in diesem Feld ist ein Wort. Jedes Wort enthält 2 Bytes, das bedeutendste Byte zuerst. Word1 ist das erste Wort aus der Startadresse, das zweite Wort ist Word 2, und so weiter.

Datenaufzeichnung

Die Funktion dieses Befehls besteht darin, mehrere Wörter in einem einzigen Kommunikationszyklus in den reservierten Speicher, TID-Speicher oder Benutzerspeicher zu schreiben.

**Befehlsrahmen:**

Len	Adr	Cmd	Data[]	CRC-16	
0xXX	0xXX	0x03	—	LSB	MSB

**Daten[]-Parameter:**

Data[]					
WNum	ENum	EPC	Mem	WordPtr	Wdt
0xXX	0xXX	variable	0xXX	0xXX	variable
Pwd	MaskMem	MaskAdr	MaskLen	MaskData	

4 bytes	0xXX	2 bytes	0xXX	variable
---------	------	---------	------	----------

### Parameterdefinition

**WNum:** die Anzahl der zu schreibenden Wörter. Jedes Wort enthält 2 Bytes. Der WNum-Wert muss größer als 0 und kleiner als oder gleich 32 sein sowie identisch mit der tatsächlichen Menge des zu schreibenden Wortes. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein Wert von 0 oder ein ungültiger Wert des WNum-Parameters angegeben ist.

### ENum:

0x00 ~ 0x0f – ENum steht für die Länge der EPC-Nummer. Die ENum-Einheit ist das Wort, und die EPC-Länge sollte weniger als 15 Wörter betragen. In dieser Situation sollten die Parameter MaskMem, MaskAdr, MaskLen und MaskData leer gelassen werden.

0xff – anstelle von EPC-Parametern sollten die Parameter MaskMem, MaskAdr, MaskLen und MaskData angegeben werden. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn ein anderer Wert im Feld angegeben ist.

**EPC:** Die EPC-Nummer des Ziel-Tags. Die Länge des EPC hängt von der angegebenen EPC-Nummer ab. Die Einheit EPC-Nummer ist das Wort. Beim Kommunizieren werden zuerst das bedeutungsvollste

Wort und das bedeutendste Byte im Wort übertragen. Es ist unerlässlich, in diesem Befehl die VOLLE EPC-Nummer anzugeben.

**Meme:** 1 Byte, Zielspeichertyp.

0x00 – reservierter Speicher

0x01 – EPC-Speicher;

0x02 – TID-Speicher;

0x03 – Benutzergedächtnis.

Alle anderen Werte sind reserviert. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben ist.

**WordPtr:** 1 Byte, die Anfangsadresse des Zielworts.

**Wdt:** Wörter, die im Speicher des Tags gespeichert werden. Das Wortvolumen in Wdt muss identisch mit dem WNum-Wert sein. Das bedeutendste Byte ist das erste in einem Wort. Wörter werden gemäß der Reihenfolge in Date[] von einer niedrigeren zu einer höheren Adresse auf das Tag geschrieben. Zum Beispiel hat WordPtr den Wert 0x02, das erste Wort

in Date[] (von links nach rechts) wird an der Adresse 0x02 im vom Meme angegebenen Speicher geschrieben, das zweite Wort wird an der Adresse 0x03 geschrieben, und so weiter.

**Pwd:** 4 Bytes, dies ist ein 4-Byte-Tag-Zugriffspasswort. Von links nach rechts sind die ersten zwei Bytes die hohen Bytes des Zugriffspassworts, und die letzten zwei Bytes sind die niedrigen Bytes. Das bedeutendste Bit des 32-Bit-Tag-Zugriffspassworts ist das bedeutendste Bit im 1. Byte von Pwd (von links nach rechts). Ebenso ist das am wenigsten signifikante Bit eines 32-Bit-Tag-Zugriffspassworts das am wenigsten signifikante Bit in

Das Zugangspasswort muss korrekt angegeben werden und vor den Schreibvorgängen validiert werden. Wenn der Zielspeicher nicht passwortgeschützt ist, kann der Pwd auf null gesetzt werden.

**MaskMem:** 1 Byte, Anzeige des Maskenbereichs.

0x01 – EPC-Speicher;

0x02 – TID-Speicher;

0x03 – Benutzergedächtnis.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn ein anderer Wert in diesem Feld angegeben wird.

**MaskAdr:** 2 Bytes, die bitweise Eingabeadresse der Maske, der gültige Bereich von MaskAdr ist 0~16383.

**MaskLen:** 1 Byte, Maskenbitlänge (Einheit: Bits).

**MaskData:** Maskendaten. Die Länge von MaskData entspricht MaskLen/8. Ist MaskLen kein Vielfaches einer ganzen Zahl von 8, so ist die Länge von MaskData gleich  $\text{int}[\text{MaskLen}/8]+1$ . Unbestimmte untere signifikante Ziffern sollten mit 0 gefüllt werden.

#### **HINWEIS:**

Wenn MaskMem, MaskAdr, MaskLen und MaskData nicht bereitgestellt werden, wird die vollständige EPC-Nummer als Maskenmeister im Inventar verwendet.

#### **Antwortrahmen:**

Len	Adr	reCmd	Status	Data[]	CRC-16	
0x05	0xXX	0x03	0x00	—	LSB	MSB

## Eingabe der EPC-Nummer

Die Funktion dieses Befehls besteht darin, die EPC-Nummer auf das Tag zu schreiben. Während dieses EPC-Schreibvorgangs kann nur ein Tag im effektiven Bereich der Antenne platziert werden.

Len	Adr	Cmd	Data[]			CRC-16	
			ENum	Pwd	WEPC		
0xXX	0xXX	0x04	0xXX	4 bytes	variable	LSB	MSB

### Parameterdefinition

**ENum:** 1 Byte, Länge der EPC-Nummer. Die ENum-Einheit ist ein Wort, und die Länge des EPC kann 0 sein, aber es sollte weniger als 15 Wörter sein. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben ist.

**Pwd:** 4 Bytes, dies ist ein 4-Byte-Tag-Zugriffspasswort. Von links nach rechts sind die ersten zwei Bytes die hohen Bytes des Zugriffspassworts, und die letzten zwei Bytes sind die niedrigen Bytes. Das bedeutendste Bit des 32-Bit-Tag-Zugriffspassworts ist das bedeutendste Bit im 1. Byte von Pwd (von links nach rechts). Ebenso ist das am wenigsten signifikante Bit eines 32-Bit-Tag-Zugriffspassworts das am wenigsten signifikante Bit im vierten (letzten) Pwd-Byte. Der Passcode wird zur Verifizierung während der Leseoperationen des reservierten Speichers verwendet, wenn der zugehörige Zielspeicher passwortgeschützt ist und das Tag-Zugriffspasswort nicht null ist. Andernfalls sollte das PSW auf Null oder auf ein gültiges Zugangspasswort gesetzt werden.

**WEPC:** Die EPC-Nummer, die im Tag gespeichert wird. Die Wortanzahl in WEPC muss identisch mit dem ENum-Wert sein. Die Mindest- und Maximallänge von WEPC beträgt 1 und 31. Der Leser gibt den

Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld Daten der falschen Länge angegeben sind.

Len	Adr	reCmd	Status	Data[]	CRC-16	
0x05	0xXX	0x04	0x00	—	LSB	MSB

Lese-/SchreibschutzEinstellung für einen bestimmten Speicher

Die Funktion dieses Befehls besteht darin, den Lese-/Schreibschutzzustand für die folgenden Speicher festzulegen

- Reservierter Speicher  
Lesbar/ungeschützt, dauerhaft lesbar/beschreibbar, passwortgeschützt/lesbar/beschreibbar oder dauerhaft unlesbar/unbeschreibbar;
- EPC/benutzerreservierter Speicher  
Ungeschützt, dauerhaft schreibbar, passwortgeschützt oder dauerhaft unbeschreibbar;

EPC-Speicher, benutzerreservierter Speicher und TID-Speicher sind dauerhaft lesbar. Außerdem ist der TID-Speicher nur lesegeschützt.

Wenn der Schutzzustand des reservierten Speichers auf dauerhaft lesbar/beschreibbar oder dauerhaft unlesbar/unbeschreibbar gesetzt wird, kann die Zustandskonfiguration keine zukünftigen Änderungen mehr akzeptieren. Ebenso kann die Statuskonfiguration, wenn der Schutzstatus von EPC-Speicher, benutzerreserviertem Speicher oder TID-Speicher auf dauerhaft beschreibbar/nicht beschreibbar gesetzt wird, keine zukünftigen Änderungen mehr akzeptieren. Das Tag gibt den Fehlerstatus im Antwortrahmen zurück, wenn der Benutzer versucht, eine zukünftige Änderung vorzunehmen. Um einen bestimmten Speicher als passwortgeschütztes Lesen/Schreiben zu setzen oder den Status von passwortgeschützt auf ungeschützt zu ändern, müssen Sie ein

Zugriffspasswort eingeben. Daher sollten Sie vor dieser Operation sicherstellen, dass das Zugangspasswort für das RFID-Tag bereits konfiguriert ist.

Len	Adr	Cmd	Data[]	CRC-16	
0xXX	0xXX	0x06	—	LSB	MSB

Data[]								
ENum	EPC	Select	SetProtect	Pwd	MaskMem	MaskAdr	MaskLen	MaskData
0xXX	variable	0xXX	0xXX	4 bytes	0xXX	2 bytes	0xXX	variable

## Parameterdefinition

### **ENum:**

0x00~0x0f – ENum steht für die Länge der EPC-Nummer. Die ENum-Einheit ist das Wort, und die EPC-Länge sollte weniger als 15 Wörter betragen. In dieser Situation sollten die Parameter MaskMem, MaskAdr, MaskLen und MaskData leer gelassen werden.

0xff – Anstelle von EPC-Parametern sollten MaskMem, MaskAdr, MaskLen und MaskData angegeben werden. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben ist.

**EPC:** Die EPC-Nummer des Ziel-Tags. Die Dauer des Energieeffizienzcertifikats hängt von der angegebenen EPC-Nummer ab. Die Einheit EPC-Nummer ist das Wort. Beim Kommunizieren werden zuerst das bedeutendste Wort und das bedeutendste Byte im Wort übertragen. Es ist unerlässlich, in diesem Kommando das VOLLSTÄNDIGE EPC bereitzustellen.

**Auswählen:** 1 Byte.

0x00 – Änderung des Kill-Passwort-Lese-/Schreibschutzes;

0x01 – Änderung der Sicherheit gegen Lesen/Schreiben des Zugriffspassworts;

0x02 – Änderung des EPC-

Speicherschutzes gegen

Lesen/Schreiben; 0x03 – Änderung

des TID-Speicherschutzes gegen

Lesen/Schreiben; 0x04 – Änderung

des Speicherschutzes des Benutzers

gegen Lesen/Schreiben.

Alle anderen Werte sind reserviert, der Leser führt den Befehl nicht aus und gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn ein anderer Wert in diesem Feld angegeben wird.

**SetProtect:** Der Wert von SetProtect hängt vom Wert von Select ab.

- Für Select = 0x00 oder 0x01, also die Kill-Passwort- oder Zugriffspasswortschutz-Einstellung, kann der SetProtect-Wert wie folgt aussehen:

0x00 – auf lesbar/beschreibbar ohne Schutz gesetzt;

0x01 – auf dauerhaft lesbar/schreibbar gesetzt;

0x02 – auf lesbar/beschreibbar mit

Passwortschutz gesetzt; 0x03 – auf

dauerhaft unlesbar/unbeschreibbar

gesetzt.

- Für Select = 0x02, 0x03, 0x04, also EPC-, TID- und Benutzerspeicherschutzeinstellungen, kann der SetProtect-Wert wie folgt lauten:

0x00 – auf beschreibbar ohne Sicherheit gesetzt;

0x01 – auf dauerhaft beschreibbar gestellt;

0x02 – auf beschreibbar mit

Passwortschutz gesetzt; 0x03 –

Machen Sie es dauerhaft  
unaufzeichnungsfähig.

Alle anderen Select- und SetProtect-Werte sind reserviert, der Leser führt den Befehl nicht aus und gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn ein anderer Wert in diesem Feld angegeben wird.

**Pwd:** 4 Bytes, dies ist ein 4-Byte-Tag-Zugriffspasswort. Von links nach rechts sind die ersten zwei Bytes die oberste Anzahl von Bytes des Passcodes, und die letzten zwei Bytes sind die unteren Bytes. Das bedeutendste Bit eines 32-Bit-Tag-Zugriffspassworts ist das bedeutendste Bit in Pwd 1 Byte (von links nach rechts). Ebenso ist das am wenigsten signifikante Bit des 32-Bit-Tag-Passcodes das am wenigsten signifikante Bit im vierten (letzten) Pwd-Byte.

**MaskMem:** 1 Byte, Anzeige des Maskenbereichs.

0x01 – EPC-Speicher;

0x02 – TID-Speicher;

0x03 – Benutzergedächtnis.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**MaskAdr:** 2 Bytes, Masken-Eingangsbiteadresse, der gültige Bereich von MaskAdr ist 0~16383.

**MaskLen:** 1 Byte, Maskenbitlänge (Einheit: Bits).

**MaskData:** Maskendaten. Die Länge von MaskData entspricht MaskLen/8. Ist MaskLen kein Vielfaches einer ganzen Zahl von 8, so ist die Länge von MaskData gleich  $\text{int}[\text{MaskLen}/8]+1$ . Unbestimmte untere signifikante Ziffern sollten mit der Ziffer 0 gefüllt werden.

**HINWEIS:**

Wenn MaskMem, MaskAdr, MaskLen und MaskData nicht bereitgestellt werden, wird die vollständige EPC-Nummer als Maskenmuster im Inventar verwendet.

**Antwortrahmen:**

Nur	Adr	reCmd	Zustand	Daten[]	CRC-16-Relais	
0x05	0xXX	0x06	0x00	--	LSB	MSB

**Blöcke löschen**

Die Funktion dieses Befehls besteht darin, mehrere Wörter aus dem reservierten Speicher, EPC, TID oder Benutzerspeicher zu löschen.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0xXX	0xXX	0x07	--	LSB	MSB

**Datenparameter[]:**

Daten[]							
EEnum	Energieleistungszertifikat	Meme	WordPtr (Word)	Num	Pwd	MaskMem	M
0xXX	Variable	0xXX	0xXX	0xXX	4 Bytes	0xXX	

**EEnum:**

0x00~0x0f – EEnum steht für die Länge der EPC-Nummer. Die EEnum-Einheit ist das Wort, und die EPC-Länge sollte weniger als 15 Wörter betragen. In dieser Situation sollten die Parameter MaskMen, MaskAdr, MaskLen und MaskData leer gelassen werden.

0xff – Anstelle von EPC-Parametern sollten MaskMem, MaskAdr, MaskLen und MaskData angegeben werden. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben ist.

**EPC:** Die EPC-Nummer des Ziel-Tags. Die Dauer des Energieeffizienzcertifikats hängt von der angegebenen EPC-Nummer ab. Die Einheit EPC-Nummer ist das Wort. Beim Kommunizieren werden zuerst das bedeutendste Wort und das bedeutendste Byte im Wort übertragen. Es ist unerlässlich, in diesem Kommando das VOLLSTÄNDIGE EPC bereitzustellen.

**Meme:** 1 Byte, Zielspeichertyp.

0x00 – reserviertes Gedächtnis

0x01 – EPC-Speicher;

0x02 – TID-

Speicher; 0x03 –

Benutzergedächt

nis.

Alle anderen Werte sind reserviert. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben ist.

**WordPtr:** 1 Byte, die Anfangsadresse des Zielworts. Zum Beispiel stellt 0x01 eine Löschoption dar, die mit dem zweiten Wort beginnt (dem zweiten 16-Bit-Speicherbereich) usw. Für EPC-Löschoptionen sollte der Mindest-WordPtr 0x01 sein. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein falscher Wert angegeben wird.

**Anzahl:** 1 Byte, die Anzahl der Wörter, die in dieser Operation gelöscht werden sollen. Das Löschen beginnt mit den in WordPtr angegebenen Adressen und löscht die in Num angegebene Anzahl der Wörter. 0x00 ist kein gültiger Wert für diesen Parameter. Der Leser gibt den Fehlerzustand des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein Wert von 0 oder größer als 120 angegeben wird.

**Pwd:** 4 Bytes, dies ist ein 4-Byte-Tag-Zugriffspasswort. Von links nach rechts sind die ersten zwei Bytes die oberste Anzahl von Bytes des Passcodes, und die letzten zwei Bytes sind die unteren Bytes. Das bedeutendste Bit eines 32-Bit-Tag-Zugriffspassworts ist das bedeutendste Bit in Pwd 1 Byte (von links nach rechts). Ebenso ist das am wenigsten signifikante Bit des 32-Bit-Passcodes eines Tags das am wenigsten signifikante Bit im vierten (letzten) Pwd-Byte. PSW sollte auf null gestellt oder auf das korrekte Zugriffspasswort gestellt werden.

**MaskMem:** 1 Byte, Anzeige des Maskenbereichs.

0x01 – EPC-Speicher;

0x02 – TID-Speicher;

0x03 – Benutzergedächtnis.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**MaskAdr:** 2 Bytes, Masken-Eingangsbitaladresse, der gültige Bereich von MaskAdr ist 0~16383.

**MaskLen:** 1 Byte, Maskenbitlänge (Einheit: Bits).

**MaskData:** Maskendaten. Die Länge von MaskData entspricht  $\text{MaskLen}/8$ . Ist MaskLen kein Vielfaches einer ganzen Zahl von 8, so ist die Länge von MaskData gleich  $\text{int}[\text{MaskLen}/8]+1$ . Unbestimmte untere signifikante Ziffern sollten mit der Ziffer 0 gefüllt werden.

#### **HINWEIS:**

Wenn MaskMem, MaskAdr, MaskLen und MaskData nicht bereitgestellt werden, wird die vollständige EPC-Nummer als Maskenmuster im Inventar verwendet.

**Antwortrahmen:**



Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0x05	0xXX	0x07	0x00	--	LSB	MSB

Leseschutzkonfiguration (gemäß EPC-Nummer)

Die Funktion dieses Befehls besteht darin, den Leseschutz für ein Tag mit einer bestimmten EPC-Nummer einzustellen, sodass das Tag für das Gerät unlesbar ist. Das Gerät kann die EPC-Nummer dieses Tags mit der Zähloperation nicht abfragen. Dieser Befehl gilt nur für **NXP UCODE EPC G2X-Tags**.

#### Befehlsrahmen:

Nur	Adr	Cmd	Daten[]	CRC-16- Relais	
0xXX	0xXX	0x08	--	LSB	MSB

#### Datenparameter[]:

Daten[]						
ENum	Energieleistungsz ertifikat	Pw d	MaskM em	Mask Adr	MaskL en	Maskend aten
0xX X	Variable	4 Byt es	0xXX	2 Bytes	0xXX	Variable

#### ENum:

0x00~0x0f – ENum steht für die Länge der EPC-Nummer. Die ENum-Einheit ist das Wort, und die EPC-Länge sollte weniger als 15 Wörter betragen. In dieser Situation sollten die Parameter MaskMem, MaskAdr, MaskLen und MaskData leer gelassen werden.

0xff – Anstelle von EPC-Parametern sollten MaskMem, MaskAdr, MaskLen und MaskData angegeben werden.

Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben ist.

**EPC:** Die EPC-Nummer des Ziel-Tags. Die Dauer des Energieeffizienzcertifikats hängt von der angegebenen EPC-Nummer ab. Die Einheit EPC-Nummer ist das Wort. Beim Kommunizieren werden zuerst das bedeutendste Wort und das bedeutendste Byte im Wort übertragen. Es ist unerlässlich, in diesem Kommando das VOLLSTÄNDIGE EPC bereitzustellen.

**Pwd:** 4 Bytes, dies ist ein 4-Byte-Tag-Zugriffspasswort. Von links nach rechts sind die ersten zwei Bytes die oberste Anzahl von Bytes des Passcodes, und die letzten zwei Bytes sind die unteren Bytes. Das bedeutendste Bit eines 32-Bit-Tag-Zugriffspassworts ist das bedeutendste Bit in Pwd 1 Byte (von links nach rechts). Ebenso ist das am wenigsten signifikante Bit des 32-Bit-Passcodes eines tags das am wenigsten signifikante Bit im 4. (letzten) Pwd-Byte. Es ist nicht möglich, einen Lese-Schutz-Status festzulegen, wenn der Tag einen nullbasierten Passcode hat, daher gibt der Leser den Parameterfehlerstatus im Antwortrahmen zurück, wenn in diesem Feld Nullen oder ein falscher Passcode angegeben werden.

**MaskMem:** 1 Byte, Anzeige des Maskenbereichs.

0x01 – EPC-Speicher;

0x02 – TID-Speicher;

0x03 – Benutzergedächtnis.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**MaskAdr:** 2 Bytes, Masken-Eingangsbitadresse, der gültige Bereich von MaskAdr ist 0~16383

**MaskLen:** 1 Byte, Maskenbitlänge (Einheit: Bits).

**MaskData:** Maskendaten. Die Länge von MaskData entspricht MaskLen/8. Ist MaskLen kein Vielfaches einer ganzen Zahl von 8, so ist die Länge von MaskData gleich  $\text{int}[\text{MaskLen}/8]+1$ . Unbestimmte untere signifikante Ziffern sollten mit der Ziffer 0 gefüllt werden.

**HINWEIS:**

Wenn MaskMem, MaskAdr, MaskLen und MaskData nicht bereitgestellt werden, wird die vollständige EPC-Nummer als Maskenmuster im Inventar verwendet.

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0x05	0xXX	0x08	0x00	---	LSB	MSB

Leseschutzkonfiguration (ohne EPC-Nummer)

Die Funktion dieses Befehls besteht darin, den Leseschutz für Tags in einem effektiven Feld zu konfigurieren. Im Gegensatz zum vorherigen Befehl in 8.2.8 führt dieser Befehl eine Operation an mehreren abgefragten Tags aus, ohne den Tag zu identifizieren. Um eine Operation an mehreren Tags durchzuführen, ist es wichtig, dass die Passphrase mit diesen Tags übereinstimmt. Dieser Befehl gilt nur für **NXP UCODE EPC G2X-Tags**.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16- Relais	
			Pwd		
0x08	0xXX	0x09	4 Bytes	LSB	MSB

## Parameterdefinition

**Pwd:** 4 Bytes, dies ist ein 4-Byte-Tag-Zugriffspasswort. Von links nach rechts sind die ersten zwei Bytes die oberste Anzahl von Bytes des Passcodes, und die letzten zwei Bytes sind die unteren Bytes. Das bedeutendste Bit eines 32-Bit-Tag-Zugriffspassworts ist das bedeutendste Bit in Pwd 1 Byte (von links nach rechts). Ebenso ist das am wenigsten signifikante Bit des 32-Bit-Passcodes eines tags das am wenigsten signifikante Bit im 4. (letzten) Pwd-Byte. Es ist nicht möglich, einen Lese-Schutz-Status festzulegen, wenn der Tag einen nullbasierten Passcode hat, daher gibt der Leser den Parameterfehlerstatus im Antwortrahmen zurück, wenn in diesem Feld Nullen oder ein falscher Passcode angegeben werden.

### Antwortrahmen:

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0x05	0xXX	0x09	0x00	--	LSB	MSB

## Entsperren Leseschutz

Die Funktion dieses Befehls ist es, den Leseschutz des Tags zu entsperren. Im Einsatzbereich der Antenne darf nur ein Tag angebracht werden. Dieser Befehl gilt nur für **NXP UCODE EPC G2X-Tags**.

### Befehlsrahmen:

Nur	Adr	Cmd	Daten[]	CRC-16- Relais	
			Pwd		
0x08	0xXX	0x0a	4 Bytes	LSB	MSB

## Parameterdefinition

**Pwd:** 4 Bytes, dies ist ein 4-Byte-Tag-Zugriffspasswort. Von links nach rechts sind die ersten zwei Bytes die oberste Anzahl von Bytes

des Passcodes, und die letzten zwei Bytes sind die unteren Bytes. Das bedeutendste Bit eines 32-Bit-Tag-Zugriffspassworts ist das bedeutendste Bit in Pwd 1 Byte (von links nach rechts). Ebenso ist das am wenigsten signifikante Bit des 32-Bit-Tag-Passcodes das am wenigsten signifikante Bit im 4. (letzten) Pwd-Byte. Es ist essenziell, in diesem Befehl den korrekten Passcode anzugeben; der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein ungültiger Passcode angegeben wird.

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0x05	0xXX	0x0a	0x00	--	LSB	MSB

**HINWEIS:**

Bei Tags ohne Leseschutz wird der Standardzustand freigeschaltet.

Lesen Sie die Schutzstatusprüfung

Dieser Befehl ist NICHT in der Lage, festzustellen, ob ein bestimmtes Tag Leseschutz unterstützt, der Befehl bezieht sich NUR auf den Leseschutzstatus des Tags. Bei Tags ohne Leseschutz wird der Standardzustand freigeschaltet. Im Einsatzbereich der Antenne darf nur ein Tag angebracht werden. Dieser Befehl gilt nur für **NXP UCODE EPC G2X-Tags**.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16- Relais	
0x04	0xXX	0x0b	--	LSB	MSB

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	

0x06	0xXX	0x0b	0x00	ReadPro Reader	LSB	MSB
------	------	------	------	-------------------	-----	-----

Parameterdefinition

**ReadPro:** 0x00 – der Leseschutz ist für den Tag deaktiviert; 0x01 – der Leseschutz ist für den Tag aktiviert.

Bei Tags ohne Leseschutz wird der Standardzustand freigeschaltet.

EAS-Konfiguration

Die Funktion dieses Befehls besteht darin, den EAS-Zustand zu ändern oder zurückzusetzen. Dieser Befehl gilt nur für **NXP UCODE EPC G2**-Tags.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0xXX	0xXX	0x0c	--	LSB	MSB

**Datenparameter[]:**

Date n[]							
ENum	Energieleistungszertifikat	Powerd	EAS	MaskMem	MaskAdr	MaskLen	Maskendaten
0xXX	Variable	4 Bytes	0xXX	0xXX	2 Bytes	0xXX	Variable

**ENum:**

0x00~0x0f – ENum steht für die Länge der EPC-Nummer. Die ENum-Einheit ist das Wort, und die EPC-Länge sollte weniger als 15 Wörter betragen. In dieser Situation sollten die Parameter MaskMen, MaskAdr, MaskLen und MaskData leer gelassen werden.

0xff – Anstelle von EPC-Parametern sollten MaskMem, MaskAdr, MaskLen und MaskData angegeben werden. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben ist.

**EPC:** Die EPC-Nummer des Ziel-Tags. Die Dauer des Energieeffizienzcertifikats hängt von der angegebenen EPC-Nummer ab. Die Einheit EPC-Nummer ist das Wort. Beim Kommunizieren werden zuerst das bedeutendste Wort und das bedeutendste Byte im Wort übertragen. Es ist unerlässlich, in diesem Kommando das VOLLSTÄNDIGE EPC bereitzustellen.

**Pwd:** 4 Bytes, dies ist ein 4-Byte-Tag-Zugriffspasswort. Von links nach rechts sind die ersten zwei Bytes die oberste Anzahl von Bytes des Passcodes, und die letzten zwei Bytes sind die unteren Bytes. Das bedeutendste Bit eines 32-Bit-Tag-Zugriffspassworts ist das bedeutendste Bit in Pwd 1 Byte (von links nach rechts). Ebenso ist das am wenigsten signifikante Bit des 32-Bit-Passcodes eines Tags das am wenigsten signifikante Bit im vierten (letzten) Pwd-Byte. Es ist nicht möglich, EAS zu setzen, wenn der Tag einen Null-Passcode hat, daher gibt der Leser den Fehlerstatusparameter im Antwortrahmen zurück, wenn in diesem Feld Nullen oder ein ungültiger Passcode angegeben sind.

**EAS:** 1 Byte.

bit0 = 0, ist der EAS-

Alarm deaktiviert; bit0

= 1, ist die EAS-

Warnung aktiviert.

bit1 ~ bit7 sind reserviert, mit 0 als Standard.

**MaskMem:** 1 Byte, Anzeige des Maskenbereichs.

0x01 – EPC-Speicher;

0x02 – TID-Speicher;

0x03 – Benutzergedächtnis.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**MaskAdr:** 2 Bytes, Masken-Eingangsbitadresse, der gültige Bereich von MaskAdr ist 0~16383.

**MaskLen:** 1 Byte, Maskenbitlänge (Einheit: Bits).

**MaskData:** Maskendaten. Die Länge von MaskData entspricht  $\text{MaskLen}/8$ . Ist MaskLen kein Vielfaches einer ganzen Zahl von 8, so ist die Länge von MaskData gleich  $\text{int}[\text{MaskLen}/8]+1$ . Unbestimmte untere signifikante Ziffern sollten mit der Ziffer 0 gefüllt werden.

## HINWEIS

Wenn MaskMem, MaskAdr, MaskLen und MaskData nicht bereitgestellt werden, wird die vollständige EPC-Nummer als Maskenmuster im Inventar verwendet.

## Antwortrahmen:

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0xXX	0xXX	0x0c	0x00	--	LSB	MSB

## EAS-Alarmerkennung

Die Funktion dieses Befehls ist es, einen EAS-Alarm zu erkennen, der nur für **NXP UCODE EPC G2-Tags** gilt.

### Befehlsrahmen:

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0x04	0xXX	0x0d	--	LSB	MSB

### Antwortrahmen:

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16-Relais	
0x05	0xXX	0x0d	0x00	--	LSB	MSB

Gibt "kein funktionierendes Tag" zurück, wenn die EAS-Warnung nicht erkannt wird.

## Einzel-Tag-Inventar

### Befehlsrahmen:

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0x04	0xXX	0x0f	--	LSB	MSB

### Antwortrahmen:

Nur	Adr	reCmd sagte:	Zustand	Daten[]			CRC-16-Relais
				Ameis e	Nu m	EPC-ID	

0xX X	0xX X	0x0f	0x01	0xXX	0x01	EPC-1 (EPC-1 Version )	LS B	MS B
----------	----------	------	------	------	------	---------------------------------	---------	---------

### Parameterdefinition

**Ameise:** Zeigt an, welche Antenne nach dem Tag gefragt hat. Zum Beispiel, 0x04 im Binärcode 0000 0100 ist, bedeutet das, dass Antenne 3 nach diesem speziellen Tag gefragt hat. Für 0x08 ist es im Binärcode 0000 1000, was bedeutet, dass Antenna 4 nach Tag-Informationen gefragt hat.

**Anzahl:** 0x01 (fest)

**EPC-ID:** Das EPC-Daten-Tag wird abgefragt. Zum Beispiel ist EPC-1 die EPC-Länge + EPC-Nummer + **RSSI-Wert** des ersten Tags usw. Die RSSI- und EPC-Längewerte sind Einzelbyte-Parameter. Die EPC C1 G2 Dateneinheit ist das Wort. Beim Kommunizieren werden zuerst das bedeutendste Wort und das bedeutendste Byte im Wort übertragen.

### Blocktypisierungsbefehl

Die Funktion dieses Befehls besteht darin, mehrere Wörter im reservierten Speicher, EPC, TID und Benutzerspeicher zu speichern.

### Befehlsrahmen:

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0xXX	0xXX	0x10	--	LSB	MSB

### Datenparameter[]:

Daten[]
---------

WNum	ENum	Energieleistungszertifikat	Mem e	WordPtr (Word)	Wdt
0xXX	0xXX	Variable	0xXX	0xXX	Variab le
Pwd	MaskMem	MaskAdr	MaskLen	Maskdaten	
4 Bytes	0xXX	2 Bytes	0xXX	Variable	

### Parameterdefinition

**WNum:** die Anzahl der Wörter, die man schreiben muss. Jedes Wort enthält 2 Bytes. Der WNum-Wert muss größer als 0 sein und auch identisch mit der tatsächlichen Anzahl der zu schreibenden Wörter sein. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld eine 0 oder ein ungültiger WNum-Wert angegeben wird.

### ENum:

0x00~0x0f – ENum steht für die Länge der EPC-Nummer. Die ENum-Einheit ist das Wort, und die EPC-Länge sollte weniger als 15 Wörter betragen. In dieser Situation sollten die Parameter MaskMem, MaskAdr, MaskLen und MaskData leer gelassen werden.

0xff – Anstelle von EPC-Parametern sollten MaskMem, MaskAdr, MaskLen und MaskData angegeben werden. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben ist.

**EPC:** Die EPC-Nummer des Ziel-Tags. Die Dauer des Energieeffizienzertifikats hängt von der angegebenen EPC-Nummer ab. Die Einheit EPC-Nummer ist das Wort. Beim Kommunizieren werden zuerst das bedeutendste Wort und das bedeutendste Byte

im Wort übertragen. Es ist unerlässlich, in diesem Kommando das VOLLSTÄNDIGE EPC bereitzustellen.

**Mem:** 1 Byte, Zielspeichertyp.

0x00 – reserviertes Gedächtnis  
0x01 – EPC-Speicher;  
0x02 – TID-Speicher; 0x03 –  
Benutzergedächtnis.

Alle anderen Werte sind reserviert. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben ist.

**WordPtr:** 1 Byte, die Anfangsadresse des Zielworts.

**Wdt:** Wörter, die im Tag-Speicher gespeichert werden. Das Wortvolumen in Wdt muss identisch mit dem WNum-Wert sein. Das bedeutendste Byte ist das erste in einem Wort. Wörter werden geschrieben, um von einer niedrigeren zu einer höheren Adresse zu taggen, entsprechend der Reihenfolge in Date[ ]. Zum Beispiel ist WordPtr 0x02, das erste Wort in Date[ ] (von links nach rechts) wird an die Adresse 0x02 im vom Mem angegebenen Speicher geschrieben, das zweite Wort wird dann an 0x03 geschrieben, und so weiter.

**Pwd:** 4 Bytes, dies ist ein 4-Byte-Tag-Zugriffspasswort. Von links nach rechts sind die ersten zwei Bytes die oberste Anzahl von Bytes des Passcodes, und die letzten zwei Bytes sind die unteren Bytes. Das bedeutendste Bit eines 32-Bit-Tag-Zugriffspassworts ist das bedeutendste Bit in Pwd 1 Byte (von links nach rechts). Ebenso ist das am wenigsten signifikante Bit eines 32-Bit-Tag-Zugriffspassworts das am wenigsten signifikante Bit im vierten (letzten) Pwd-Byte.

Wenn der Zielspeicher nicht passwortgeschützt ist, kann Pwd auf komplett null gesetzt werden.

**MaskMem:** 1 Byte, Anzeige des Maskenbereichs.

0x01 – EPC-Speicher;

0x02 – TID-Speicher;

0x03 – Benutzergedächtnis.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**MaskAdr:** 2 Bytes, Masken-Eingangsbitadresse, der gültige Bereich von MaskAdr ist 0~16383.

**MaskLen:** 1 Byte, Maskenbitlänge (Einheit: Bits).

**MaskData:** Maskendaten. Die Länge von MaskData entspricht MaskLen/8. Ist MaskLen kein Vielfaches einer ganzen Zahl von 8, so ist die Länge von MaskData gleich  $\text{int}[\text{MaskLen}/8]+1$ . Unbestimmte untere signifikante Ziffern sollten mit der Ziffer 0 gefüllt werden.

#### **HINWEIS:**

Wenn MaskMem, MaskAdr, MaskLen und MaskData nicht bereitgestellt werden, wird die vollständige EPC-Nummer als Maskenmuster im Inventar verwendet.

#### **Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0x05	0xXX	0x10	0x00	--	LSB	MSB

Ermittlung der Betriebsparameter von Monza4QT

Dieser Befehl wird verwendet, um die aktuellen Betriebsparameter des Tags zu erhalten, gültig nur für **Impinjs Monza 4QT-Tags**.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0xXX	0xXX	0x11	--	LSB	MSB

**Datenparameter[]:**

ENum	Energieleistungszertifikat	Pwd	MaskMem	MaskAdr	MaskLen	MaskDaten
0xXX	Variable	4 Bytes	0xXX	2 Bytes	0xXX	Variable

**ENum:**

0x00~0x0f – ENum steht für die Länge der EPC-Nummer. Die ENum-Einheit ist das Wort, und die EPC-Länge sollte weniger als 15 Wörter betragen. In dieser Situation sollten die Parameter MaskMem, MaskAdr, MaskLen und MaskData leer gelassen werden.

0xff – Anstelle von EPC-Parametern sollten MaskMem, MaskAdr, MaskLen und MaskData angegeben werden. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben ist.

**EPC:** Die EPC-Nummer des Ziel-Tags. Die Dauer des Energieeffizienzertifikats hängt von der angegebenen EPC-Nummer ab. Die Einheit EPC-Nummer ist das Wort. Beim Kommunizieren werden zuerst das bedeutendste Wort und das bedeutendste Byte im Wort übertragen. Es ist unerlässlich, in diesem Kommando das VOLLSTÄNDIGE EPC bereitzustellen.

**Pwd:** 4 Bytes, dies ist ein 4-Byte-Tag-Zugriffspasswort. Von links nach rechts sind die ersten zwei Bytes die oberste Anzahl von Bytes des Passcodes, und die letzten zwei Bytes sind die unteren Bytes. Das

bedeutendste Bit eines 32-Bit-Tag-Zugriffspassworts ist das bedeutendste Bit in Pwd 1 Byte (von links nach rechts). Ebenso ist das am wenigsten signifikante Bit des 32-Bit-Tag-Passcodes das am wenigsten signifikante Bit im vierten (letzten) Pwd-Byte.

**MaskMem:** 1 Byte, Anzeige des Maskenbereichs.

0x01 – EPC-Speicher;

0x02 – TID-Speicher;

0x03 – Benutzergedächtnis.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**MaskAdr:** 2 Bytes, Masken-Eingangsbitadresse, der gültige Bereich von MaskAdr ist 0~16383.

**MaskLen:** 1 Byte, Maskenbitlänge (Einheit: Bits).

**MaskData:** Maskendaten. Die Länge von MaskData entspricht MaskLen/8. Ist MaskLen kein Vielfaches einer ganzen Zahl von 8, so ist die Länge von MaskData gleich  $\text{int}[\text{MaskLen}/8]+1$ . Unbestimmte untere signifikante Ziffern sollten mit der Ziffer 0 gefüllt werden.

**HINWEIS:**

Wenn MaskMem, MaskAdr, MaskLen und MaskData nicht bereitgestellt werden, wird die vollständige EPC-Nummer als Maskenmuster im Inventar verwendet.

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Data[]		CRC-16-Relais	
0x07	0xXX	0x11	0x00	NC	QT- Intervallkontrolle.	LSB	MSB

				0x00	1 Byte		
--	--	--	--	------	--------	--	--

Parameterdefinition

**QTcontrol:** Funktionsparameter markieren.

bit0: Die aktuelle

Einstellung der

Spiegelseite. 0 – privat; 1

– öffentlich. Bit1:

Entfernungsschutz

einstellen. 0 – aus; 1 –

aktiviert.

Alle anderen Werte sind reserviert.

Modifiziere die Betriebsparameter von Monza4QT

Die Funktion dieses Befehls besteht darin, die aktuellen

Betriebsparameter des Tags zu ändern; dies gilt nur für **Monza 4QT-**

**Tags von Impinj.**

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0xXX	0xXX	0x12	--	LSB	MSB

**Datenparameter[]:**

ENum	Energieleistungszertifikat	QT1-Lückenkontrolle	QT0-Steuerung	Pwd	MaskMen
0xXX	Variable	0x00	0xXX	4 Bytes	0xXX

**ENum:**

0x00~0x0f – ENum steht für die Länge der EPC-Nummer. Die ENum-

Einheit ist das Wort, und die EPC-Länge sollte weniger als

15 Wörter betragen. In dieser Situation sollten die Parameter MaskMen, MaskAdr, MaskLen und MaskData leer gelassen werden.

0xff – Anstelle von EPC-Parametern sollten MaskMem, MaskAdr, MaskLen und MaskData angegeben werden. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben ist.

**EPC:** Die EPC-Nummer des Ziel-Tags. Die Dauer des Energieeffizienzcertifikats hängt von der angegebenen EPC-Nummer ab.

Die Einheit EPC-Nummer ist das Wort. Beim Kommunizieren werden zuerst das bedeutendste Wort und das bedeutendste Byte im Wort übertragen. Es ist unerlässlich, in diesem Kommando das VOLLSTÄNDIGE EPC bereitzustellen.

**QTcontrol:** 0x00 (Constnat), reserviert.

**QTcontrol0:** Tag-Betriebsparameter.

bit0: Die aktuelle

Einstellung der

Spiegelseite. 0 – privat; 1

– öffentlich. Bit1:

Entfernungsschutz

einstellen. 0 – aus; 1 –

aktiviert.

Alle anderen Werte sind reserviert.

**Pwd:** 4 Bytes, dies ist ein 4-Byte-Tag-Zugriffspasswort. Von links nach rechts sind die ersten zwei Bytes die oberste Anzahl von Bytes des Passcodes, und die letzten zwei Bytes sind die unteren Bytes. Das bedeutendste Bit eines 32-Bit-Tag-Zugriffspassworts ist das

bedeutendste Bit in Pwd 1 Byte (von links nach rechts). Ebenso ist das am wenigsten signifikante Bit des 32-Bit-Tag-Passcodes das am wenigsten signifikante Bit im vierten (letzten) Pwd-Byte.

**MaskMem:** 1 Byte, Anzeige des Maskenbereichs.

0x01 – EPC-Speicher;

0x02 – TID-Speicher;

0x03 – Benutzergedächtnis.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**MaskAdr:** 2 Bytes, Masken-Eingangsbitadresse, der gültige Bereich von MaskAdr ist 0~16383.

**MaskLen:** 1 Byte, Maskenbitlänge (Einheit: Bits).

**MaskData:** Maskendaten. Die Länge von MaskData entspricht MaskLen/8. Ist MaskLen kein Vielfaches einer ganzen Zahl von 8, so ist die Länge von MaskData gleich  $\text{int}[\text{MaskLen}/8]+1$ . Unbestimmte untere signifikante Ziffern sollten mit der Ziffer 0 gefüllt werden.

**HINWEIS:**

Wenn MaskMem, MaskAdr, MaskLen und MaskData nicht bereitgestellt werden, wird die vollständige EPC-Nummer als Maskenmuster im Inventar verwendet.

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0x05	0xXX	0x12	0x00	--	LSB	MSB

## Erweiterte Datenlesung mit zugewiesener Maske

Die Funktion dieses Befehls besteht darin, Daten aus dem reservierten Speicher, EPC-Speicher, TID-Speicher und Tag-Benutzerspeicher auszulesen. Die Leseoperation beginnt mit einer definierten Adresse, und die Dateneinheit ist ein Wort.

### Befehlsrahmen:

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0xXX	0xXX	0x15	--	LSB	MSB

### Datenparameter[]:

Daten[]						
ENum	Energieleistungszertifikat	Meme	WordPtr (Word)	Num	Pwd	MaskMem
0xXX	Variable	0xXX	2 Bytes	0xXX	4 Bytes	0xXX

### ENum:

0x00~0x0f – ENum steht für die Länge der EPC-Nummer. Die ENum-Einheit ist das Wort, und die EPC-Länge sollte weniger als 15 Wörter betragen. In dieser Situation sollten die Parameter MaskMen, MaskAdr, MaskLen und MaskData leer gelassen werden.

0xff – Anstelle von EPC-Parametern sollten MaskMem, MaskAdr, MaskLen und MaskData angegeben werden. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben ist.

**EPC:** Die EPC-Nummer des Ziel-Tags. Die Dauer des Energieeffizienz-zertifikats hängt von der angegebenen EPC-Nummer ab. Die Einheit EPC-Nummer ist das Wort. Beim Kommunizieren werden zuerst das bedeutendste Wort und das bedeutendste Byte

im Wort übertragen. Es ist unerlässlich, in diesem Kommando das VOLLSTÄNDIGE EPC bereitzustellen.

**Meme:** 1 Byte, Zielspeichertyp.

0x00 – reserviertes Gedächtnis  
0x01 – EPC-Speicher;  
0x02 – TID-Speicher; 0x03 –  
Benutzergedächtnis.

Alle anderen Werte sind reserviert. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben ist.

**WordPtr:** 2 Bytes, die Anfangsadresse des Zielworts. Zum Beispiel stellen 0x0000 dar, dass eine Leseoperation mit dem ersten Wort beginnt (dem ersten 16-Bit-Speicherbereich), 0x0001 eine Operation darstellt, die mit dem zweiten Wort beginnt, und so weiter.

**Anzahl:** 1 Byte, die Anzahl der in dieser Operation gelesenen Wörter, daher ist 0x00 kein gültiger Wert für diesen Parameter. Außerdem darf der Num-Wert 120 nicht überschreiten, d. h. die maximale Anzahl gelesener Wörter beträgt 120. Der Leser gibt den Fehlerzustand des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein Wert von 0 oder größer als 120 angegeben wird.

**Pwd:** 4 Bytes, dies ist ein 4-Byte-Tag-Zugriffspasswort. Von links nach rechts sind die ersten zwei Bytes die hohen Bytes des Zugriffspassworts, und die letzten zwei Bytes sind die unteren Bytes. Das bedeutendste Bit eines 32-Bit-Tag-Zugriffspassworts ist das bedeutendste Bit in Pwd 1 Byte (von links nach rechts). Ebenso ist das am wenigsten signifikante Bit eines 32-Bit-Tag-Zugriffspassworts das am wenigsten signifikante Bit im 4. (letzten) Pwd-Byte. Der

Passcode wird nur zur Verifizierung während der Leseoperationen im reservierten Speicher verwendet, wenn der zugehörige Zielspeicher passwortgeschützt ist und das Tag-Zugriffspasswort nicht null ist. Andernfalls sollte das PSW auf null gestellt oder auf das korrekte Zugriffspasswort gestellt werden.

**MaskMem:** 1 Byte, Anzeige des Maskenbereichs.

- 0x01 – EPC-Speicher;
- 0x02 – TID-Speicher;
- 0x03 – Benutzergedächtnis.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**MaskAdr:** 2 Bytes, Masken-Eingangsbithadresse, der gültige Bereich von MaskAdr ist 0~16383.

**MaskLen:** 1 Byte, Maskenbitlänge (Einheit: Bits).

**MaskData:** Maskendaten. Die Länge von MaskData entspricht MaskLen/8. Ist MaskLen kein Vielfaches einer ganzen Zahl von 8, so ist die Länge von MaskData gleich  $\text{int}[\text{MaskLen}/8]+1$ . Unbestimmte untere signifikante Ziffern sollten mit der Ziffer 0 gefüllt werden.

**HINWEIS:**

Wenn MaskMem, MaskAdr, MaskLen und MaskData nicht bereitgestellt werden, wird die vollständige EPC-Nummer als Maskenmuster im Inventar verwendet.

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0xXX	0xXX	0x15	0x00	Word1,Word2,...	LSB	MSB

## Parameterdefinition

**Word1, Word2...:** Die Einheit dieses Körpers ist das Wort. Jedes Wort enthält 2 Bytes, das bedeutendste Byte zuerst. Wort 1 ist das erste Wort aus der Anfangsadresse, das zweite Wort ist Wort 2, und so weiter.

## Erweiterte Datenlesung mit zugewiesener Maske

Die Funktion dieses Befehls besteht darin, mehrere Wörter in einem einzigen Kommunikationszyklus in den reservierten Speicher, TID-Speicher oder Benutzerspeicher zu schreiben.

## Befehlsrahmen:

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0xXX	0xXX	0x16	--	LSB	MSB

## Datenparameter[]:

Daten[]					
WNum	ENum	Energieleistungszertifikat	Mem	WordPtr (Word)	Wdt
0xXX	0xXX	Variable	0xXX	2 Bytes	Variab le
Pwd	MaskMem	MaskAdr	MaskLen	Maskendaten	
4 Bytes	0xXX	2 Bytes	0xXX	Variable	

**WNum:** die Anzahl der Wörter, die man schreiben muss. Jedes Wort enthält 2 Bytes. Der WNum-Wert muss größer als 0 und kleiner als oder gleich 32 sein sowie identisch mit der tatsächlichen Anzahl der zu schreibenden Wörter sein. Der Leser gibt den Fehlerstatus des

Parameters im Antwortrahmen zurück, wenn in diesem Feld eine 0 oder ein ungültiger WNum-Wert angegeben wird.

**ENum:**

0x00~0x0f – ENum steht für die Länge der EPC-Nummer. Die ENum-Einheit ist das Wort, und die EPC-Länge sollte weniger als 15 Wörter betragen. In dieser Situation sollten die Parameter MaskMem, MaskAdr, MaskLen und MaskData leer gelassen werden.

0xff – Anstelle von EPC-Parametern sollten MaskMem, MaskAdr, MaskLen und MaskData angegeben werden. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben ist.

**EPC:** Die EPC-Nummer des Ziel-Tags. Die Dauer des Energieeffizienzcertifikats hängt von der angegebenen EPC-Nummer ab. Die Einheit EPC-Nummer ist das Wort. Beim Kommunizieren werden zuerst das bedeutendste Wort und das bedeutendste Byte im Wort übertragen. Es ist unerlässlich, in diesem Kommando das VOLLSTÄNDIGE EPC bereitzustellen.

**MaskMem:** 1 Byte, Anzeige des Maskenbereichs.

0x01 – EPC-Speicher;

0x02 – TID-Speicher;

0x03 – Benutzergedächtnis.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**WordPtr:** 2 Bytes, das bedeutendste Byte zuerst die Anfangsadresse des Zielworts.

**Wdt:** Wörter, die im Tag-Speicher gespeichert werden. Das Wortvolumen in Wdt muss identisch mit dem WNum-Wert sein. Das bedeutendste Byte ist das erste in einem Wort. Wörter werden geschrieben, um von einer niedrigeren zu einer höheren Adresse zu taggen, entsprechend der Reihenfolge in Date[]. Zum Beispiel ist WordPtr 0x02, das erste Wort in Date[] (von links nach rechts) wird an die Adresse 0x02 im vom Mem angegebenen Speicher geschrieben, das zweite Wort wird dann an 0x03 geschrieben, und so weiter.

**Pwd:** 4 Bytes, dies ist ein 4-Byte-Tag-Zugriffspasswort. Von links nach rechts sind die ersten zwei Bytes der oberste Wert des Zugriffspassworts, und die letzten zwei Bytes der unterste Bytewert. Das bedeutendste Bit eines 32-Bit-Tag-Zugriffspassworts ist das bedeutendste Bit in Pwd 1 Byte (von links nach rechts). Ebenso ist das am wenigsten signifikante Bit eines 32-Bit-Tag-Zugriffspassworts das am wenigsten signifikante Bit im vierten (letzten) Pwd-Byte. Wenn der Zielspeicher nicht passwortgeschützt ist, kann Pwd auf komplett null gesetzt werden.

**MaskMem:** 1 Byte, Anzeige des Maskenbereichs.

0x01 – EPC-Speicher;

0x02 – TID-Speicher;

0x03 – Benutzergedächtnis.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**MaskAdr:** 2 Bytes, Masken-Eingangsbitadresse, der gültige Bereich von MaskAdr ist 0~16383.

**MaskLen:** 1 Byte, Maskenbitlänge (Einheit: Bits).

**MaskData:** Maskendaten. Die Länge von MaskData entspricht MaskLen/8. Ist MaskLen kein Vielfaches einer ganzen Zahl von 8, so ist die Länge von MaskData gleich  $\text{int}[\text{MaskLen}/8]+1$ . Unbestimmte untere signifikante Ziffern sollten mit der Ziffer 0 gefüllt werden.

**HINWEIS:**

Wenn MaskMem, MaskAdr, MaskLen und MaskData nicht bereitgestellt werden, wird die vollständige EPC-Nummer als Maskenmuster im Inventar verwendet.

**Befehlsrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0x05	0xXX	0x16	0x00	--	LSB	MSB

Speicherpuffer-Inventar

Der Inventarbefehl wird verwendet, um RFID-Tags zu suchen/zu identifizieren und speichert die abgefragten Tag-Daten im Speicherpuffer des Lesers.

Der Hauptunterschied zwischen dem Zählen mit und ohne Speicherpuffer (siehe Abschnitt 8.2.1) besteht darin, dass der Leser durch die Verwendung eines internen Speicherpuffers während des Zählens ständig nach Tags fragen kann. Am Ende jeder Zählzeit wird der Leser über die Gesamtzahl der im Speicherpuffer gespeicherten Tags informiert (die Wiederverwendung des Tags erhöht ebenfalls die Menge) sowie über die Anzahl der im aktuellen Inventar abgefragten Tags. Der Benutzer kann auf Daten aus dem Speicherpuffer zugreifen, indem er die Befehle verwendet, um Tag-

Daten abzurufen, die Tag-Menge zu erhalten oder den Speicherpuffer zu löschen.

Die Standard-Inventarzeit beträgt 2 Sekunden, sie kann mit dem benutzerdefinierten Lesebefehl "Reader inventory time modifizieren" (siehe Abschnitt 8.4.4) geändert oder eine temporäre Konfiguration mit einem optionalen Parameter im Inventar durchgeführt werden. Der korrekte Inventarzeitbereich liegt zwischen  $3 * 100 \text{ ms}$  und  $255 * 100 \text{ ms}$  (die tatsächliche Antwortzeit kann etwas länger als der Einstellungswert sein, und dieser Unterschied variiert von 0 bis 75 ms).

**HINWEIS**

- Eine unverhältnismäßig kurze Inventarzeit kann dazu führen, dass das Tag-Inventar ausfällt.
- Der Puffer wird gelöscht, wenn der Befehl "Count without Memory Buffer" vom Host bereitgestellt wird.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0xXX	0xXX	0x18	--	LSB	MSB

**Datenparameter[]:**

Daten[]							
QValue	Sitzung	MaskMem	MaskAdr	MaskLen	Maskdaten	AdrTI D sagt e:	LenTI D
0xXX	0xXX	0xXX	2 Bytes	0xXX	Variable	0xX X	0xX X

Daten []		
Zweck	Ameis e	Scan- Zeit
0xXX	0xXX	0xXX

**QValue:** 1 Byte.

bit7: reserviert, auf 0 gesetzt.

bit6:

Strategieindik  
ator. 0 –  
allgemeine  
Strategie; 1 –  
Spezialstrate  
gie.

bit5 ~ bit0: Der ursprüngliche Q-Wert des EPC-Tag-Inventars, die Q-Werteinstellung sollte der Regel folgen:  $2^Q \approx$  die Gesamtzahl der Tags im effektiven Feld. Der Bereich der Q-Werte ist 0~15; wenn in diesem Feld ein weiterer Wert angegeben wird, gibt der Leser den Fehlerstatus des Parameters im Antwortrahmen zurück.

**Session:** 1 Byte, EPC-Tag Inventarisierungs-Sessionswert.

0x00 – verwenden Sie S0 als Sitzungswert;  
0x01 – S1 als Sitzungswert verwenden;  
0x02 – verwenden Sie S2 als Sitzungswert;

---

0x03 – S3 als Sitzungswert verwenden;

0xff – eine smarte Lesekonfiguration verwenden (nur im EPC-Bestand anwendbar).

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**MaskMem:** 1 Byte, Anzeige des Maskenbereichs.

0x01 – EPC-Speicher;

0x02 – TID-Speicher;

0x03 – Benutzergedächtnis.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**MaskAdr:** 2 Bytes, Masken-Eingangsbithadresse, der gültige Bereich von MaskAdr ist 0~16383.

**MaskLen:** 1 Byte, Maskenbitlänge (Einheit: Bits).

**MaskData:** Maskendaten. Die Länge von MaskData entspricht  $\text{MaskLen}/8$ . Ist MaskLen kein Vielfaches einer ganzen Zahl von 8, so ist die Länge von MaskData gleich  $\text{int}[\text{MaskLen}/8]+1$ . Unbestimmte untere signifikante Ziffern sollten mit der Ziffer 0 gefüllt werden.

**AdrTID:** Die Eingabeadresse der Inventar-TID. Wenn der Befehl diesen Parameter und LenTID im selben Rahmen liefert, sollte der Zählmodus TID und nicht EPC sein.

**LenTID:** Die Datenlänge für TID-Inventaroperationen, der gültige Bereich von LenTID beträgt 0~15.

Alle anderen Werte sind reserviert, und der Leser gibt den Fehlerzustand des Parameters im Antwortrahmen zurück, wenn ein anderer Wert in diesem Feld angegeben wird. Wenn der Befehl

---

diesen Parameter und AdrTID bereitstellt, sollte der Zählmodus TID und nicht EPC sein.

**Ziel (optional):** 1 Byte, Zielwert des EPC-Tag-Inventars. 0x00 – A als Zielwert verwenden; 0x01 – B als Zielwert verwenden.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**Ameise (optionaler Parameter):** 1 Byte, Antennenauswahl für den aktuellen Bestand.

0x80 – Antenne 1;  
0x81 – Antenne 2;  
0x82 –  
Antenne 3;  
0x83 –  
Antenne 4.

**ScanTime (optional):** 1 Byte, die Zählzeit-Konfiguration für die aktuelle Zählung.

Der Leser stellt die maximale Betriebszeit auf  $\text{ScanTime} \cdot 100$  ms ein.

**HINWEIS:**

1. Wenn MaskMem, MaskAdr, MaskLen und MaskData nicht bereitgestellt werden, wird kein Maskenmuster im Inventar enthalten
2. Ziel, Ameise und ScanTime sind optionale Parameter, alle drei Parameter müssen in einem einzigen Befehlsrahmen bereitgestellt werden. Der Leser wendet die Standardeinstellung auf diese Parameter an; bis alle drei Parametereinstellungen korrekt vom Host

übermittelt wurden, greifen die neuen Einstellungen für die Inventarverfolgung in Kraft. **Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]		CRC-16-Relais	
				Zähler	Tag-Nummer		
0x09	0xXX	0x18	0x00	2 Bytes	2 Bytes	LSB	MSB

**BufferCount:** 2 Bytes, die Gesamtzahl der im Speicherpuffer gespeicherten Tags, Tags mit identischen EPC/TID-Daten werden als ein Tag behandelt. BufferCount ist die Summe der Anzahl der Tags aus mehreren Zählungen, bis der Speicherpuffer gelöscht ist.

**TagNum:** 2 Bytes, die Anzahl der im aktuellen Inventar abgefragten Tags, der mehrfach verwendete Tag erhöht ebenfalls die Menge.

### Mischinventar

Der Inventarbefehl wird verwendet, um RFID-Tags mit dem entsprechenden Protokoll im effektiven Feld zu suchen und spezifische Tag-Informationen zu erhalten, wenn die EPC-Nummer des Tags abgefragt wird.

Je nach Umständen kann der Nutzer die maximale Arbeitszeiteinstellung (Inventarzeit) anpassen. Der Leser gibt das Zählergebnis zu einer vordefinierten Zählzeit zurück. Wenn der Leser die Zählprozesse für alle Tags im effektiven Feld nicht innerhalb der vordefinierten Zählzeit abschließt, führt der Leser nach Überschreitung der Zählzeit keine weiteren Zählungen mehr durch. Der Leser meldet alle bereits gesuchten Tags sowie das aktuelle (unfertige) Inventar an den Host. Der Leser wartet dann auf den nächsten Befehl des Hosts.

Die Standard-Inventarzeit beträgt 2 Sekunden, sie kann mit dem benutzerdefinierten Lesebefehl "modify reader inventory time" geändert werden (siehe Abschnitt 8.4.4). Der korrekte Inventarzeitbereich liegt zwischen  $3 * 100 \text{ ms}$  und  $255 * 100 \text{ ms}$  (die tatsächliche Antwortzeit kann etwas länger als der Einstellungswert sein, und dieser Unterschied variiert von 0 bis 75 ms).

**HINWEIS:**

Eine unverhältnismäßig kurze Inventarzeit kann dazu führen, dass das Tag-Inventar ausfällt.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0xXX	0xXX	0x19	--	LSB	MSB

**Datenparameter[]:**

Daten[]						
QValue	Sitzung	MaskMem	MaskAdr	MaskLen	Maskendaten	
0xXX	0xXX	0xXX	2 Bytes	0xXX	Variable	
Daten[]						
Mem Reading	Adr lesen	ReadLen (Odczyt Len)	Pwd	Zweck	Ameise	Scan-Zeit
0xXX	2 Bytes	0xXX	4 Bytes	0xXX	0xXX	0xXX

**QValue:** 1 Byte.

bit7: Flagge für statistische Datenpakete.

- 
- 0 – Nach dem Inventur NICHT das statistische Datenpaket aus dem Inventarprozess bereitstellen;
  - 1 – Nach der Zählung wird das Bit4 Inventory Process Statistical Data Packet: Phase Data Flag bereitgestellt.
  - 0 – Ausschalten;
  - 1 – bit3 ~ bit0 aktivieren: der ursprüngliche Q-Wert des EPC-Tag-Inventars, die Q-Werteinstellung sollte der Regel folgen:  $2^Q \approx$  der Gesamtzahl der Tags im effektiven Feld. Der Bereich der Q-Werte ist 0~15; wenn in diesem Feld ein weiterer Wert angegeben wird, gibt der Leser den Fehlerstatus des Parameters im Antwortrahmen zurück.

**Session:** 1 Byte, EPC-Tag Inventarisierungs-Sessionswert.

- 0x00 – verwenden Sie S0 als Sitzungswert;
- 0x01 – S1 als Sitzungswert verwenden;
- 0x02 – verwenden Sie S2 als Sitzungswert;
- 0x03 – S3 als Sitzungswert verwenden;
- 0xff – eine smarte Lesekonfiguration verwenden (nur im EPC-Bestand anwendbar).

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**MaskMem (optional):** 1 Byte, Angabe des Maskenbereichs.

- 0x01 – EPC-Speicher;
- 0x02 – TID-Speicher;
- 0x03 – Benutzergedächtnis.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**MaskAdr (optionaler Parameter):** 2 Bytes, Masken-Eingangsbildadresse, der gültige Bereich von MaskAdr ist 0~16383.

---

**MaskLen (optional):** 1 Byte, Maskenbitlänge (Einheit: Bits).

**MaskData (optional):** Maskendaten. Die Länge von MaskData entspricht  $\text{MaskLen}/8$ . Ist MaskLen kein Vielfaches einer ganzen Zahl von 8, so ist die Länge von MaskData gleich  $\text{int}[\text{MaskLen}/8]+1$ . Unbestimmte untere signifikante Ziffern sollten mit der Ziffer 0 gefüllt werden.

**ReadAdr:** 2 Bytes, Ziel-Eingangsadresse, das wichtigste Byte zuerst. Zum Beispiel stellen 0x0000 dar, dass eine Leseoperation mit dem ersten Wort beginnt (dem ersten 16-Bit-Speicherbereich), 0x0001 eine Operation darstellt, die mit dem zweiten Wort beginnt, und so weiter.

**ReadLen:** 1 Byte, die Anzahl der in dieser Operation gelesenen Wörter, die minimale Länge beträgt 1 Wort und das Maximum 120 Wörter. 0x00 ist kein gültiger Wert für diesen Parameter; außerdem sollte der Num-Wert 120 nicht überschreiten. Der Leser gibt den Fehlerzustand des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein Wert von 0 oder größer als 120 angegeben wird.

**Pwd:** 4 Bytes, dies ist ein 4-Byte-Tag-Zugriffspasswort. Von links nach rechts sind die ersten zwei Bytes die oberste Anzahl von Bytes des Passcodes, und die letzten zwei Bytes sind die unteren Bytes. Das bedeutendste Bit eines 32-Bit-Tag-Zugriffspassworts ist das bedeutendste Bit in Pwd 1 Byte (von links nach rechts). Ebenso ist das am wenigsten signifikante Bit eines 32-Bit-Tag-Zugriffspassworts das am wenigsten signifikante Bit im 4. (letzten) Pwd-Byte. Der Passcode wird nur zur Verifizierung während der Leseoperationen im reservierten Speicher verwendet, wenn der zugehörige Zielspeicher passwortgeschützt ist und das Tag-Zugriffspasswort nicht null ist. Andernfalls sollte PSW auf null gesetzt oder auf das korrekte Zugriffspasswort gesetzt werden.

---

**Ziel (optional):** 1 Byte, Zielwert des EPC-Tag-Inventars. 0x00 – A als Zielwert verwenden; 0x01 – B als Zielwert verwenden.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**Ameise (optionaler Parameter):** 1 Byte, Antennenauswahl für den aktuellen Bestand.

0x80 – Antenne 1;  
0x81 – Antenne 2;  
0x82 –  
Antenne 3;  
0x83 –  
Antenne 4.

**ScanTime (optional):** 1 Byte, die Zählzeit-Konfiguration für die aktuelle Zählung.

Der Leser stellt die maximale Betriebszeit auf  $\text{ScanTime} \cdot 100$  ms ein.

**HINWEIS:**

1. Wenn MaskMem, MaskAdr, MaskLen und MaskData nicht bereitgestellt werden, wird kein Maskenmuster im Inventar enthalten
2. Ziel, Ameise und ScanTime sind optionale Parameter, alle drei Parameter müssen in einem einzigen Befehlsrahmen bereitgestellt werden. Der Leser wendet die Standardeinstellung auf diese Parameter an; bis alle drei Parametereinstellungen korrekt vom Host übermittelt wurden, greifen die neuen Einstellungen für die Inventarverfolgung in Kraft.

(1) Der Antwortstatus ist 0x26, die Antwort sollte wie folgt lauten:

Nur	Adr	reCm d sagte:	Zustan d	Daten[]			CRC- 16- Relais	
				Amei se	Lesegeschwind igkeit	Gesamtz ahl		
0xX X	0xX X	0x19	0xXX	0xXX	2 Bytes	4 Bytes	LS B	MS B

**Status:** Antwortstatus, detaillierte Beschreibung ist in der untenstehenden Tabelle dargestellt

Zustand	Beschreibung
0x26	Nach der Inventur stellen Sie statistische Daten zum aktuellen Inventarprozess bereit

**Ameise:** Zeigt an, welche Antenne nach dem Tag gefragt hat. Zum Beispiel, 0x04 im Binärcode 0000 0100 ist, bedeutet das, dass Antenne 3 nach diesem speziellen Tag gefragt hat. Für 0x08 ist es im Binärcode 0000 1000, was bedeutet, dass Antenna 4 nach Tag-Informationen gefragt hat.

**ReadRate:** Tag-Identifikationsindikator für aktuelles Inventar (erfolgreiche Identifikationszeit/Sekunde), mehrfach identifizierte Tag-Erkennung erhöht auch die Zeit der erfolgreichen Identifikation.

**TotalCount:** Die Gesamtzahl der im aktuellen Inventar entdeckten Tags sowie der Tag, den der Nutzer mehrfach aufruft, erhöht ebenfalls die Gesamtanzahl.

(2) Der Antwortstatus ist nicht 0x26, sollte die Antwort wie folgt lauten:

Nur	Adr	reCm d sagte:	Zustan d	Daten[]			CRC-16- Relais	
				Ameis e	Nu m	Datenpake t		

0xX X	0xX X	0x19	0xXX	0xXX	0xXX	Paket-1, Paket-2, Paket-3...	LS B	MS B
----------	----------	------	------	------	------	------------------------------------	---------	---------

**Status:** Antwortstatus, detaillierte Beschreibung ist in der untenstehenden Tabelle dargestellt

Zustand	Beschreibung
0x01	Operation abgeschlossen, meldet der Leser alle angeforderten Tag-Informationen an den Host
0x02	Der Bestand lief ab, der Betrieb wurde abgebrochen. Der Leser meldet alle bereits angeforderten Tags an den Host
0x03	Der Leser kann nicht auf alle Daten in einem Frame antworten, weitere Daten werden in folgenden Frames gesendet.
0x04	Der Leser füllte nur Teile des Inventars aus, aber der Speicherplatz ging wegen der Anzahl der Tags aus. Der Leser meldet alle bereits abgefragten Tags sowie diesen Zustandswert an den Host.
0xF8	Ein Antennenfehler wurde erkannt, die aktuelle Antenne kann getrennt sein.

**Ameise:** Zeigt an, welche Antenne nach dem Tag gefragt hat. Zum Beispiel, 0x04 im Binärcode 0000 0100 ist, bedeutet das, dass Antenne 3 nach diesem speziellen Tag gefragt hat. Für 0x08 ist es im Binärcode 0000 1000, was bedeutet, dass Antenna 4 nach Tag-Informationen gefragt hat.

**Anzahl:** Die Anzahl der Datenpakete mit Tag-Informationen, die in dieser Antwort enthalten sind.

**Datenpaket:** Tag-Daten werden durchsucht, Packet-1 sind die Daten des ersten Pakets, Packet-2 sind die Daten des zweiten Pakets usw. Das Format eines typischen Pakets ist unten dargestellt:

Paket-Format					
Paketparameter	Nur	Daten	RSSI-Indikator (RSSI)	Phase	Frequenz
0xXX	0xXX	Variable	0xXX	4 Bytes	3 Bytes

Datenpaket-Parameter:

**PacketParam:** 1 Byte, Parameter dieses Pakets. bit7: Zeiger des Typs dieses Datenpakets.

- 0 – dieses Datenpaket enthält die EPC-Nummer des Tags;
- 1 – dieses Datenpaket enthält die von Ihnen gesuchten Tag-Daten (die EPC-Nummer dieses Tags wird im letzten Datenpaket angegeben).

bit6 ~ bit0: Die Seriennummer des Datenpakets. Der Seriennummernbereich ist 0~127. Zu Beginn des Inventars jeder Mischung wird die Batch-Seriennummer von 0 neu gestartet und für jedes übertragene Datenpaket erhöht. Wenn die Seriennummer einen maximalen Wert von 127 erreicht, wird sie von 0 neu berechnet.

**Len:** 1 Byte, die Länge der Daten in diesem Paket. Wenn Bit6 =1, Phase und Frequenz aktivieren, sind diese Daten 7 Bytes,

**Daten:** EPC-Nummer oder Daten, die Sie aus einem bestimmten Tag-Bereich suchen. Die Dateneinheit des **EPC C1 G2-Tags** ist das Wort. Beim Kommunizieren werden zuerst das bedeutendste Wort und das bedeutendste Byte im Wort übertragen.

---

**RSSI:** 1 Byte, die Signalstärke des Lesers zum Zeitpunkt der Tag-Abfrage.

**Phase:** 4 Bytes, wenn **Lena-Bit 6 = 1** ist, starten 2 Bytes die Phase, 2 Bytes beenden die Phase;

**Frequenz:** 3 Bytes, wenn **Lena 6=1 bitt** , ist die Einheit kHz;

**HINWEIS:**

Das Datenpaket wird gemäß der von EPC vorgegebenen Reihenfolge gesendet. Für jedes Tag lädt der Leser zunächst das EPC-Tag im ursprünglichen Datenpaket hoch, weitere Tag-Daten werden im zweiten Datenpaket hochgeladen. Diese beiden Pakete haben eine aufeinanderfolgende Seriennummer.

Wenn der Leser die Tag-Daten nach dem Hochladen des EPC-Pakets nicht liest, liest er kein Tag-Datenpaket mehr. Der Leser fragt weiterhin nach dem nächsten Tag und beginnt den normalen Mischinventarprozess neu.

Inventar mit EPC-Nummer

Die Funktion dieses Befehls besteht darin, das Tag nach einem bestimmten EPC abzufragen. Der Leser antwortet nur mit dem entsprechenden EPC auf die Daten des Tags. Alle anderen Tags mit einem anderen EPC werden ignoriert.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0xXX	0xXX	0x1a	--	LSB	MSB

**Datenparameter[]:**

Daten[]			
Spieltyp	MatchLen	Anpassung versetzen	EPCData (Klimakonferenz)
0xXX	2 Bytes	2 Bytes	Variable

**MatchType:** Die Art der EPC-Nummernübereinstimmung.

0 – Label-Anfragen bei der zuständigen EPC;

1 – alle anderen Tags mit unterschiedlichen EPCs anfordern.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**MatchLen:** 2 Bytes, das bedeutendste Byte zuerst die Bitlänge der definierten EPC-Nummer. Der Bereich von MatchLen beträgt 1~196 Bits.

**MatchOffset:** 2 Bytes, das bedeutendste Byte zuerst. Die Eingangsadresse der definierten EPC (Einheit: Bits).

Der Bereich von MatchOffset beträgt 0~495 Bit.

**HINWEIS:**

Die Summe von **MatchLen** und **MatchOffset** darf 496 Bit nicht überschreiten. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein ungültiger Wert angegeben wird.

**EPCData:** Definierte EPC-Nummerndaten. Die Länge der EPCData entspricht  $\text{MatchLen}/8$ . Ist MatchLen kein Vielfaches einer ganzen Zahl von 8, so ist die Länge der EPCData gleich  $\text{int}[\text{MatchLen}/8]+1$ . Unbestimmte untere signifikante Ziffern sollten mit der Ziffer 0 gefüllt werden.

**Antwortrahmen:**

Nummer	Adresse	reCm d sagte :	Zustand	Daten []			CRC-16- Relais	
				Ameise	Num	EPC-ID		
0xXX X	0xXX X	0x1a	0xXX	0xXX	0xX X	EPC-1 (EPC-1 Versio n)	LS B	MS B

**Status:** Antwortstatus, detaillierte Beschreibung ist in der untenstehenden Tabelle dargestellt

Zustand	Beschreibung
0x01	Operation abgeschlossen, meldet der Leser alle angeforderten Tag-Informationen an den Host
0x02	Der Bestand lief ab, der Betrieb wurde abgebrochen. Der Leser meldet alle bereits angeforderten Tags an den Host
0x03	Der Leser kann nicht auf alle Daten in einem Frame antworten, weitere Daten werden in folgenden Frames gesendet.
0xF8	Ein Antennenfehler wurde erkannt, die aktuelle Antenne kann getrennt sein.

**Ameise:** Zeigt an, welche Antenne nach dem Tag gefragt hat. Zum Beispiel, 0x04 im Binärcode 0000 0100 ist, bedeutet das, dass Antenne 3 nach diesem speziellen Tag gefragt hat. Für 0x08 ist es im Binärcode 0000 1000, was bedeutet, dass Antenna 4 nach Tag-Informationen gefragt hat.

**Num:** die Höhe des EPC-Tags, der in dieser Antwort enthalten ist.

**EPC-ID:** EPC/TID-Daten abgefragt. Zum Beispiel ist EPC-1 die EPC/TID-Länge + EPC-Nummer/TID-Daten + RSSI-Wert des ersten Tags usw.

---

Die RSSI- und EPC-Längewerte sind Einzelbyte-Parameter. Die EPC C1 G2 Dateneinheit ist das Wort. Beim Kommunizieren werden zuerst das bedeutendste Wort und das bedeutendste Byte im Wort übertragen.

## QT Wrestling

Verwenden Sie diesen Befehl, um die private EPC-Nummer des Monza4QT-Tags mit öffentlichem Spiegeln zu überprüfen. Der Befehl sucht nach RFID-Tags mit dem entsprechenden Protokoll. Um weitere Operationen an einem neuen Tag mit unbekanntem EPC durchzuführen, sollte der Benutzer zunächst die EPC-Nummer des Tags mit dem Inventarbefehl erhalten.

Je nach Umständen kann der Nutzer die maximale Arbeitszeiteinstellung (Inventarzeit) anpassen. Der Leser gibt das Zählergebnis zu einer vordefinierten Zählzeit zurück. Wenn der Leser die Zählprozesse für alle Tags im effektiven Feld nicht innerhalb der vordefinierten Zählzeit abschließt, führt der Leser nach Überschreitung der Zählzeit keine weiteren Zählungen mehr durch. Der Leser meldet alle bereits gesuchten Tags sowie das aktuelle (unfertige) Inventar an den Host. Der Leser wartet dann auf den nächsten Befehl des Hosts.

Die Standard-Inventarzeit beträgt 1 Sekunden, sie kann mit dem benutzerdefinierten Lesebefehl "Inventarzeit des Lesers ändern" modifiziert werden (siehe Abschnitt 8.4.4). Der korrekte Inventarzeitbereich liegt zwischen  $3 * 100$  ms und  $255 * 100$  ms (die tatsächliche Antwortzeit kann etwas länger als der Einstellungswert sein, und dieser Unterschied variiert von 0 bis 75 ms).

**Hinweis:** Eine unzumutbar kurze Inventarzeit kann dazu führen, dass das Tag-Inventar abstürzt.

---

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0x0XX	0xXX	0x1b	--	LSB	MSB

**Datenparameter[]:**

Daten[]				
QValue	Sitzung	Zweck	Ameise	Scan-Zeit
0xXX	0xXX	0xXX	0xXX	0xXX

**QValue:** 1 Byte.

bit7: Flagge für statistische Datenpakete.

0 – Nach dem Inventur NICHT das statistische Datenpaket aus dem Inventarprozess bereitstellen;

1 – Nach der Inventur wird das statistische Datenpaket des Inventarprozesses bit6 ~ bit0 bereitgestellt: der ursprüngliche Q-Wert des EPC-Tag-Inventars; die Einstellung des Q-Werts sollte folgender Regel folgen:  $2^Q \approx$  die Gesamtzahl der Tags im effektiven Feld. Der Bereich der Q-Werte ist 0~15; wenn in diesem Feld ein weiterer Wert angegeben wird, gibt der Leser den Fehlerstatus des Parameters im Antwortrahmen zurück.

**Session:** 1 Byte, EPC-Tag Inventarisierungs-Sessionswert.

0x00 – verwenden Sie S0 als Sitzungswert;

0x01 – S1 als Sitzungswert verwenden;

0x02 – verwenden Sie S2 als Sitzungswert;

0x03 – S3 als Sitzungswert verwenden;

0xff – eine smarte Lesekonfiguration verwenden (nur im EPC-Bestand anwendbar).

---

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**Ziel (optional):** 1 Byte, Zielwert des EPC-Tag-Inventars. 0x00 – A als Zielwert verwenden; 0x01 – B als Zielwert verwenden.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**Ameise (optionaler Parameter):** 1 Byte, Antennenauswahl für den aktuellen Bestand.

0x80 – Antenne 1;  
0x81 – Antenne 2;  
0x82 –  
Antenne 3;  
0x83 –  
Antenne 4.

Alle anderen Werte sind reserviert, und der Leser gibt den Fehlerzustand des Parameters im Antwortrahmen zurück, wenn ein anderer Wert in diesem Feld angegeben wird.

**ScanTime (optional):** 1 Byte, die Zählzeit-Konfiguration für die aktuelle Zählung.

Der Leser stellt die maximale Betriebszeit auf  $\text{ScanTime} \cdot 100$  ms ein.

**HINWEIS:**

1. Ziel, Ameise und ScanTime sind optionale Parameter, alle drei Parameter müssen in einem Befehlsrahmen bereitgestellt werden. Der Leser wendet die Standardeinstellung auf diese Parameter an; bis alle drei Parametereinstellungen korrekt vom Host übermittelt

wurden, greifen die neuen Einstellungen für die Inventarverfolgung in Kraft.

(1) Der Antwortstatus ist 0x26, die Antwort sollte wie folgt lauten:

Nur	Adr	reC md sagt e:	Zustand	Daten[]			CRC- 16- Relais	
				Amei se	Lesegeschwind igkeit	Gesamtz ahl		
0xX X	0xX X	0xlb	0xXX	0xXX	2 Bytes	4 Bytes	LS B	MS B

**Status:** Der Status der Antwort, eine detaillierte Beschreibung finden Sie in der untenstehenden Tabelle,

Zustand	Beschreibung
0x26	Nach der Inventur stellen Sie statistische Daten zum letzten Inventarprozess bereit

**Ameise:** Zeigt an, welche Antenne nach dem Tag gefragt hat. Zum Beispiel, 0x04 im Binärcode 0000 0100 ist, bedeutet das, dass Antenne 3 nach diesem speziellen Tag gefragt hat. Für 0x08 ist es im Binärcode 0000 1000, was bedeutet, dass Antena 4 nach Tag-Informationen gefragt hat.

**ReadRate:** Tag-Identifikationsindikator für aktuelles Inventar (erfolgreiche Identifikationszeit/Sekunde), mehrfach identifizierte Tag-Erkennung erhöht auch die Zeit der erfolgreichen Identifikation.

**TotalCount:** Die Gesamtzahl der im aktuellen Inventar entdeckten Tags sowie der Tag, den der Nutzer mehrfach aufruft, erhöht ebenfalls die Gesamtanzahl.

(2) Der Antwortstatus ist nicht 0x26, sollte die Antwort wie folgt lauten:

Nur	Adr		Zustand	Daten[]	
-----	-----	--	---------	---------	--

		reCmd sagte:		Ameise	Num	EPC-ID	CRC-16- Relais	
0xXX	0xXX	0x1b	0xXX	0xXX	0xXX	EPC-1, EPC-2, EPC-3...	LSB	MSB

**Status:** Der Status der Antwort, eine detaillierte Beschreibung finden Sie in der untenstehenden Tabelle,

Zustand	Beschreibung
0x01	Operation abgeschlossen, meldet der Leser alle angeforderten Tag-Informationen an den Host
0x02	Der Bestand lief ab, der Betrieb wurde abgebrochen. Der Leser meldet alle bereits angeforderten Tags an den Host
0x03	Der Leser kann nicht auf alle Daten in einem Frame antworten, weitere Daten werden in folgenden Frames gesendet.
0x04	Der Leser hat nur Teile des Inventars ausgefüllt, aber in seinem Gedächtnis ist ihm der Platz ausgegangen
	aufgrund der Anzahl der Tags. Der Leser meldet alle bereits abgefragten Tags sowie diesen Zustandswert an den Host.

**Ameise:** Zeigt an, welche Antenne nach dem Tag gefragt hat. Zum Beispiel, 0x04 im Binärcode 0000 0100 ist, bedeutet das, dass Antenne 3 nach diesem speziellen Tag gefragt hat. Für 0x08 ist es im Binärcode 0000 1000, was bedeutet, dass Antenna 4 nach Tag-Informationen gefragt hat.

**Nummer:** Die in dieser Antwort enthaltene EPC/TID-Menge.

**EPC-ID:** EPC/TID-Daten abgefragt. Zum Beispiel ist EPC-1 die EPC/TID-Länge + EPC-Nummer/TID-Daten + **RSSI-Wert** des ersten Tags usw. Die RSSI- und EPC-Längewerte sind Einzelbyte-Parameter. Die EPC C1 G2 Dateneinheit ist das Wort. Beim Kommunizieren werden zuerst das bedeutendste Wort und das bedeutendste Byte im Wort übertragen.

18000-6B Kommandos

### Einzel-Tag-Inventar

Die Funktion dieses Befehls besteht darin, nach einem einzelnen Tag abzufragen. Wenn Sie mehrere Tags im effektiven Feld haben, kann der Leser nach keinem Tag fragen.

#### Befehlsrahmen:

Nur	Adr	Cmd	CRC-16-Relais	
0x04	0xXX	0x50	LSB	MSB

#### Antwortrahmen:

Nur	Adr	reCmd sagte:	Zustand	Daten[]		CRC-16-Relais	
				Ameise	ID	LSB	MSB
0x10	0xXX	0x50	0x00	0Xxx sagte:	10 Bytes		

**Ameise:** Zeigt an, welche Antenne nach dem Tag gefragt hat. Zum Beispiel, 0x04 im Binärcode 0000 0100 ist, bedeutet das, dass Antenne 3 nach diesem speziellen Tag gefragt hat. Für 0x08 ist es im Binärcode 0000 1000, was bedeutet, dass Antenna 4 nach Tag-Informationen gefragt hat.

**ID:** 10 Bytes, das 1. Byte ist die Länge der UID, 0x08. Das 2. ~ 9. Byte ist die UID-Nummer des Tags, das unbedeutendste Byte zuerst. Das zehnte Byte ist RSSI.

Fordern Sie Etiketten entsprechend einer bestimmten Bedingung an  
Die Funktion dieses Befehls besteht darin, nach Tags gemäß einer bestimmten Bedingung zu suchen.

**Befehlsrahmen:**

Nur	Adr	Cm d	Daten[]				CRC-16- Relais	
			Zustan d	Adress e	Mask e	Word_dat a		
0x0 f	0xX X	0x51	0xXX	0xXX	0xXX	8 Bytes	LS B	MS B

**Zustand:** Lagerzustand;

- 0x00 – gleichbedeutend mit Referenz;
- 0x01 – ist nicht gleich einer Referenz;
- 0x02 – größer als die Referenz
- 0x03 – weniger als die Referenz

**Adresse:** Adresse der zu vergleichenden Daten

**Maske:** Maske, gib die Daten zum Vergleich an.

**Word\_data:** Referenzwert. Jedes Bit in der Maske entspricht einem Byte im Word\_data. Das bedeutendste Bit (Bit7) entspricht dem ersten Byte im Word\_data (von links nach rechts). Daher entspricht das am wenigsten signifikante Bit (Bit0) dem letzten Byte im Word\_data (von links nach rechts). Word\_data Referenzdaten zum Vergleich mit den Daten im Tag beginnen, beginnt mit dem im Adressfeld angegebenen Ort. Das erste Byte von Word\_data wird mit dem Datenspeicher im Adressfeld verglichen, und das letzte Byte

von Word\_data wird mit dem Datenspeicher im Adress+7-Feld verglichen. Setzen Sie das entsprechende Bytebit der Maske auf 1, der Leser führt den Vergleich für dieses spezielle Byte durch. Setze das Bit auf 0, es wird kein Vergleich für dieses spezielle Byte angestellt.

**Antwortrahmen:**

Nur	Adr	reCm d sagte:	Zustan d	Ameis e	Nu m	Daten[ ]	CRC-16- Relais	
0xX X	0xX X	0x51	0xXX	0xXX	0xXX	UID1, UID2...	LS B	MS B

**Status:** Antwortstatus, detaillierte Beschreibung ist in der untenstehenden Tabelle dargestellt

Zustand	Beschreibung
0x15	Der Vorgang ist abgeschlossen, der Leser meldet alle abgefragten Tag-Informationen an den Host.
0x16	Der Bestand lief ab, der Betrieb wurde abgebrochen. Der Leser meldet alle bereits angeforderten Tags an den Host.
0x17	Der Leser kann nicht auf alle Daten in einem Frame antworten, weitere Daten werden in folgenden Frames gesendet.
0x18	Der Leser füllte nur Teile des Inventars aus, aber der Speicherplatz ging wegen der Anzahl der Tags aus. Der Leser meldet alle bereits abgefragten Tags sowie diesen Zustandswert an den Host.
0xF8	Ein Antennenfehler wurde erkannt, die aktuelle Antenne kann getrennt sein.

**Ameise:** Zeigt an, welche Antenne nach dem Tag gefragt hat. Zum Beispiel, 0x04 im Binärcode 0000 0100 ist, bedeutet das, dass Antenne 3 nach diesem speziellen Tag gefragt hat. Für 0x08 ist es im Binärcode 0000 1000, was bedeutet, dass Antenna 4 nach Tag-Informationen gefragt hat.

**Num:** Die Menge an UID, die in dieser Antwort enthalten ist.

**Data[]:** Die UID des Tags. Die Länge jedes UID-Datenblocks beträgt 10 Bytes. Das 1. Byte ist der Wert der Länge der UID-Zahl, also 0x08. Das 2. ~ 9. Byte ist die UID-Nummer des Tags, das unbedeutendste Byte zuerst. Das zehnte Byte ist der RSSI-Wert. Für Num = 0, d. h. es werden keine UID-Daten bereitgestellt, ist Data[] leer.

#### Leserdaten

Die Funktion dieses Befehls besteht darin, mehrere Bytes von einer bestimmten Adresse zu lesen.

#### Befehlsrahmen:

Nur	Adr	Cmd	Daten[]			CRC-16-Relais	
			Adresse	ID	Num		
0x0e	0xXX	0x52	0xXX	8 Bytes	0xXX	LSB	MSB

**Adresse:** Die Eingangsadresse des Ziels. Ein gültiger Bereich von Adresswerten ist 0~223. Der Leser gibt den Fehlerstatusparameter im Antwortrahmen zurück, wenn eine ungültige Adresse angegeben wird.

**Zählung:** Die Anzahl der Wörter, die in dieser Operation gelesen werden. Der gültige Bereich der Num-Werte beträgt 1~32. Wenn die Summe der Adress- und Nummerwerte größer als 224 ist, ist der Num-Wert gleich oder größer als 32, und gibt den Fehlerzustand des Parameters im Antwortrahmen zurück.

**ID:** ID des Ziel-Tags, 8 Bytes, am wenigsten signifikantes Byte zuerst.

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0xXX	0xXX	0x52	0x00	Daten	LSB	MSB

Daten: Abfrage der Daten vom Tag, zuerst das am wenigsten signifikante Byte.

Datenaufzeichnung

Die Funktion dieses Befehls besteht darin, mehrere Bytes auf ein bestimmtes Tag zu schreiben.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]			CRC-16- Relais	
			Adresse	ID	Indate		
0xXX	0xXX	0x53	0xXX	8 Bytes	Variable	LSB	MSB

**Adresse:** 1 Byte, Zielbyte-Adresse. Der Adressbereich beträgt 8~223. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben ist.

**ID:** 8 Bytes, die ID-Nummer des Ziel-Tags, zuerst das am wenigsten signifikante Byte.

**Wdate:** Die Daten, die im Tag gespeichert werden sollen. Die Länge von Wdata darf 32 Bytes nicht überschreiten. Das bedeutendste Byte wird auf das niedrigste Byte aus dem im Adressfeld angegebenen

Eintrag geschrieben. Wenn die Länge von Wdata 0 (keine Daten) beträgt oder 32 überschreitet, gibt der Leser den Fehlerstatus des Parameters im Antwortrahmen zurück. Außerdem liefert der Leser, wenn die Summe von Adresse und Wdata 224 übersteigt, ebenfalls einen Parameterfehler zurück.

**Befehlsrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0x05	0xXX	0x53	0x00	--	LSB	MSB

Erhalten Sie einen Lock-Zustand

Die Funktion dieses Befehls besteht darin, den Sperrzustand eines bestimmten Bytes zu erhalten.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]		CRC-16- Relais	
			Adresse	ID	LSB	MSB
0x0d	0xXX	0x54	0xXX	8 Bytes		

**Adresse:** 1 Byte, Zielbyte-Adresse. Der Adressbereich ist 0~223. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben ist.

**ID:** 8 Bytes, die ID-Nummer des Ziel-Tags. Das am wenigsten bedeutende Byte sollte zuerst gesendet werden.

**Befehlsrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0x06	0xXX	0x54	0x00	Lockdown- Status	LSB	MSB

LockState: 0x00: Das Byte ist entsperrt; 0x01: Das Byte ist gesperrt.

### Lock-Byte

Die Funktion dieses Befehls besteht darin, ein bestimmtes Byte im Tag zu sperren.

#### Befehlsrahmen:

Nur	Adr	Cmd	Daten[]		CRC-16-Relais	
			Adresse	ID		
0x0d	0xXX	0x55	0xXX	8 Bytes	0x0d	0xXX

**Adresse:** 1 Byte, Zielbyte-Adresse. Der Adressbereich beträgt 8~223. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben ist.

**ID:** 8 Bytes, die ID-Nummer des Ziel-Tags. Das am wenigsten bedeutende Byte sollte zuerst gesendet werden.

#### Antwortrahmen:

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16-Relais	
0x05	0xXX	0x55	0x00	--	LSB	MSB

### Individuelle Lesebefehle

#### Informationen über einen Leser erhalten

Die Funktion dieses Befehls besteht darin, Informationen über den Leser zu erhalten, einschließlich der Leseadresse, Firmware-Version, Lesermodellcode, unterstützten Protokollen, Betriebsfrequenzband,

HF-Leistung, Inventarzeit, Buzzerstatus, GPIO für Benachrichtigungen usw.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0x04	0xXX	0x21	--	LSB	MSB

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16-Relais	
0x0f	0xXX	0x21	0x00	Version, Typ, Tr_Type, dmaxfre, dminfre, Power, Scntm, Ant, Gebucht, gebucht, CheckAnt	LSB	MSB

**Datenparameter[]:**

Parameter	Länge (Bytes)	Beschreibung
Version	2	In der Firmware-Version gibt das High Byte die Hauptversionsnummer an, und das untere Byte die Minor-Versionsnummer.
Typ	1	Der Leser-Modellcode 0x0c ist RRU2881M/RRU2881LITE
Tr_Type	1	Unterstützte Prototypen. bit1 = 1: unterstützt 18000-6C. bit0 = 1: unterstützt 18000-6B.

		Alle anderen Teile sind reserviert.
dmaxfre sagte:	1	bit7~bit6: Frequenzbandkonfiguration; bit5 ~ bit0: der maximale Frequenzpunkt.
Dminfre sagte:	1	bit7~bit6: Frequenzbandkonfiguration; Bit5 ~ Bit0: Minimaler Frequenzpunkt.
Macht	1	HF-Ausgangsleistung, Bereich von 0 bis 30.
SCNTM sagte:	1	Inventarzeit. Der Leser reagiert auf den Zählbefehl des Gastgebers zu dieser bestimmten Zählzeit.
Ameise	1	Antennenkonfiguration.
Eingeschrän kt	1	Zurückhaltend.
Eingeschrän kt	1	Zurückhaltend.
Schach	1	Antennenkontrollkonfig uration 0: Antennenabprüfung; 1: Antennen-Check an.

### Frequenzband-Konfigurationstabelle

MaxFre(bit 7)	MaxFre(bit 6)	MinFre(bit 7)	MinFre(bit 6)	Frequenzba nd
0	0	0	0	Eingeschrän kt
0	0	0	1	Chinesisches Team2

0	0	1	0	Amerikanische Band
0	0	1	1	Koreanische Mannschaft
0	1	0	0	EU-Band
0	1	0	1	Eingeschränkt
0	1	1	0	Team aus der Ukraine
0	1	1	1	Team aus Peru
1	0	0	0	Chinesisches Team1
1	0	0	1	EU3-Band
1	0	1	0	Taiwanesehe Mannschaft
1	1	0	0	US Band3
...	...	...	...	...
1	1	1	1	Eingeschränkt

### Ändern Sie die Betriebsfrequenz

Die Funktion dieses Befehls besteht darin, das Frequenzband auszuwählen und die maximalen und minimalen Frequenzpunkte zu modifizieren. Der maximale Frequenzpunkt muss größer als oder gleich dem minimalen Frequenzpunkt sein.

### Befehlsrahmen:

Nur	Adr	Cmd	Daten[]	
-----	-----	-----	---------	--

			MaxFre	MinFre	CRC-16-Relais	
0x06	0xXX	0x22	0xXX	0xXX	LSB	MSB

**MaxFre:** 1 Byte.

bit7~bit6:

Frequenzbandkonfiguration;

bit5 ~ bit0: der maximale

Frequenzpunkt.

**MinFre:** 1 Byte.

bit7~bit6:

Frequenzbandkonfiguration;

Bit5 ~ Bit0: Minimaler

Frequenzpunkt.

**HINWEIS:**

Der maximale Frequenzpunkt muss größer als oder gleich dem minimalen Frequenzpunkt sein. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben ist.

**Frequenzband-Konfigurationstabelle**

MaxFre(bit 7)	MaxFre(bit 6)	MinFre(bit 7)	MinFre(bit 6)	Frequenzband
0	0	0	0	Eingeschränkt
0	0	0	1	Chinesisches Team2
0	0	1	0	Amerikanische Band

0	0	1	1	Koreanische Mannschaft
0	1	0	0	EU-Band
0	1	0	1	Eingeschränkt
0	1	1	0	Team aus der Ukraine
0	1	1	1	Team aus Peru
1	0	0	0	Chinesisches Team1
1	0	0	1	EU3-Band
1	0	1	0	Taiwanesisch e Mannschaft
1	1	0	0	US Band3
...	...	...	...	...
1	1	1	1	Eingeschränkt

### Antwortrahmen:

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0x05	0xXX	0x22	0x00	--	LSB	MSB

### Formeln für verschiedene Frequenzbänder:

Chinesisches

Team2:  $F_s = 920,125 + N * 0,25$  (MHz), wobei  $N \in [0, 19]$ .

American  $F_s = 902,75 + N * 0,5$  (MHz), wobei  $N \in [0, 49]$ .

Band:

Koreanische Mannschaft:  $F_s = 917,1 + N * 0,2$  (MHz), wobei  $N \in [0, 31]$ .

EU-Band:  $F_s = 865,1 + N * 0,2$  (MHz), wobei  $N \in [0, 14]$ .

Ukraine-Team:  $F_s = 868,0 + N * 0,1$  (MHz), wobei  $N \in [0, 6]$ .

Das Team aus Peru:  $F_s = 916,2 + N * 0,9$  (MHz), wobei  $N \in [0, 11]$ .

Chinesisches Team1:  $F_s = 840,125 + N * 0,25$  (MHz), wobei  $N \in [0, 19]$ .

EU3-Band:  $F_s = 865,7 + N * 0,6$  (MHz), wobei  $N \in [0, 3]$ .

US Band3:  $F_s = 902 + N * 0,5$  (MHz), wobei  $N \in [0, 52]$ .

Team aus Taiwan:  $F_s = 922,25 + N * 0,5$  (MHz), wobei  $N \in [0, 11]$ .

Ändere die Adresse des Lesers

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
			Adresse		
0x05	0xXX	0x24	0xXX	LSB	MSB

**Adr:** Neue Leseradresseinstellung. 0xFF keine gültige Adresse ist, gibt der Leser den Parameterfehlerstatus im Antwortrahmen zurück, wenn 0xFF in diesem Feld angegeben ist.

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16-Relais	
0x05	0xXX	0x24	0x00	--	LSB	MSB

**HINWEIS:**

Der Leser antwortet auf diesen Befehl mit der Adresse ORIGINAL.

Ändern Sie die Zählzeit des Lesers

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0x05	0xXX	0x25	Scan-Zeit	LSB	MSB
			0xXX		

**Scan-Zeit:** Inventarzeit. Der Leser passt die maximale Antwortzeit entsprechend dem benutzerdefinierten Wert (0\*100ms ~ 255\*100ms) an, und der Leser wendet diese neue Einstellung für zukünftige Inventarisierungen an. Die Standardeinstellung für die Scanzeit ist 0x14 (entspricht 20\*100 ms). Die korrekte Einstellung der Scanzeit beträgt 0x00~0xff (was 3\*100ms~255\*100ms) entspricht.

**HINWEIS:** Die tatsächliche Antwortzeit kann etwas länger als der Einstellungswert sein, und diese Differenz variiert von 0 bis 75 ms. Für Scanzeit = 0 ist die Betriebszeit erst begrenzt, wenn alle Tags inventarisiert wurden.

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16-Relais	
0x05	0xXX	0x25	0x00	--	LSB	MSB

Modifiziere die serielle Baudrate

Die Funktion dieses Befehls besteht darin, die Baudrate für den seriellen Anschluss zu verändern.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais
-----	-----	-----	---------	---------------

0x05	0xXX	0x28	Baud- Rate	LSB	MSB
			0xXX		

Baudrate-Parameterdefinition: neue Baudrate-Einstellung, Baudrate-Optionen sind 0/1/2/5/6 (siehe bitte die untenstehende Tabelle für die korrekte Baudrate), alle anderen Werte sind reserviert.

*Baud-Rate-Konfigurationstabelle*

BauDrate- Wert	Aktuelle Bitrate
0	9600 Basispunkte
1	Basispunkte 19200
2	38.400 Basispunkte
5	57.600 Basispunkte
6	115200 Basispunkte

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0x05	0xXX	0x28	0x00	--	LSB	MSB

**HINWEIS:**

Die Antwort wird mit der ORIGINAL-Bitrate-Einstellung bereitgestellt. Die neue Bitrate-Einstellung tritt in der NÄCHSTEN NACHRICHT in Kraft.

## RF-Leistung modifizieren

Die Funktion dieses Befehls besteht darin, die HF-Leistung des Lesers zu verändern. Der gültige Wert dieses Parameters ist 0~30. Bei Einstellung 30 beträgt die Ausgangsleistung etwa 1 W.

### Befehlsrahmen:

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0x05	0xXX	0x2F	Pwr	LSB	MSB
			sagte: 0xXX		

**Pwr:** 1 Byte, RF-Leistungsparameter.

bit0~bit6: RF-Leistungseinstellung, der gültige Wert dieses Parameters ist 0~30. Bei Einstellung 30 beträgt die Ausgangsleistung etwa 1 W.

bit7: Zustand des Konfigurationsverhaltens nach dem Ausschalten. 0 – Konfiguration wird gespeichert, nachdem der Leser ausgeschaltet wurde; 1 – Die Konfiguration bleibt nicht erhalten.

### Befehlsrahmen:

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16-Relais	
0x05	0xXX	0x2F	0x00	--	LSB	MSB

---

## Summer-/LED-Steuerung

Die Funktion dieses Befehls besteht darin, den TAG und den Summer (mit GPO1-Pin zu teilen) zu steuern, um bestimmte Aktionen auszuführen.

### Befehlsrahmen:

Nur	Adr	Cmd	Daten[]			CRC-16-Relais	
			ActiveT	Silent T	Zeiten		
0x07	0xXX	0x33	0xXX	0xXX	0xXX	LSB	MSB

**ActiveT:** TAG LED und Summer (Pin GPO1 niedrig TTL) Einschaltzeit beträgt  $\text{ActiveT} \cdot 50$  ms. Der ActiveT-Bereich beträgt 0~255. Die Standardeinstellung ist 0.

**SilentT:** TAG LED und Summer (Pin GPO1 niedriges TTL-Niveau) Abschaltzeit ist  $\text{SilentT} \cdot 50$  ms. Der Bereich von SilentT beträgt 0~255. Die Standardeinstellung ist 0.

**Zeiten:** TAG-LED und Buzzer-Arbeitszeit, die Zeitspanne beträgt 0~255. Die Standardeinstellung ist 0.

### Antwortrahmen:

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16-Relais	
0x05	0xXX	0x33	0x00	—	LSB	MSB

## Antennenmultiplex-Konfiguration

Die Funktion dieses Befehls besteht darin, die Konfiguration der Leseantenne zu verändern.

### Befehlsrahmen:

Nur	Adr	Cmd	Daten[]	CRC-16-Relais
			Ameise	

---

0x05	0xXX	0x3f	0xXX	LSB	MSB
------	------	------	------	-----	-----

**Ameisen:** Antennenkonfigurationsinformationen. 1 Antenne ist die Mindestanforderung für den Betrieb.

bit0: 1

Antennenkonfiguratio

nsbit. 0 – Antenne 1

ausschalten; 1 –

Antenne einschalten 1.

bit1:

Antennenkonfiguratio

nsbit 2.0 – Antenne 2

ausschalten; 1 –

Antenne einschalten

2. bit2:

Antennenkonfiguratio

nsbit 3.0 – Antenne

ausschalten 3; 1 –

Antenne einschalten

3. bit3:

Antennenkonfiguratio

nsbit 4.0 – Antenne ausschalten

4; 1 – Antenne

einschalten 4.

bit4 ~ bit6: Reserviert, der Standardwert ist 0.

Bit7: Konfigurationserhaltungszustand.

0 – Konfiguration, die gespeichert wird, wenn der Leser ausgeschaltet wird; 1 – Die Konfiguration wurde nicht gespeichert.

---

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0x05	0xXX	0x3f	0x00	--	LSB	MSB

Schalte den Summer an/aus

Die Funktion dieses Befehls besteht darin, den Summer ein- und auszuschalten (der mit dem GPIO-Pin geteilt wird).

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16- Relais	
			Piep		
0x05	0xXX	0x40	0Xxx sagte:	LSB	MSB

**BeepEn:** Auswahl des Buzzer-

Modus. bit0: 0 – Buzzer  
ausschalten;

1 – Schalten Sie den Summer ein, der Buzzer piept bei jeder Operation des Erfolgstats (niedriger Ausgangspuls des GPIO-Pins).  
bit1 ~ bit7 sind reserviert, der Standardwert ist 0.

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0x05	0xXX	0x40	0x00	--	LSB	MSB

GPIO-Regelung

Die Funktion dieses Befehls besteht darin, den GPIO-Ausgang zu steuern. Der Standardausgang ist ein hoher TTL-Pegel.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	
-----	-----	-----	---------	--

			Ausgangspin	CRC-16-Relais	
0x05	0xXX	0x46	0xXX	LSB	MSB

Parameterdefinition

**OutputPin:** Der Status des GPIO-Ausgangs (Pin Out1 zu Out2). Bit0 ~ bit3 steuern jeweils Out1 bis Out3. Bit2 ~ bit7 sind reserviert.

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16-Relais	
0x05	0xXX	0x46	0x00	--	LSB	MSB

Erhalten Sie GPIO-Status

Die Funktion dieses Befehls besteht darin, die Ein- und Ausgabezustände des GPIO-Befehlsrahmens zu erhalten:

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0x04	0xXX	0x47	--	LSB	MSB

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16-Relais	
				Ausgangspin		
0x06	0xXX	0x47	0x00	0xXX	LSB	MSB

Parameterdefinition

**OutputPin:** GPIO-Ein- und Ausgabezustände. Bit0 repräsentiert den Zustand von Pin IN1, Bits 4 und Bit5 den Zustand von Out1 zu Out2. Alle anderen Werte sind reserviert.

Holen Sie sich eine eindeutige Seriennummer für Ihren Leser

Die Funktion dieses Befehls besteht darin, eine eindeutige Seriennummer für den Leser zu erhalten.

---

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0x04	0xXX	0x4c	--	LSB	MSB

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16-Relais	
				SeriaNo	LSB	MSB
0x09	0xXX	0x4c	0x00	4 Bytes	LSB	MSB

**Seriennummer:** 4 Bytes, eindeutige Seriennummer des Lesers.

Modifiziere die benutzerdefinierte Funktion eines Tags

Die Funktion dieses Befehls besteht darin, die benutzerdefinierten Werkzeuge bestimmter Tags auszuführen, um eine bestimmte Tag-Funktion zu erreichen.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
			Patrontyp	LSB	MSB
0x05	0xXX	0x3a	0xXX	LSB	MSB

Parameterdefinition

**InlayType:** Tag-Typ, der gültige Wert dieses Parameters ist 0~254.

0 – Der Standardwert dieses Parameters, also die Art des Tags, den er darstellt, ist nicht angegeben.

1 – die Peek-Funktion des Monza4QT-Tags ausführen (der Tag-Status wechselt vorübergehend von öffentlich zu privat). Die Ausführung dieser Funktion beeinflusst die Lese-/Schreibdaten das Lesen/Schreiben von Blöcken, das Schreiben von Schutzwörtern und den Schreibvorgang der EPC-Nummer.

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0x06	0xXX	0x3a	0x00	Patronentyp	LSB	MSB

**InlayType:** Tag-Typ, der gültige Wert dieses Parameters ist 0~254.

### Antennenprüfung aktivieren

Die Funktion dieses Befehls besteht darin, die Antennenprüfung während der Tag-Lese/Schreibvorgänge zu aktivieren oder zu deaktivieren.

### Befehlsrahmen:

Nur	Adr	Cmd	Daten[]	CRC-16- Relais	
			Schach	LSB	MSB
0x05	0xXX	0x66	0xXX	LSB	MSB

**CheckAnt:** 1 Byte,

Antennenprüfschalter.

0x00 –

Antennenprüfung

deaktivieren; 0x01 –

Antennenprüfung

aktivieren.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

### Antwortrahmen:

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0x05	0xXX	0x66	0x00	--	LSB	MSB

---

## Änderung der Kommunikationsschnittstelle

Die Funktion dieses Befehls besteht darin, die Kommunikationsschnittstelle zwischen dem RRU2881 und dem Host zu konfigurieren. Die Einstellung dieses Parameters bleibt erhalten, auch wenn der Leser keinen Zugang zu Strom hat. Es kann nur eine Art von Schnittstelle ausgewählt werden; der Benutzer sollte den Leser entsprechend dieser Konfiguration anschließen.

### **Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
			Kommunikationsart		
0x05	0xXX	0x6A	0xXX	LSB	MSB

CommType: 1 Byte,

Kommunikationsschnittstellentyp.

0x00 – die USB-Verbindung

einschalten; 0x01 – Aktivieren Sie die UART-Verbindung.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

### **HINWEIS:**

1. Wenn die USB-Verbindung aktiviert ist, ist die UART-Verbindung nicht verfügbar. Ebenso ist die USB-Verbindung nicht verfügbar, wenn die UART-Verbindung aktiviert ist.
2. Nach der Änderung der Kommunikationsschnittstelle tritt die Einstellung erst in Kraft, wenn der Leser ordnungsgemäß deaktiviert und neu gestartet wurde.

### **Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0x05	0xXX	0x6A	0x00	--	LSB	MSB

Konfiguration der Antennen-Reflexionsdämpfungsschwellenwerte modifizieren oder belasten

Die Funktion dieses Befehls besteht darin, die Reflexionsdämpfungsschwelle zu modifizieren oder zu belasten, um die Antenne zu überprüfen. Die Standard-Spiegelungsdämpfungsschwelle liegt bei -6 dB. Die Einstellung dieses Parameters bleibt erhalten, auch wenn der Leser keinen Zugang zu Strom hat.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16- Relais	
			ReturnLoss		
0x05	0xXX	0x6e	0xXX	LSB	MSB

**ReturnLoss:** Aktion auf die Return-Loss-Konfiguration. bit7: Lade/Ändern des Return Loss-Flags.

0 – die aktuelle Konfiguration des Reflexionsverlusts belasten; 1 – Modifiziere die Konfiguration der Reflexionsdämpfung. bit6 ~ bit0: Rückgabe-Verlust-Konfiguration Für bit7 = 0 wird bit6 ~ bit0 ignoriert.

Für bit7 = 1 ist bit6 ~ bit0 die gewünschte Reflexionsdämpfungskonfiguration. Der Reflexionsdämpfungsbereich beträgt 0~20, laut 0dB~20dB.

Andere Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn ein anderer Wert in diesem Feld angegeben wird.

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16-Relais	
				ReturnLoss		
0x06	0xXX	0x6e	0x00	0xXX	LSB	MSB

**ReturnLoss:** die aktuelle Konfiguration der Reflexionsdämpfung. bit7: reserviert, die Standardeinstellung ist 0. bit6 ~ bit0: der aktuelle Schwellenwert der Antennen-Reflexionsdämpfung. Der Bereich ist 0~20.

Änderung der maximalen EPC/TID-Längenkonfiguration für den Speicherpuffer

Die Funktion dieses Befehls besteht darin, die maximale EPC/TID-Länge für ein Speicherpufferinventar zu ändern. Diese Änderung löscht alle Tag-Daten, die zuvor im Speicherpuffer gespeichert waren.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
			Nur speichern		
0x05	0xXX	0x70	0xXX	LSB	MSB

**SaveLen:** 1 Byte, maximale EPC/TID-Länge.

0 – Länge von 128 Bits, also 16 Bytes. Die maximale Anzahl an Speichertags, die für diese Einstellung gespeichert werden kann, beträgt 528; 1 – Länge 496 Bit, also 62 Bytes. Die

maximale Anzahl an Tags, die für diese Einstellung gespeichert werden können, beträgt 160.

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0x05	0xXX	0x70	0x00	--	LSB	MSB

EPC/TID-Konfiguration mit maximaler Länge

Die Funktion dieses Befehls besteht darin, die maximale EPC/TID-Längenkonfiguration für den Speicherpuffer des Lesers zu laden

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16- Relais	
0x04	0xXX	0x71	--	LSB	MSB

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0x06	0xXX	0x71	0x00	Nur speichern	LSB	MSB

**SaveLen:** 1 Byte, maximale

EPC/TID-Länge. 0 – 128 Bit

lang, also 16 Bytes; 1 – 496

Bit, oder 62 Bytes.

Daten aus einem Speicherpuffer abholen

Die Funktion dieses Befehls besteht darin, alle Tag-Informationen aus dem Speicher des Lesers zu erhalten. Dieser Befehl beeinflusst nicht den Zustand der gespeicherten Daten. Du kannst mehrfach auf die

Daten zugreifen. Andererseits entfernt der Tag-Counting-Befehl (siehe Abschnitt 8.2.1) alle Daten aus dem Speicherpuffer.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0x04	0xXX	0x72	--	LSB	MSB

**Antwortrahmen:**

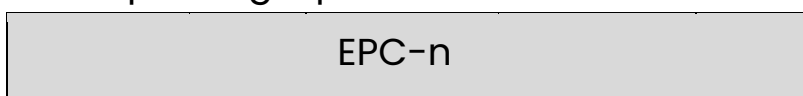
Nur	Adr	reCmd sagte:	Zustand	Daten[]		CRC-16-Relais	
				Num	EPC-Daten		
0xXX	0xXX	0x72	0xXX	0xXX	EPC-1, EPC-2, ... EPC-n	LSB	MSB

**Status:** Antwortstatus, detaillierte Beschreibung ist in der untenstehenden Tabelle dargestellt

Zustand	Beschreibung
0x01	Nach Abschluss des Vorgangs meldet der Leser alle abgefragten Tag-Informationen an den Host
0x03	Der Leser kann nicht auf alle Daten in einem Frame antworten, weitere Daten werden in folgenden Frames gesendet.

**Nummer:** 1 Byte, die EPC/TID-Größe, die in dieser Antwort enthalten ist.

**EPC-Daten:** Beschriftung von EPC/TID-Daten, die in einem Speicherpuffer gespeichert sind.



Ameise	Nur	EPC/TID-Zertifikat	RSSI-Indikator (RSSI)	Graf
0xXX	0xXX	N Bytes	0xXX	0xXX

**Ameise:** Gibt an, welche Antenne die Tag-Anfrage gesendet hat. Zum Beispiel ist 0x05 im Binärcode 0101, was für Antenne 3 und Antenne 1 steht.

**Len:** 1 Byte, EPC/TID-Bytelänge.

**EPC/TID:** EPC-Nummer/TID-Tag-Daten, die Datenlänge sollte identisch mit der von Len sein. Jede EPC-Tagnummer oder TID-Daten ist das bedeutendste Wort und das bedeutendste Byte im Wort zuerst.

**RSSI:** 1 Byte, die Signalstärke des Lesers zum Zeitpunkt der Tag-Abfrage.

**Anzahl:** Die Gesamtzahl der Erkennungsmaße für ein bestimmtes Tag. 0xFF – Die Gesamterkennungszeit beträgt mehr als oder gleich 255.

### Speicherpuffer löschen

Die Funktion dieses Befehls besteht darin, alle gespeicherten Tag-Daten aus dem Speicherpuffer zu löschen.

#### Befehlsrahmen:

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0x04	0xXX	0x73	--	LSB	MSB

#### Befehlsrahmen:

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0x05	0xXX	0x73	0x00	--	LSB	MSB

Holen Sie die Gesamtmenge der Tags aus dem Speicherpuffer

Die Funktion dieses Befehls besteht darin, die Gesamtmenge der im Speicherpuffer gespeicherten Tags zu ermitteln. **Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16- Relais	
0x04	0xXX	0x74	--	LSB	MSB

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
				Graf		
0x07	0xXX	0x74	0x00	2 Bytes	LSB	MSB

**Anzahl:** 2 Bytes, die Gesamtmenge des im Speicherpuffer gespeicherten Tags, zuerst das bedeutendste Byte.

Echtzeit-Inventarmodus-Parameteränderung

Die Funktion dieses Befehls besteht darin, die mit dem Inventar verbundenen Parameter in Echtzeit zu verändern.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16- Relais	
0xXX	0xXX	0x75	--	LSB	MSB

**Datenparameter[]:**

Daten[]
---------

TagProtocol	Lesezeit	FliterTime (Czas Określenia)	QValue	Sitzun g	
0xXX	0xXX	0xXX	0xXX	0xXX	
daten[]					
MaskMem	MaskAd r	MaskLe n	Maskendate n	AdrTID sagte:	LenTI D
0xXX	2Bytes	0xXX	Variable	0xXX	0xXX

**TagProtocol:** 1 Byte, Tag-Typdefinition für Echtzeitählung.

0 – EPC C1G2 (ISO18000-6C) Etiketten.

1 – Markierungen ISO18000-6B. Die folgenden Parameterkonfigurationen werden für 6B, QValue, Session, MaskMem, MaskAdr, MaskLen, MaskData, AdrTID, LenTID ignoriert.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**ReadPauseTime:** 1 Byte, Zeitintervall zwischen 2 Echtzeitählungen.

0x00 – 10 ms;

0x01 – 20 ms;

0x02 – 30 ms;

0x03 – 50 ms;

0x04 – 100 ms.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**FliterTime:** 1 Byte, die Zeit, um Tags zu filtern. Der gültige Wert dieses Parameters ist 0~255, er entspricht  $(0\sim 255) \cdot 1s$ . Im Echtzeitinventar sendet der Leser die Tag-Informationen dieses Tags nur einmal

---

innerhalb der angegebenen Filterzeit, wenn er ein bestimmtes Tag mehr als einmal erkennt. Für FilterTime = 0 deaktivieren Sie die Tag-Filterfunktion.

**QValue:** 1 Byte.

bit7: reserviert, auf 0 gesetzt.

bit6:

Strategieindi-  
kator. 0 –  
allgemeine  
Strategie; 1 –  
Spezialstrate-  
gie.

bit5 ~ bit0: Der ursprüngliche Q-Wert des EPC-Tag-Inventars, die Q-Werteinstellung sollte der Regel folgen:  $2^Q \approx$  die Gesamtzahl der Tags im effektiven Feld. Der Bereich der Q-Werte ist 0~15, Wird in diesem Feld ein anderer Wert angegeben, gibt der Leser den Fehlerstatus des Parameters im Antwortrahmen zurück.

**Session:** 1 Byte, EPC-Tag Inventarisierungs-Sessionswert.

0x00 – verwenden Sie S0 als Sitzungswert;

0x01 – S1 als Sitzungswert verwenden;

0x02 – verwenden Sie S2 als Sitzungswert;

0x03 – S3 als Sitzungswert verwenden;

0xff – eine smarte Lesekonfiguration verwenden (nur im EPC-Bestand anwendbar).

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**MaskMem:** 1 Byte, Anzeige des Maskenbereichs.

- 0x01 – EPC-Speicher;
- 0x02 – TID-Speicher;
- 0x03 – Benutzergedächtnis.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**MaskAdr:** 2 Bytes, Masken-Eingangsbithadresse, der gültige Bereich von MaskAdr ist 0~16383.

**MaskLen:** 1 Byte, Maskenbitlänge (Einheit: Bits).

**MaskData:** Maskendaten. Die Länge von MaskData entspricht MaskLen/8. Ist MaskLen kein Vielfaches einer ganzen Zahl von 8, so ist die Länge von MaskData gleich  $\text{int}[\text{MaskLen}/8]+1$ . Unbestimmte untere signifikante Ziffern sollten mit der Ziffer 0 gefüllt werden.

**HINWEIS:**

1. Wenn MaskMem, MaskAdr, MaskLen und MaskData nicht bereitgestellt werden, wird kein Maskenmuster im Inventar enthalten
2. Im Echtzeit-Inventarmodus wendet der Leser die vordefinierten Antennen- und Inventarzeitparameter auf das Inventar an.

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0x05	0xXX	0x75	0x00	--	LSB	MSB

Ändern Sie den Betriebsmodus des Lesers

Die Funktion dieses Befehls besteht darin, den Betriebsmodus des Lesers zu verändern.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	
-----	-----	-----	---------	--

			Lesemodus	CRC-16-Relais	
0x05	0xXX	0x76	0xXX	LSB	MSB

**ReadMode:** 1 Byte, aktueller Leserbetrieb.

0 – Empfangsmodus;

1 – Echtzeit-Inventarmodus;

2 – Echtzeit-Inventarmodus mit Auslöser. Mit dieser Konfiguration nimmt der Leser nur dann eine Bestandsaufnahme auf, wenn das TTL-Level des GPII-Ports niedrig ist. Wenn das TTL-Level von GPII hoch ansteigt, geht der Leser in den Standby-Modus.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

#### **Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16-Relais	
0x05	0xXX	0x76	0x00	--	LSB	MSB

#### **HINWEIS:**

Die Einstellung dieses Parameters bleibt auch dann erhalten, wenn der Leser nicht mit Strom versorgt ist, d. h. der Leser wendet bei jedem Neustart die vorherige Betriebsmoduseinstellung an.

#### **Ausgabeformat im Echtzeit-Inventarmodus:**

Wenn Sie den Leser auf Echtzeit-Inventarmodus stellen, führt der Leser automatisch eine Inventur gemäß der vordefinierten Konfiguration durch.

Wenn Sie nach einem Tag gefragt werden, der einer vordefinierten Bedingung entspricht, sendet der Leser die Daten automatisch im folgenden Format:

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0xXX	0xXX	0xee	0xXX	--	LSB	MSB

**Status:** Antwortstatus, detaillierte Beschreibung ist in der untenstehenden Tabelle dargestellt

Zustand	Beschreibung
0x00	Das entsprechende Tag wurde erkannt, der Leser meldet auch die erkannten Tag-Daten.
0x28	Innerhalb des Zeitintervalls des Impulspakets wird kein entsprechendes Tag erkannt, der Leser übergibt das Pulspaket.

(1) Der Antwortstatus ist 0x00, die Datenparameter [] sind unten dargestellt,

Daten[]			
Ameise	Nur	EPC/TID- Zertifikat	RSSI- Indikator (RSSI)
0xXX	0xXX	N Bytes	0xXX

Parameterdefinition

**Ameise:** Gibt an, welche Antenne die Tag-Anfrage gesendet hat. Zum Beispiel ist 0x05 im Binärcode 0101, was für Antenne 3 und Antenne 1 steht.

**Len:** 1 Byte, EPC/TID-Bytelänge.

**EPC/TID:** EPC-Nummer/TID-Tag-Daten, die Datenlänge sollte identisch mit der von Len sein. Die EPC C1 G2 Dateneinheit ist das

---

Wort. Jede EPC-Tagnummer oder TID-Daten ist das bedeutendste Wort und das bedeutendste Byte im Wort zuerst.

**RSSI:** 1 Byte, die Signalstärke des Lesers zum Zeitpunkt der Tag-Abfrage.

(2) Der Antwortstatus ist 0x28, die Datenparameter [] sind unten dargestellt

Daten[]		
Paket Nr.	AntStatus (AntStatus)	Gesamtzahl
4 Bytes	4 Bytes	4 Bytes

Parameterdefinition

**PacketNo:** die Seriennummer des Impulses. 4 Bytes, zuerst das bedeutendste Byte.

Die Seriennummer wird für jede Herzschlag-Paketoperation erhöht. Überschreitet die Herzschlag-Seriennummer den maximalen Wert von 4 Bytes, wird die Seriennummer von 0 neu berechnet.

**AntStatus:** Der aktuelle Zustand der 4 Antennen, von 1 bis 4 gemäß der Byte-Reihenfolge eingestuft.

- 0x00 – die Zielantenne ist leer, sie ist keine effektive Inventarantenne
- 0x01 – die Zielantenne funktioniert ordnungsgemäß;
- 0x02 – die Zielantenne ist getrennt.

**TotalCount:** Die Gesamtzahl der Tags, die seit Beginn der aktuellen automatisierten Zählung angefordert wurden. Das wiederholte Identifizieren des Tags erhöht ebenfalls den TotalCount-Wert. 4 Bytes, mit dem bedeutendsten Byte am Anfang. Wenn die

Gesamtmenge des angeforderten Tags den maximalen Wert von 4 Bytes übersteigt, berechnet TotalCount von 0 neu.

Im Echtzeit-Inventarmodus reagiert der Leser nur auf 3 Befehle:

1. Informationen über den Leser zu erhalten (siehe Abschnitt 8.4.1);
2. Ändern Sie den Betriebsmodus des Lesers (siehe Abschnitt 8.4.22);
3. Die Parameter des automatischen Modus erhalten (siehe Abschnitt 8.4.23).

#### Hol den Lesemodus

Die Aufgabe dieses Befehls ist es, die Parameter des Betriebsmodus des Lesers zu ermitteln.

#### Befehlsrahmen:

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0x04	0xXX	0x77	--	LSB	MSB

#### Antwortrahmen:

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16-Relais	
0xXX	0xXX	0x77	0x00	ReadMode, TagProtocol, ReadPauseTime, FliterTime, QValue, Sitzung, MaskMem, MaskAdr, MaskLen, MaskData, AdrTID, LenTID	LSB	MSB

---

## Parameterdefinition

**ReadMode:** 1 Byte, aktueller Leserbetrieb.

**TagProtocol:** 1 Byte, Tag-Typdefinition für Echtzeitählung.

**ReadPauseTime:** 1 Byte, Zeitintervall zwischen 2 Echtzeitählungen.

**FliterTime:** 1 Byte, die Zeit, um Tags im Echtzeit-Inventar zu filtern.

**QValue:** 1 Byte.

bit7: reserviert, auf 0 gesetzt.

bit6:

Strategieindi

kator. 0 –

allgemeine

Strategie; 1 –

Spezialstrate

gie.

bit5 ~ bit0: Der ursprüngliche Q-Wert des EPC-Tag-Inventars, die Q-Werteinstellung sollte der Regel folgen:  $2^Q \approx$  die Gesamtzahl der Tags im effektiven Feld. Der Bereich der Q-Werte ist 0~15; wenn in diesem Feld ein weiterer Wert angegeben wird, gibt der Leser den Fehlerstatus des Parameters im Antwortrahmen zurück.

**Sitzung:** 1 Byte, Echtzeit-Sitzungswert für EPC-Inventar.

**MaskMem, MaskAdr, MaskLen, MaskData:** Maskenstatus für **Echtzeit-Inventar von EPC C1G2-Tags**. MaskMem und MaskLen sind jeweils 1 Byte groß. MaskAdr ist 2 Bytes lang, das bedeutendste Byte zuerst. MaskData hat eine feste Länge von 32 Bytes, füllen Sie den Inhalt des MaskLen-Werts mit null.

**AdrTID:** Die anfängliche verbale Adresse des TID-Speicherinventars.

**LenTID:** Die Länge der Daten für TID-Zählungsoperationen.

Laden oder verändern Sie die Pause von Herzschlagpaketen im Echtzeit-Inventar

Die Funktion dieses Befehls besteht darin, den Timeout der Übertragung von Herzschlagpaketen im Echtzeit-Zählmodus zu ändern. Für HeartBeatTime = 0 deaktivieren Sie die Heartbeat-Paketlieferung.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
			Herzschlagzeit		
0x05	0xXX	0x78	0xXX	LSB	MSB

**HeartBeatTime:** Zeitintervall zwischen der Zustellung von 2 Beat-Paketen  
 7: Aktion bei Pulszeitintervall-Konfiguration 0 – Lade-Impuls-Zeitintervall-Konfiguration;

1 – Verändern Sie die Konfiguration der Pulszeitpause.

bit6 ~ bit0:

Herzschlagzeitintervall-

Konfiguration Für bit7 = 0 wird

bit6 ~ bit0 ignoriert.

Für bit7 = 1 ist bit6 ~ bit0 die Konfiguration der Zeitlücke zwischen 2 Heartbeat-Paketlieferungen Für HeartBeatTime = 0 deaktivieren Sie die Zustellung von Herzschlagpaketen.

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16-Relais	
				Herzschlagzeit		
0x06	0xXX	0x78	0x00	0xXX	LSB	MSB

---

**HeartBeatTime:** Die aktuelle Timeshare-Konfiguration des Heartbeat-Pakets. Bit7: Reserviert, der Standardwert ist 0.

bit6 ~ bit0: Die aktuelle Konfiguration des Heartbeat-Paket-Timeouts. Der gültige Wert dieses Parameters ist 0~127, er entspricht  $(0\sim127)*30s$ .

**HINWEIS:**

Der Leser liefert ein Pulspaket nur, wenn für einen weiteren Zeitraum (länger als eine Pause im Herzschlag) kein Tag erkannt wird.

Anpassung der HF-Leistungskonfiguration separat für Schreibvorgänge

Die Funktion dieses Befehls besteht darin, die HF-Leistung für Schreiboperationen zu modifizieren. Der Standardwert für den Leistungsparameter ist 0x00. Die Einstellung dieses Parameters bleibt erhalten, auch wenn der Leser keinen Zugang zu Strom hat.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
			Schreibkraft		
0x05	0xXX	0x79	0xXX	LSB	MSB

**WritePower:** Energiekonfiguration.

bit7: Separate RF-Energiekonfiguration für Schreibvorgänge aktivieren/deaktivieren.

- 0 – schalten Sie die separate RF-Leistungskonfiguration aus, die Leistung ist für beide Lese- und Schreibbefehle identisch.
- 1 – eine separate HF-Leistungskonfiguration aktivieren. Der korrekte HF-Leistungswert ist 0~30, für Einstellung 30 beträgt die Ausgangsleistung etwa 1 W.

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0x05	0xXX	0x79	0x00	--	LSB	MSB

Separate HF-Leistung gilt für die folgenden Befehle:

**EPC C1G2-Befehle (ISO18000-6C)**

- den Befehl zum Speichern von Daten;
- geben Sie den EPC-Nummerbefehl ein;
- Erweiterter Befehl zum Schreiben von Daten mit einer zugewiesenen Maske; Block-Delete-Befehl ; Befehl zum Schreiben von Blöcken.

**Kommando ISO18000-6B**

- Write Data Befehl

Last-RF-Leistungskonfiguration für den Schreibbetrieb

Die Funktion dieses Befehls besteht darin, eine vordefinierte, separate RF-Leistungskonfiguration der Schreiboperation zu laden.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16- Relais	
0x04	0xXX	0x7A	--	LSB	MSB

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais
				Schreibkraft	

0x06	0xXX	0x7A	0x00	0xXX	LSB	MSB
------	------	------	------	------	-----	-----

**WritePower:** Energiekonfiguration.

bit7: Konfigurieren Sie eine separate RF-Energiekonfiguration für Schreiboperationen.

- 0 – die separate HF-Leistungskonfiguration deaktiviert ist, die Leistung ist für Lese- und Schreibbefehle identisch.
- 1 – Eine separate HF-Leistungskonfiguration wurde aktiviert. Der korrekte HF-Leistungswert ist 0~30, für Einstellung 30 beträgt die Ausgangsleistung etwa 1 W.

Ändern oder laden Sie die maximale Schreibzeit-Konfiguration

Die Funktion dieses Befehls besteht darin, die maximale Schreib-Wiederholungszeit-Konfiguration zu modifizieren oder zu laden. Beim Schreiben von Tags und zugehörigen Befehlen versucht der Leser die Operation erneut, bis der Erfolg oder die maximale Schreibzeit erreicht ist. Der Standardwert für die Wiederholungszeit beträgt 3. Die Einstellung dieses Parameters bleibt erhalten, auch wenn der Leser keinen Zugang zu Strom hat.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
			Wiederholungsversuch	LSB	MSB
0x05	0xXX	0x7B	0xXX	LSB	MSB

**RetryTimes:** Aktion bei der Konfiguration der maximalen Schreib-Wiederholungszeit bit7:

Laden/Ändern der maximalen Schreib-Wiederholungszeit-Konfiguration

- 0 – die maximale Zeit für einen erneuten Schreibversuch zu laden;

1 – Ändern Sie die maximale Wiederholungszeit für Schreibversuche; bit6 ~ bit3: reserviert.

bit2 ~ bit0: maximale Schreibwiederholungszeit  
Für bit7 = 0 wird bit6 ~ bit0 ignoriert.

Für bit7 = 1 ist bit2 ~ bit0 die maximale Wiederversuchszeit-Konfiguration. Der Bereich dieses Parameters ist 0~7.

### Antwortrahmen:

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16-Relais	
				Wiederholungsversuch	LSB	MSB
0x06	0xXX	0x7B	0x00	0xXX		

Parameterdefinition

**RetryTimes:** der Wert der maximalen Schreib-Wiederholungszeit bit7 ~ bit3: reserviert, der Standardwert ist 0. bit2 ~ bit0: die Einstellung der maximalen Schreib-Wiederholungszeit. Der gültige Wiederversuchszeitraum beträgt 0~7.

Der Wiederholungstimer tritt für den folgenden Befehl in Kraft:

### EPC C1G2-Befehle (ISO18000-6C)

- den Befehl zum Speichern von Daten;
- geben Sie den EPC-Nummerbefehl ein;
- Erweiterter Befehl zum Schreiben von Daten mit einer zugewiesenen Maske; Block-Delete-Befehl;
- Blockiere den Tippbefehl.

### Kommando ISO18000-6B

- Write Data Befehl

Ändern Sie das Passwort der benutzerdefinierten Tag-Funktion

Die Funktion dieses Befehls besteht darin, das Passwort zu ändern, um auf tag-spezifische Funktionen zuzugreifen. Dieses Passwort tritt in Kraft, wenn du die Peek-Funktion des Monza4QT-Tags startest. Die Einstellung dieses Parameters bleibt erhalten, auch wenn der Leser keinen Zugang zu Strom hat.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
			UserPwd	LSB	MSB
0x08	0xXX	0x7D	4 Bytes	LSB	MSB

**UserPwd:** 4 Bytes, das bedeutendste Byte zuerst. Ein auf die Bedürfnisse des Nutzers zugeschnittenes Zugangspasswort.

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16-Relais	
0x05	0xXX	0x7D	0x00	--	LSB	MSB

Holen Sie sich das Passwort für die benutzerdefinierte Tag-Funktion

Die Funktion dieses Befehls besteht darin, ein Passwort zu erhalten, um auf tag-spezifische Funktionen zuzugreifen.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0x04	0xXX	0x7E	--	LSB	MSB

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16-Relais	
				UserPwd	LSB	MSB
0x09	0xXX	0x7E	0x00	4 Bytes	LSB	MSB

---

**UserPwd:** 4 Bytes, das bedeutendste Byte zuerst. Ein auf die Bedürfnisse des Nutzers zugeschnittenes Zugangspasswort.

### Leseprofil laden/verändern

Die Funktion dieses Befehls besteht darin, die Konfiguration des internen Leseprofiles zu ändern oder zu laden. Der Standardwert des Profils ist 1. Die Einstellung dieses Parameters bleibt erhalten, auch wenn der Leser keinen Zugang zu Strom hat.

### Befehlsrahmen:

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
			Profil		
0x05	0xXX	0x7F	0xXX	LSB	MSB

### Parameterdefinitionsprofi

! Konfigurieren Sie die Profilnummer.

Bit7: Aktion bei der Konfiguration der Profilnummer. 0 – Lastprofilnummer; 1 – Ändern Sie die

Profilnummer. bit6 ~ bit0:

Konfiguration der Profilnummer.

Für bit7 = 0 wird bit6 ~ bit0 ignoriert.

Für bit7 = 1 ist bit6 ~ bit0 die neue Profilnummernkonfiguration. Der gültige Bereich dieses Parameters ist 0~3.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

### Antwortrahmen:

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16-Relais	
				Profil		
0x06	0xXX	0x7F	0x00	0xXX	LSB	MSB

Parameterdefinitionsprofil: die aktuelle Konfiguration der Profilnummer. bit7: reserviert, der Standardwert ist 0. bit6 ~ bit0: die aktuelle Leserprofilnummer. Der gültige Bereich dieses Parameters ist 0~3.

### Konfiguration verschiedener Profile

Profile0: Tari 25uS, FM0 40KHz;

Profil1: Tari 25uS, Miller 4 250 kHz (empfohlene Konfiguration, Systemstandard) Profil 2: Tari 25uS, Miller 4 300 kHz; Profil 3: Tari 6,25uS, FM0 400KHz.

### Synchronisations-EM4325-Zeitstempel

Die Funktion dieses Befehls ist es, allen EM4325-Tags im effektiven Feld einen Zeitstempel zuzusenden.

### Befehlsrahmen:

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
			UTC-Zeit		
0x08	0xXX	0x85	4 Bytes	LSB	MSB

**UTC-Zeit**: 4 Bytes, UTC-Zeitstempel bis zum Broadcast, das wichtigste Byte zuerst. Dies ist der zweite Betrag seit dem 01.01.1970, 00:00:00. Die Synchronisation scheitert, wenn der UTC-MSB 0 ist.

### Antwortrahmen:

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16-Relais
-----	-----	--------------	---------	---------	---------------

0x05	0xXX	0x85	0x00	--	LSB	MSB
------	------	------	------	----	-----	-----

**HINWEIS:**

1. Das EM4325-Tag muss im BAP-Modus bleiben, um den Zeitstempel zu empfangen und zu synchronisieren.
2. Die Synchronisation scheitert, wenn das angegebene UTC-MSB 0 ist.
3. Die Synchronisation schlägt fehl, wenn ein Warnsignal gesetzt wird. Das Warnsignal kann mit dem Befehl "Reset EM4325 Alert" entfernt oder Alarmfunktionen vorübergehend deaktiviert werden, indem die zugehörigen EM4325-Register geändert werden.
4. EM4325-Tags erzeugen keine Antwort auf den Zeitstempel-Synchronisationsbefehl. Daher meldet der Leser einen erfolgreichen Host-Zustand, wenn der Zeitstempel-Synchronisationsbefehl an das Tag übermittelt wird.
5. Wenn du das interne UTC EM4325-Zeitregister einrichtest, erhöht sich dieses Register um 1 pro Sekunde. Dieser Timer stoppt, wenn das 4-Byte-Überlaufregister 0xFFFFFFFF und wieder auf 0 zurückkehrt oder das zugehörige Systemkonfigurationsregister im Benutzerbereich verändert wird.

Erhalten Sie EM4325 Temperaturdaten

Die Funktion dieses Befehls besteht darin, Temperaturdaten vom EM4325-Tag zu erhalten.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0xXX	0xXX	0x86	--	LSB	MSB

**Datenparameter[]:**

Daten[ ]				
ENum	Energieleistungszertifikat	SendUID (Send SendUID ID)	NewSample	Pwd
0xXX	Variable	0xXX	0xXX	4 Byte s
MaskMem	MaskAdr	MaskLen	MaskDaten	
0xXX	2Bytes	0xXX	Variable	

**ENum:**

0x00~0x0f – ENum steht für die Länge der EPC-Nummer. Die ENum-Einheit ist das Wort, und die EPC-Länge sollte weniger als 15 Wörter betragen. In dieser Situation sollten die Parameter MaskMem, MaskAdr, MaskLen und MaskData leer gelassen werden.

0xff – Anstelle von EPC-Parametern sollten MaskMem, MaskAdr, MaskLen und MaskData angegeben werden. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben ist.

**EPC:** Die EPC-Nummer des Ziel-Tags. Die Dauer des Energieeffizienzcertifikats hängt von der angegebenen EPC-Nummer ab. Die Einheit EPC-Nummer ist das Wort. Beim Kommunizieren werden zuerst das bedeutendste Wort und das bedeutendste Byte im Wort übertragen. Es ist unerlässlich, in diesem Kommando das VOLLSTÄNDIGE EPC bereitzustellen.

---

**SendUID:** 1 Byte, UID-  
Antwort-Flag. 0x00 –  
nicht antworten UID; 0x01  
– UID-Antwort.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**NewSample:** 1 Byte, neue  
Temperaturdaten-Flag. 0x00 –  
erhalte frühere Temperaturdaten;  
0x01 – beziehe neue  
Temperaturdaten.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**Pwd:** 4 Bytes, dies ist ein 4-Byte-Tag-Zugriffspasswort. Von links nach rechts sind die ersten zwei Bytes die oberste Anzahl von Bytes des Passcodes, und die letzten zwei Bytes sind die unteren Bytes. Das bedeutendste Bit eines 32-Bit-Tag-Zugriffspassworts ist das bedeutendste Bit in Pwd 1 Byte (von links nach rechts). Ebenso ist das am wenigsten signifikante Bit eines 32-Bit-Tag-Passcodes das am wenigsten signifikante Bit im vierten (letzten) Pwd-Byte. PSW sollte auf All-Zero oder auf den korrekten Passcode gesetzt werden.

**MaskMem:** 1 Byte, Anzeige des Maskenbereichs.

- 0x01 – EPC-Speicher;
- 0x02 – TID-Speicher;
- 0x03 – Benutzergedächtnis.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**MaskAdr:** 2 Bytes, Masken-Eingangsbitadresse, der gültige Bereich von MaskAdr ist 0~16383.

**MaskLen:** 1 Byte, Maskenbitlänge (Einheit: Bits).

**MaskData:** Maskendaten. Die Länge von MaskData entspricht MaskLen/8. Ist MaskLen kein Vielfaches einer ganzen Zahl von 8, so ist die Länge von MaskData gleich  $\text{int}[\text{MaskLen}/8]+1$ . Unbestimmte untere signifikante Ziffern sollten mit der Ziffer 0 gefüllt werden.

**HINWEIS:**

Wenn MaskMem, MaskAdr, MaskLen und MaskData nicht bereitgestellt werden, wird die vollständige EPC-Nummer als Maskenmuster im Inventar verwendet.

**Antwortrahmen:**

Nur	Adr	reCm d sagte:	Zustan d	Daten[]			CRC- 16- Relais	
				UID	Dane SensorDat a	UTC		
0xX X	0xX X	0x86	0x00	Variabl e	4Bytes	4Byte s	LS B	MS B

**UID:** Eindeutige Tag-ID EM4325.

Für SendUID = 0 wird in der Antwort kein UID-Feld angegeben;  
 Für SendUID=1 wird das UID-Feld in der Antwort angegeben, die Länge des Feldes hängt vom Tag ab; für verschiedene Tag-Antworten kann die UID-Länge zwischen 8 Bytes, 10 Bytes und 12 Bytes variieren.

**SensorData:** Temperaturdaten, die vom EM4325-Tag zurückgegeben werden, die Datenlänge beträgt 4 Bytes, das wichtigste Byte zuerst. Für eine detaillierte Beschreibung siehe die Definition des Sensordatenregisters im Datenblatt EM4325.

**UTC:** 4 Bytes, der aktuelle UTC-Zeitstempel des Tags, das bedeutendste Byte zuerst. Dies ist der zweite Betrag seit dem 01.01.1970, 00:00:00. Wenn das Ziel-Tag nicht synchronisiert wurde oder der UTC-Zeitstempel des Tags nicht konfiguriert wurde, liegt der Rückgabewert bei null.

Externe Daten über EM4325 SPI erhalten

Die Funktion dieses Befehls besteht darin, das EM4325-Tag als SPI-Mastergerät zu verwenden und Daten vom angeschlossenen SPI-Slave-Gerät zu erhalten.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0xXX	0xXX	0x87	--	LSB	MSB

**Datenparameter[]:**

Date n[]						
SPICmdSize	ENUM	Energieleistungszertifikat	SPIResSize (Rozmiar Wyspy)	SPISclk	SPIInitDelay	SPIInterval sagte:

0xXX	0xXX	Variable	0xXX	0xXX	0xXX	0xXX
SPEZIELLE SPEZIFIKATION.	Pwd	MaskMem	Mask Adr	Mask Len	Masken daten	
Variable	4 Bytes	0xXX	2 Bytes	0xXX	Variable	

**ENum:**

0x00~0x0f – ENum steht für die Länge der EPC-Nummer. Die ENum-Einheit ist das Wort, und die EPC-Länge sollte weniger als 15 Wörter betragen. In dieser Situation sollten die Parameter MaskMem, MaskAdr, MaskLen und MaskData leer gelassen werden.

0xff – Anstelle von EPC-Parametern sollten MaskMem, MaskAdr, MaskLen und MaskData angegeben werden. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben ist.

**EPC:** Die EPC-Nummer des Ziel-Tags. Die Dauer des Energieeffizienzcertifikats hängt von der angegebenen EPC-Nummer ab. Die Einheit EPC-Nummer ist das Wort. Beim Kommunizieren werden zuerst das bedeutendste Wort und das bedeutendste Byte im Wort übertragen. Es ist unerlässlich, in diesem Kommando das VOLLSTÄNDIGE EPC bereitzustellen.

**SPICmdSize:** 1 Byte, Byte-Länge des SPI-Befehls. Der gültige Bereich von SPICmdSize beträgt 1~8. Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

---

**SPIResSize:** 1 Byte, die Bytelänge der Antwort des SPI-Slave-Geräts. Ein gültiger SPIResSize-Bereich ist 0~7. SPIResSize = 0 bedeutet, dass keine Antwort erzeugt wird.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**SPIScIk:** 1 Byte, SPI-Bus-Uhr.

0 – 40 kHz;

1 – 80 kHz;

2 – 160 KHz; 3 – 320 kHz.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**SPIInitDelay:** 1 Byte, Verzögerung für die Initialisierung des SPI-Takts.

0 – 1 SCLK;

1 – 50 Jahre;

2 – 500 US; 3 – 5 ms.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**SPIInterval:** 1 Byte, Zeitintervall zwischen jedem Byte SPI-Daten.

0 – keine Verzögerungen;

1 – 50 Jahre;

2 – 100 US; 3 – 500 US.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

---

**SPICmd:** Die SPI-Befehlsdaten müssen identisch mit dem in SPICmdSize angegebenen Wert sein.

**Pwd:** 4 Bytes, dies ist ein 4-Byte-Tag-Zugriffspasswort. Von links nach rechts sind die ersten zwei Bytes die oberste Anzahl von Bytes des Passcodes, und die letzten zwei Bytes sind die unteren Bytes. Das bedeutendste Bit eines 32-Bit-Tag-Zugriffspassworts ist das bedeutendste Bit in Pwd 1 Byte (von links nach rechts). Ebenso ist das am wenigsten signifikante Bit eines 32-Bit-Tag-Passcodes das am wenigsten signifikante Bit im vierten (letzten) Pwd-Byte. PSW sollte auf All-Zero oder auf den korrekten Passcode gesetzt werden.

**MaskMem:** 1 Byte, Anzeige des Maskenbereichs.

0x01 – EPC-Speicher;

0x02 – TID-Speicher;

0x03 – Benutzergedächtnis.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**MaskAdr:** 2 Bytes, Masken-Eingangsbithadresse, der gültige Bereich von MaskAdr ist 0~16383.

**MaskLen:** 1 Byte, Maskenbitlänge (Einheit: Bits).

**MaskData:** Maskendaten. Die Länge von MaskData entspricht MaskLen/8. Ist MaskLen kein Vielfaches einer ganzen Zahl von 8, so ist die Länge von MaskData gleich  $\text{int}[\text{MaskLen}/8]+1$ . Unbestimmte untere signifikante Ziffern sollten mit der Ziffer 0 gefüllt werden.

**HINWEIS:**

Wenn MaskMem, MaskAdr, MaskLen und MaskData nicht bereitgestellt werden, wird die vollständige EPC-Nummer als Maskenmuster im Inventar verwendet.

---

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
				Dane SPIResData		
0xXX	0xXX	0x87	0x00	Variable	LSB	MSB

**SPIResData:** Die vom EM4325 SPI-Slave-Gerät zurückgegebenen Daten, der gültige Datenbereich beträgt 0~7 Bytes.

Die Länge von SPIResData sollte identisch mit dem in SPIResSize angegebenen Wert sein.

**HINWEIS:**

Bevor externe Daten über das EM4325 SPI erhalten werden, muss das Kontrollbyte-Register des EM4325 I/O Control Word (I/O Control Word) zunächst konfiguriert werden. Der Benutzer sollte die SPI-Funktion aktivieren, den SPI-Port konfigurieren und das SPI-Pin-Pull-up-Register aktivieren.

## Reset EM4325-Warnung

Die Funktion dieses Befehls besteht darin, die gesamte EM4325-Tag-Warnflagge zurückzusetzen. Die UTC-Uhr- und Überwachungsfunktionen sind frei.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16- Relais	
0xXX	0xXX	0x88	--	LSB	MSB

**Datenparameter[]:**

Daten[]						
ENu m	Energieleistungs- zertifikat	Pw d	MaskM em	Mask Adr	MaskL en	Maskend aten

0xX X	Variable	4 Byt es	0xXX	2 Bytes	0xXX	Variable
----------	----------	----------------	------	------------	------	----------

**ENum:**

0x00~0x0f – ENum steht für die Länge der EPC-Nummer. Die ENum-Einheit ist das Wort, und die EPC-Länge sollte weniger als 15 Wörter betragen. In dieser Situation sollten die Parameter MaskMem, MaskAdr, MaskLen und MaskData leer gelassen werden.

0xff – Anstelle von EPC-Parametern sollten MaskMem, MaskAdr, MaskLen und MaskData angegeben werden. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben ist.

**EPC:** Die EPC-Nummer des Ziel-Tags. Die Dauer des Energieeffizienzcertifikats hängt von der angegebenen EPC-Nummer ab. Die Einheit EPC-Nummer ist das Wort. Beim Kommunizieren werden zuerst das bedeutendste Wort und das bedeutendste Byte im Wort übertragen. Es ist unerlässlich, in diesem Kommando das VOLLSTÄNDIGE EPC bereitzustellen.

**Pwd:** 4 Bytes, dies ist ein 4-Byte-Tag-Zugriffspasswort. Von links nach rechts sind die ersten zwei Bytes die oberste Anzahl von Bytes des Passcodes, und die letzten zwei Bytes sind die unteren Bytes. Das bedeutendste Bit eines 32-Bit-Tag-Zugriffspassworts ist das bedeutendste Bit in Pwd 1 Byte (von links nach rechts). Ebenso ist das am wenigsten signifikante Bit eines 32-Bit-Tag-Passcodes das am wenigsten signifikante Bit im vierten (letzten) Pwd-Byte. PSW sollte auf All-Zero oder auf den korrekten Passcode gesetzt werden.

**MaskMem:** 1 Byte, Anzeige des Maskenbereichs.

0x01 – EPC-Speicher;

---

0x02 – TID-Speicher;

0x03 – Benutzergedächtnis.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**MaskAdr:** 2 Bytes, Masken-Eingangsbitaladresse, der gültige Bereich von MaskAdr ist 0~16383.

**MaskLen:** 1 Byte, Maskenbitlänge (Einheit: Bits).

**MaskData:** Maskendaten. Die Länge von MaskData entspricht MaskLen/8. Ist MaskLen kein Vielfaches einer ganzen Zahl von 8, so ist die Länge von MaskData gleich  $\text{int}[\text{MaskLen}/8]+1$ . Unbestimmte untere signifikante Ziffern sollten mit der Ziffer 0 gefüllt werden.

**HINWEIS:**

Wenn MaskMem, MaskAdr, MaskLen und MaskData nicht bereitgestellt werden, wird die vollständige EPC-Nummer als Maskenmuster im Inventar verwendet.

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0x05	0xXX	0x88	0x00	--	LSB	MSB

**HINWEIS:**

Bevor Sie den Befehl "Reset EM4325 alert" verwenden, müssen Sie zuerst das EM4325I Temperatursensor-Steuerbyte (Temp Sensor Control Word 1) konfigurieren, um die EM4325-Alarm-Reset-Funktion zu aktivieren; ansonsten erzeugt das EM4325-Tag keine Antwort auf diesen Befehl.

---

## Ändern oder laden Sie DRM-Konfigurationen

Die Funktion dieses Befehls besteht darin, den DRM-Switch-Modus zu modifizieren oder zu laden. Die Einstellung dieser Parameter geht verloren, wenn der Strom abgeschaltet wird. Das DRM wird jedes Mal, wenn der Leser gestartet wird, auf den deaktivierten Zustand zurückgesetzt.

### Befehlsrahmen:

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
			DRMM-Code		
0x05	0xXX	0x90	0xXX	LSB	MSB

**DRMMode:** Aktion auf DRM-Konfiguration. bit7: DRM-Flag laden/ändern.

0 – die aktuelle DRM-Konfiguration laden; 1 – die DRM-Konfiguration modifizieren. bit6 ~ bit0: DRM-Konfiguration  
Für bit7 = 0 wird bit6 ~ bit0 ignoriert.  
Für bit7 = 1 ist bit6 ~ bit0 die gewünschte DRM-Konfiguration. 0 – DRM deaktivieren; 1 – DRM aktivieren.

Andere Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn ein anderer Wert in diesem Feld angegeben wird.

### Antwortrahmen:

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16-Relais	
				DRMM-Code		
0x06	0xXX	0x90	0x00	0xXX	LSB	MSB

**DRMMode:** Aktion bei der DRM-Konfiguration. bit7: reserviert, Standardeinstellung 0.

bit6 ~ bit0: aktuelle DRM-Konfiguration 0 – DRM ist deaktiviert; 1 – DRM ist aktiviert.

### Messung der Antennenreflexionsdämpfung

Die Funktion dieses Befehls besteht darin, die Reflexionsdämpfung der Antenne an einem bestimmten Frequenzpunkt zu messen.

### Befehlsrahmen:

Nur	Adr	Cmd	Datumsstempel $\alpha$ []		CRC-16-Relais	
			Testfrequenz	Ameise		
0x09	0xXX	0x91	4 Bytes	0xXX	LSB	MSB

### Parameterdefinition

**TestFreq:** Zielfrequenzpunkt für die Messung der Reflexabschwächung. Die Einheit von TestFreq ist KHz, das wichtigste Byte zuerst. Der korrekte Frequenzpunkt muss ein Vielfaches von 125 kHz oder 100 kHz sein. Der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben ist.

**Ameisen:** 1 Byte, Zielantenne für diese Messung  
0 – Antenne 1;

- 1 – Antenne 2;
- 2 – Antenne 3; 3 – Antenne 4.

Andere Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn ein anderer Wert in diesem Feld angegeben wird.

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]		CRC-16-Relais	
				ReturnLoss		LSB	MSB
0x06	0xXX	0x91	0x00	0xXX		LSB	MSB

**ReturnLoss:** Das Ergebnis der Reflexionsverlustmessung, die Einheit von ReturnLoss ist dB.

Messen Sie die Stromtemperatur des Lesers

Die Funktion dieses Befehls besteht darin, die aktuelle Temperatur des Lesers zu bestimmen.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0x04	0xXX	0x92	--	LSB	MSB

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]		CRC-16-Relais	
				PlusMinus (zusätzliche s Minus)	Höchste Temperatur	LSB	MSB
0x07	0xXX	0x92	0x00	0xXX	0xXX	LSB	MSB

**PlusMinus:** 1 Byte, Temperaturberei

ch. 0 – unter 0  
°C; 1 – über 0 °C.

**Temperatur:** 1 Byte, Temperaturwert, Einheit ist °C.

**HINWEIS:**

Dieser Befehl wird nur vom eingebauten Temperatursensor-Leser unterstützt.

Einstellung erweiterter Parameter

Der Host-Computer wird verwendet, um verschiedene erweiterte Leseparameter zu konfigurieren, unterstützt derzeit nur die Ex10-Serie.

**Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]			CRC-16-Relais	
0x06+n	0xXX	0xEA	Flagge	CFGNo-Zertifikat	CFGData (n Bytes)	LSB	MSB

**Flag:** 1 Byte, 0 – speichern beim Ausschalten; 1 – nicht speichern beim Ausschalten;

**CFGNo:** 1 Byte, wird verwendet, um die Sequenznummer des im Befehlsdatenblock festgelegten Parameters anzugeben.

**CFGData:** Parameterinhalte, siehe [Konfigurieren von Reader-Parametern](#) für Details.

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16-Relais	
0x05	0xXX	0xEA	0x00	--	LSB	MSB

---

## Einstellung erweiterter Parameter

Der Host-Computer wird verwendet, um erweiterte Informationen über die Leseparameter zu erhalten, unterstützt derzeit nur die Ex10-Serie

### **Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
			CFGNo-Zertifikat		
0x05	0xXX	0xEB	0xXX	LSB	MSB

**CFGNo:** 1 Byte, wird verwendet, um die Sequenznummer des im Befehlsdatenblock festgelegten Parameters anzugeben.

### **Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16-Relais	
				CFGData (auf Englisch)		
0x05+n	0xXX	0xEB	0x00	n(Byte)	LSB	MSB

**CFGData:** Parameterinhalte, siehe [Konfigurieren von Reader-Parametern](#) für Details.

## Starte eine schnelle Inventur

Der Host-Computer wird verwendet, um eine schnelle Zählung durchzuführen, unterstützt derzeit nur **den Befehlsrahmen der Ex10-**

### **Serie:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
			Zweck		
0x05	0xXX	0x50	0xXX	LSB	MSB

**Ziel:** 1 Byte, 0 – Ziel A; 1-Schild B;

**Antwortrahmen:**

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0x05	0xXX	0x50	0x00	--	LSB	MSB

Wenn Sie nach einem Tag gefragt werden, der einer vordefinierten Bedingung entspricht, sendet der Leser die Daten automatisch im folgenden Format:

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16- Relais	
0xXX	0xXX	0xee	0x00	--	LSB	MSB

die Parameter der Daten[] sind wie folgt dargestellt,

Daten[]			
Ameise	Nur	EPC/TID- Zertifikat	RSSI- Indikator (RSSI)
0xXX	0xXX	N Bytes	0xXX

**Ameise:** Gibt an, welche Antenne die Tag-Anfrage gesendet hat. Zum Beispiel ist 0x05 im Binärcode 0101, was für Antenne 3 und Antenne 1 steht.

**Len:** 1 Byte, EPC/TID-Bytelänge.

**EPC/TID:** EPC-Nummer/TID-Tag-Daten, die Datenlänge sollte identisch mit der von Len sein. Die EPC C1 G2 Dateneinheit ist das Wort. Jede EPC-Tagnummer oder TID-Daten ist das bedeutendste Wort und das bedeutendste Byte im Wort zuerst.

**RSSI:** 1 Byte, die Signalstärke des Lesers zum Zeitpunkt der Tag-Abfrage.

---

## Stopp Quick Inventory

Der Host-Computer wird verwendet, um das schnelle Inventar zu stoppen, unterstützt derzeit nur die ExI0-Serie. **Befehlsrahmen:**

Nur	Adr	Cmd	Daten[]	CRC-16-Relais	
0x04	0xXX	0x51	--	LSB	MSB

### Antwortrahmen:

Nur	Adr	reCmd sagte:	Zustand	Daten[]	CRC-16-Relais	
0x05	0xXX	0x51	0x00	--	LSB	MSB

## Individuelle Lesebefehle

Alle Konfigurationsparameter des Readers sind darin aufgeführt, was für Benutzer praktisch ist, um einheitlich einzustellen und zu lesen.

Um Parameter zu lesen, verwenden Sie den Befehl `get reader parameter` und verwenden Sie den `set reader configuration` befehl, um die Parameter festzulegen. Wenn eine Konfiguration geändert wird, treten die anderen Konfigurationen sofort in Kraft, abgesehen von einer Netzwerkkonfiguration, die neu gestartet werden muss, um angewendet zu werden. (Unterstützt derzeit nur die ExI0-Serie)

## Tabelle der Leserkonfigurationsparameter

### Konfigurationstabelle:

Nein.	Name	Länge(Byte)	Beschreibung
7	Scan-Parameter	3	Über die im Scanprozess verwendeten Parameter
8	TagFocus	1	TagFocus-Schalter

9	QAndSession (Qualitätskontroll Sitzung)	2	Q und Sitzungsparameter
10	DATENSCHUTZPARAMETER	2	TID-Parameter
11	Maskenparameter	N	Maskenparameter

Beschreibung der Funktion des Leserparameters

### CFG7: Scan-Parameter

Verwenden Sie diese Konfiguration, um die vom Leser während des Scanvorgangs verwendeten Parameter einzustellen

Byte	0	1	2
Inhalt	Intervall	Verweilzeit	Graf
Default	3	0	5

**Intervall:** Intervallzeit zwischen Abfragen, Bereich 0~6, Einheit 10 ms;

**Verweilzeit:** Abfragezeit, der Bereich beträgt 2~255, die Einheit 100 ms; wenn der TagFocus-Schalter eingeschaltet ist, liegt er bei 200 ms.

**Zahl:** die Anzahl der Abfragen, nur gültig für S1, S2, S3, der Bereich ist 0~5;

### CFG8: TagFocus-Schalter

Verwenden Sie diese Konfiguration, um den TagFocus-Schalter einzustellen

Byte	0
Inhalt	TagFocus
Default	0

**TagFocus:** 0-aus, 1-an.

### CFG9: Frage und Sitzung

Verwenden Sie diese Konfiguration, um die Anti-Kollisionsparameter einzustellen, die der Leser während des Quick Query-Prozesses verwendet.

Byte	0	1
Inhalt	Q	Sitzung
Default	6	1

**P:** Der Bereich ist 0~15.

**Sitzung:** 0-S0; 1-S1; 2-S2; 3-S3; 255-Auto;

### CFG10: TID-Parameter

Verwenden Sie diese Konfiguration, um die vom Leser verwendeten TID-Parameter während des Quick Query-Prozesses einzustellen.

Byte	0	1
Inhalt	AdrTID sagte:	LenTID
Default	6	1

**AdrTID:** Fragen Sie nach der Startadresse des TID-Gebietswortes.

**LenTID:** Abfrage der Anzahl der Wörter im TID-Bereich. LenTID liegt im Bereich von 0 bis 15. Wenn der Wert 0 ist, bedeutet das eine EPC-Abfrage.

### CFG11: Maskenparameter

Verwenden Sie diese Konfiguration, um den Maskenparameter festzulegen, den der Reader während des Quick Query-Prozesses verwendet.

Byte	MaskMem	MaskAdr	MaskLen	Maskendaten
------	---------	---------	---------	-------------

---

Inhalt	1	2	1	MaskLen
Default	1	0x0020	0	0x00

**MaskMem:** 1 Byte, Anzeige des Maskenbereichs.

0x01 – EPC-Speicher;

0x02 – TID-

Speicher; 0x03 –

Benutzergedächt

nis.

Alle anderen Werte sind reserviert, der Leser gibt den Fehlerstatus des Parameters im Antwortrahmen zurück, wenn in diesem Feld ein anderer Wert angegeben wird.

**MaskAdr:** 2 Bytes, Masken-Eingangsbitadresse, der gültige Bereich von MaskAdr ist 0~16383.

**MaskLen:** 1 Byte, Maskenbitlänge (Einheit: Bits).

**MaskData:** Maskendaten. Die Länge von MaskData entspricht MaskLen/8. Ist MaskLen kein Vielfaches einer ganzen Zahl von 8, so ist die Länge von MaskData gleich  $\text{int}[\text{MaskLen}/8]+1$ . Unbestimmte untere signifikante Ziffern sollten mit der Ziffer 0 gefüllt werden.

