

S3000 Standard, Advanced,
Professional, Remote

S300 Professional



Safety Laser Scanner



D
GB

Dieses Werk ist urheberrechtlich geschützt. Die dadurch begründeten Rechte bleiben bei der Firma SICK AG. Eine Vervielfältigung des Werkes oder von Teilen dieses Werkes ist nur in den Grenzen der gesetzlichen Bestimmungen des Urheberrechtsgesetzes zulässig. Eine Abänderung oder Kürzung des Werkes ist ohne ausdrückliche schriftliche Zustimmung der Firma SICK AG untersagt.

Inhalt

1	Zu diesem Dokument.....	4
1.1	Funktion dieses Dokuments.....	4
1.2	Zielgruppe.....	4
1.3	Informationstiefe.....	4
1.4	Geltungsbereich	4
2	Systembeschreibung	5
2.1	Systemaufbau	5
2.2	Elektrische Schnittstelle	6
2.2.1	Übertragungs- und Datenformat	6
2.3	Telegrammaufbau.....	6
2.3.1	Command-Telegramm	7
2.3.2	Reply-Telegramm.....	7
2.3.3	Fehlercodes des Reply-Telegramms	8
2.4	Standard-Kommunikation	9
2.5	Flexible Telegramme	10
2.6	Kontinuierliche Ausgabe.....	10
2.7	Datensicherung.....	11
3	Kommunikationsarten.....	13
3.1	Geräte-Adressen.....	13
3.2	System Token.....	13
3.3	Request Mode	13
3.4	Kontinuierliche Datenausgabe.....	15
3.5	Beispiel für kontinuierliche Datenausgabe	16
3.6	Anhalten der kontinuierlichen Datenausgabe.....	17
3.7	Fest konfigurierte Modi der kontinuierlichen Datenausgabe.....	17
3.8	Zusätzliche Zeitstempel und Telegramm-Nummern	17
4	Messdatenausgabe.....	18
4.1	CDS-Bedienoberfläche.....	18
4.2	Konfiguration der Messdatenausgabe	18
5	Fehlerdiagnose	19
6	Glossar.....	20
7	Appendix.....	41

1 Zu diesem Dokument

Bitte lesen Sie dieses Kapitel sorgfältig, bevor Sie mit der Dokumentation und dem S3000 bzw. S300 arbeiten.

1.1 Funktion dieses Dokuments

Das vorliegende Telegrammlisting Standard beschreibt die messdatenspezifische Funktionserweiterung des S3000 in Verbindung mit den Modulen Standard, Advanced, Professional und Remote sowie des S300 Professional. Es ist als Ergänzung zur Betriebsanleitung S3000 bzw. S300 zu verstehen.



ACHTUNG

Für allgemeine Informationen, wie beispielsweise zum Anbau, zur Installation und Inbetriebnahme des Sicherheits-Laserscanners, verwenden Sie bitte die Betriebsanleitung S3000 bzw. S300. Beachten Sie bitte die darin beschriebenen Sicherheitshinweise in Kapitel 2 und 8, bevor Sie die Anlage in Betrieb nehmen.

Die vom Laserscanner zur Verfügung gestellten Telegrammdaten dürfen nicht für Sicherheitsanwendungen verwendet werden!

1.2 Zielgruppe

Das Telegrammlisting richtet sich an Systemspezialisten im Bereich der Hard- und Softwareentwicklung, die die scannereigenen Messwertdaten in ihre Hostapplikation einbinden und auswerten wollen.

1.3 Informationstiefe

Das vorliegende Telegrammlisting enthält Informationen zu folgenden Themen:

- Beschreibung der RS 422- Schnittstelle
- Beschreibung des verwendeten RK512-Protokolls
- Beschreibung der scannerspezifischen Sonderfunktionen
- Hinweise zum Einsatz
- Fehlerdiagnose

1.4 Geltungsbereich

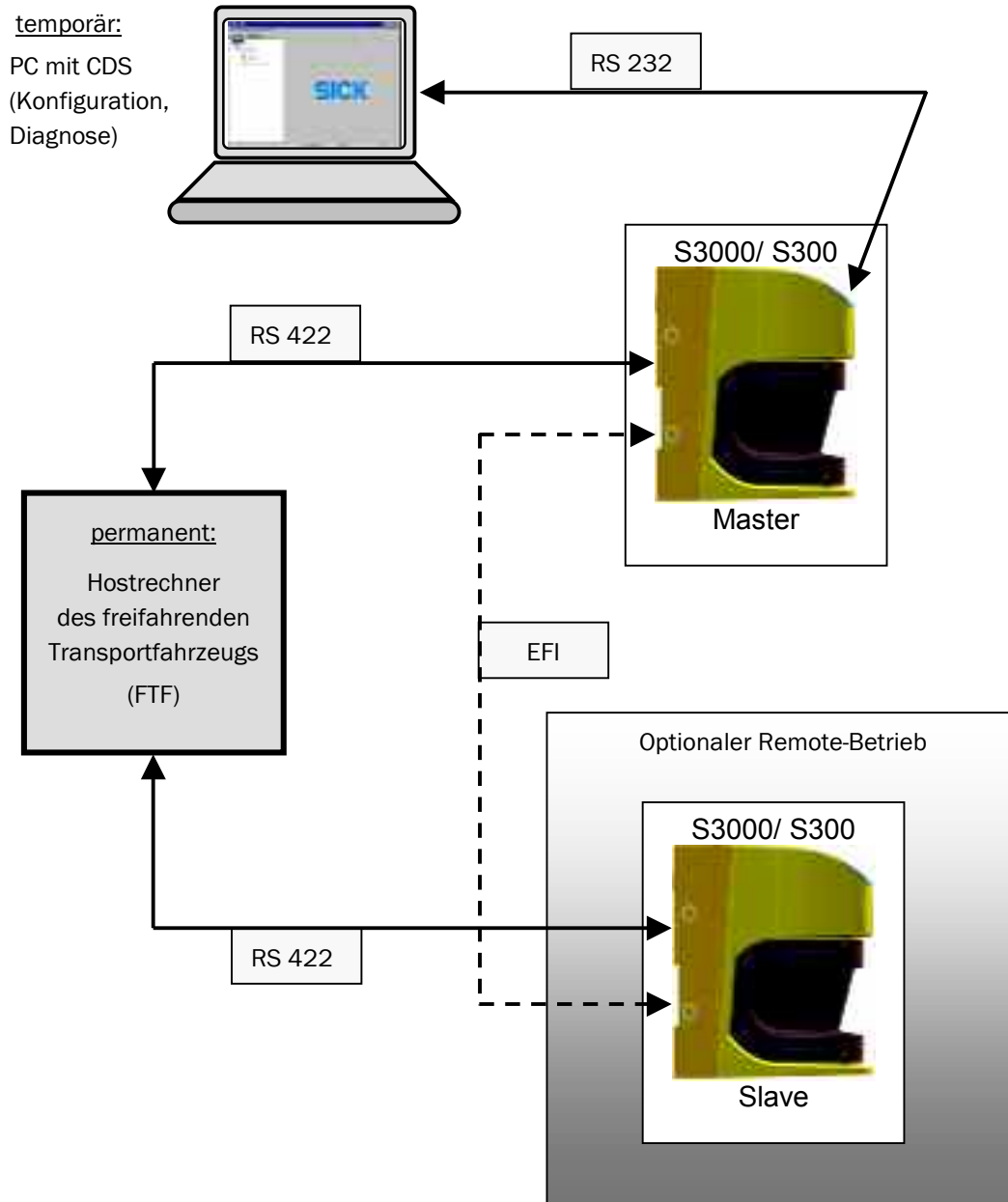
Das vorliegende Telegrammlisting Standard ist anwendbar für den Laserscanner S3000 in Verbindung mit den Modulen Standard, Advanced, Professional und Remote sowie auch für den Laserscanner S300 Professional mit folgenden Typenbezeichnungen:

S3000: S30A-XXXX XA

S300: S30B-XXXX XA.

2 Systembeschreibung

2.1 Systemaufbau



2.2 Elektrische Schnittstelle

Die elektrische Schnittstelle ist nach EIA RS-422-A Standard implementiert.

Die elektrische Anschlussmöglichkeit finden Sie in der S3000/ S300 Betriebsanleitung im Kapitel „Elektroinstallation“.

2.2.1 Übertragungs- und Datenformat

Die Baudrate an der RS 422-Messdatenausgabe ist mit Hilfe der CDS einstellbar zwischen folgenden Baudraten:

- 9600 Baud
- 19200 Baud
- 38400 Baud
- 115,2 kBaud (nur S300)
- 125 kBaud
- 230,4 kBaud (nur S300)
- 250 kBaud
- 460,8 (nur S300)
- 500 KBaud

Im Auslieferungszustand ist die Schnittstelle mit einer Baudrate von 125 kBaud vorkonfiguriert.

Ein Datenbyte setzt sich zusammen aus 1 Startbit, 8 Datenbits, 1 Stopbit, kein Paritybit

2.3 Telegrammaufbau

Die Kommunikation nach dem RK512-Standard basiert auf Command- und Reply-Telegrammen. Ein Command-Telegramm ist entweder ein Send- oder ein Fetch-Telegramm.

Bei Send-Telegrammen schickt der Hostrechner nach dem Telegrammkopf die zu übertragenen Daten, der Empfänger antwortet mit einem Reply-Telegramm ohne weitere Daten. Bei Fetch-Telegrammen schickt der Hostrechner den Kopf eines Fetch-Telegramms ohne nachfolgende Daten, und der Sensor antwortet mit einem Reply-Telegramm, welches nach dem Telegrammkopf die angeforderten Daten enthält.

Auch im EFI-Verbund ist der Hostrechner immer der aktive Teilnehmer. Die Sensoren senden von sich aus keine RK512-Telegramme. Damit entfällt auch ein möglicher Initialisierungskonflikt bei gleichzeitiger Kommunikationsaufnahme.

Der Wert im Size Field bezeichnet immer die Anzahl von 16-bit Words, es kann nicht auf einzelne Bytes zugegriffen werden.

Im Telegrammkopf werden Werte, die 2 Byte umfassen mit dem High Byte (HB) zuerst übertragen. Bei den Datenbytes werden die 16-bit Words mit dem Low Byte (LB) zuerst übertragen.

2.3.1 Command-Telegramm

Der Telegrammkopf des Command-Telegramms (Befehlstelegramm) besteht aus 10 Bytes welche folgende Bedeutung haben:

Byte	Telegram Fields	Inhalt	Bedeutung
1	Telegram Identifier	0x00	
2		0x00	
3	Command - Telegram Type	‚A‘ (0x41) oder ‚E‘ (0x45)	Send-Telegram oder Fetch-Telegram
4	Command - Data Type	‚D‘ (0x44)	
5	Destination Address/ Source Address	0x00 bis 0xFF	Data Block Nummer
6		0x00	
7	Size	0 bis 65535	Blockgröße in Word
8			
9	Coordination Flag (Byte Number)	0xFF	
10	Device Address	0 bis 15	0x07 S 3000_1 (S 300_1) 0x08 S 3000_2 (S 300_2)

Tabelle: Aufbau Command-Telegramm

2.3.2 Reply-Telegramm

Der Telegrammkopf des Reply-Telegramms (Reaktionstelegramm) besteht aus 4 Bytes, welche folgende Bedeutung haben:

Byte	Telegram Fields	Inhalt	Bedeutung
1	Telegram Identifier	0x00 (immer)	
2		0x00 (immer)	
3	Reply - Telegram Type	0x00 (immer)	Reply - Telegram
4	Reply - Error Number	0x00 0x01 bis 0xFF	Kein Fehler Fehlernummer siehe Fehlertabelle

Tabelle: Aufbau Reply-Telegramm

2.3.3 Fehlercodes des Reply-Telegramms

Das Reply-Telegramm ist die Antwort des S3000/ S300 auf ein Send- oder Fetch-Telegramm.

Stellt der S3000/ S300 einen Fehler fest, zeigt er dies in der Error Number des Reply-Telegramms an, und es werden keine Daten nach dem Reply-Telegrammkopf geschickt.

Fehlercode im Reply Telegramm	RK512-Protokoll Kommunikations-Fehler
0x00	Kein Fehler
0x0A	Eine der Überwachungen der Kommunikation ist fehlgeschlagen (u.a. Timeout der EFI-RK512 Pakets / EFI-RK512 Acknowledge oder fehlerhafte Übertragung der EFI-RK512 Pakets/ EFI-RK512 Acknowledge)
0x0C	Die Data Word Number der Destination Address bzw. Source Address im Befehlstelegramm (Byte 6) ist unzulässig (nicht im Register Interface definiert)
	Das Coordination Flag (Byte Number) im Befehlstelegramm (Byte 9) ist ungleich 0xFF
	Der Device Code im Befehlstelegramm (Byte 10, Bit 0 bis 3) ist ungültig (d.h. ist gleich 0)
	Die CPU Number im Befehlstelegramm (Byte 10, Bit 5 bis 7) ist unzulässig
0x10	Das Telegram Identifier im Befehlstelegramm (Byte 1) ist ungleich 0x00 oder 0xFF oder wird nicht gefolgt von einem weiteren Byte 0x00 (Byte 2)
	Der Command Data Type im Befehlstelegramm (Byte 4) ist unzulässig
0x14	Die Data Block Number der Destination Address bzw. Source Address im Befehlstelegramm (Byte 5) ist unzulässig (nicht im Register Interface definiert)
0x16	Der Command Telegram Type im Befehlstelegramm (Byte 3) ist unzulässig
0x34	Telegramm-Format-Fehler Mögliche Ursachen: <ul style="list-style-type: none"> • Die im Size Field des Befehlstelegramms angegebene Länge (Byte 7 und 8) übersteigt die im Data Block ab der angegebenen Adresse verfügbaren Einträge • Die Länge der Daten des Send-Telegramms war größer als im Size Field angegeben • Die Länge der Daten des Send-Telegramms war kleiner als im Size Field angegeben • Es wurde ein Fetch-Telegramm mit Nutzdaten empfangen • Das Send-Telegramm wurde korrekt empfangen, aber die CRC über die Daten ist falsch bzw. die ersten sechs Datenbytes stimmen nicht mit den Bytes 5 bis 10 aus dem Telegrammkopf überein
0x36	Es wurde ein Befehlstelegramm empfangen, obwohl noch kein Reaktionstelegramm empfangen wurde

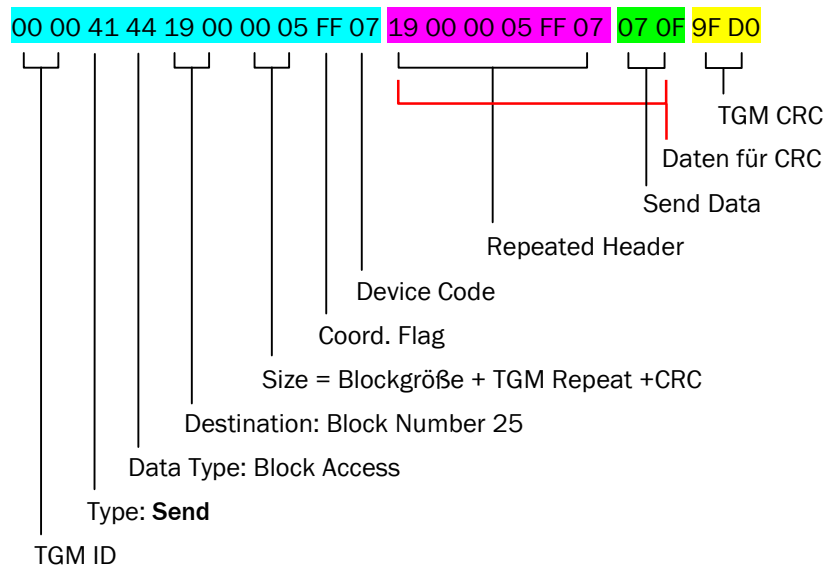
Fehlercode im Reply Telegramm	RK512-Protokoll Kommunikations-Fehler
0x01	Der aktuelle Status des Geräts erlaubt keinen Zugriff auf den Data Block
0x02	Der Zugriff auf den Data Block ist der aktuellen Benutzergruppe nicht erlaubt
0x03	Das Passwort ist nicht korrekt
0x04	Der System-Token ist belegt
0x05	Parameter nicht korrekt

Tabelle: Fehlercodes Reply-Telegramm

2.4 Standard-Kommunikation

Beispiel für ein Send-Telegramm „Schreiben des Token in Block 25“

Hostrechner send:

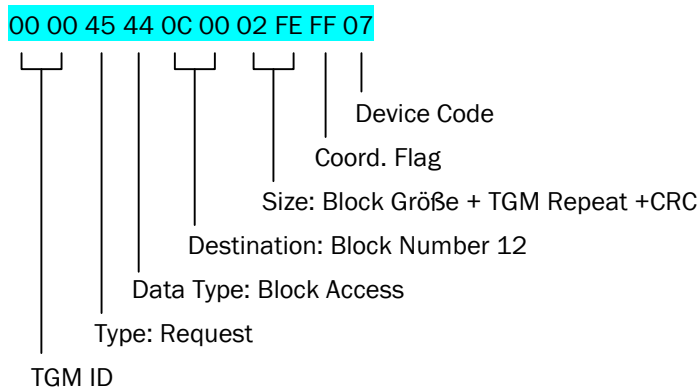


S3000 Reply:

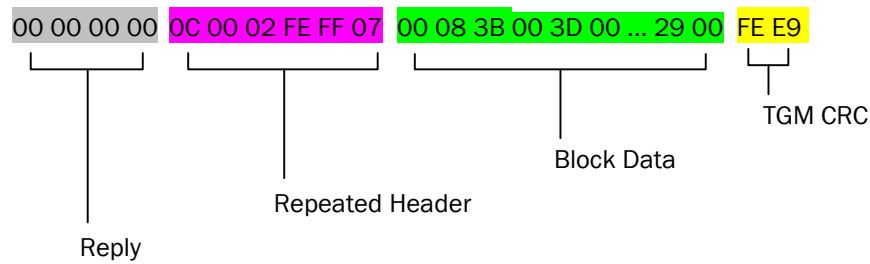
00 00 00 00

Beispiel für ein Fetch-Telegramm „Lesen des Operating Data Block 12“

Hostrechner send:



S3000 Reply



2.5 Flexible Telegramme (nur S3000)

Bei einigen Hostrechnern besteht die Forderung, die Interrupt-Belastung für die UART so gering wie möglich zu halten. In solchen Fällen kann (Request Mode) ein Datenblock mit flexibler Länge im Reply-Telegramm (Block 112) verwendet werden.

Hinweis Im EFI-Verbund ist der Block 112 nur direkt über einen an der RS 422 angeschlossenen Sensor verfügbar.

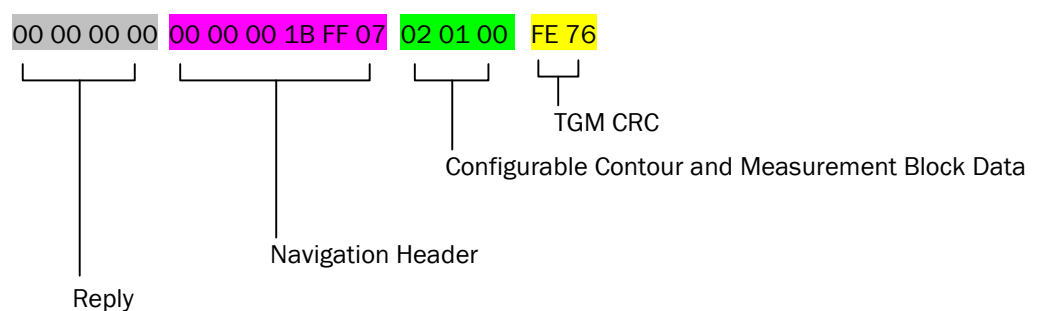
2.6 Kontinuierliche Ausgabe

Um einen möglichst effizienten Telegrammverkehr aufbauen zu können, kann der S3000/S300 so konfiguriert werden, dass er die Messwerte und einige Zusatzinformationen permanent an der RS 422-Schnittstelle aussendet. Die kontinuierliche Ausgabe wird nicht durch den Telegrammverkehr über die anderen zur Verfügung stehenden Schnittstellen beeinträchtigt. Für die kontinuierliche Datenausgabe ist es nicht erforderlich, dass der Token für die RS 422 Schnittstelle reserviert wird (bei Request Mode kann immer nur eine Schnittstelle den Token besitzen).

Um bei kontinuierlicher Datenausgabe eine Request-Anforderung vom Hostrechner über die RS 422 Schnittstelle senden zu können (z. B. Ändern der Trigger-Modi), muss zuvor die kontinuierliche Datenausgabe gestoppt werden (siehe Kap. 3.6: Anhalten der kontinuierlichen Datenausgabe).

Beispiel: Telegrammaufbau kontinuierliche Datenausgabe

00 00 00 00 00 00 00 1B FF 07 02 01 00 00 17 01 00 00 00 00
 CC CC 08 00 00 00 E8 23 19 00 E8 23 32 00 E8 23 7D 00 E8 23 96 00 E8 23 C8 00 E8
 23 E1 00 E8 23 38 4A E8 23 FE 76



Hinweis

Im EFI-Verbund ist die kontinuierliche Ausgabe der Messwerte nur direkt über einen an der RS 422 angeschlossenen Sensor möglich.

2.7 Datensicherung

Da der RK512-Standard keinen Mechanismus zur Überprüfung der Integrität der empfangenen Daten zur Verfügung stellt, ist ein solcher Mechanismus in den Daten des RK512 Telegramms ergänzt. Dieser umfasst die exakte Wiederholung der Bytes 5 bis 10 vom Kopf eines Command-Telegramms in den ersten sechs Datenbytes sowie eine CRC, die über die Datenbytes gebildet wird und an diese angehängt wird. Demnach muss die Anzahl der Words, die im Size Field des Telegrammkopfs eines Command-Telegramms angegeben wird, um 4 Words erhöht werden.

Die CRC umfasst 16 Bit und wird nach dem Polynom $x^{16}+x^{12}+x^5+x^0$ (0x1021) gebildet. Diese CCITT-CRC wird im gesamten Kommunikationsstack angewandt, wo eine CRC zur Absicherung der übertragenen Daten benötigt wird.

Nachfolgend eine einfache Routine zur CRC-Berechnung als Beispiel:

```

static const unsigned short crc_table[256] = {
0x0000, 0x1021, 0x2042, 0x3063, 0x4084, 0x50a5, 0x60c6, 0x70e7,
0x8108, 0x9129, 0xa14a, 0xb16b, 0xc18c, 0xd1ad, 0xe1ce, 0xf1ef,
0x1231, 0x0210, 0x3273, 0x2252, 0x52b5, 0x4294, 0x72f7, 0x62d6,
0x9339, 0x8318, 0xb37b, 0xa35a, 0xd3bd, 0xc39c, 0xf3ff, 0xe3de,
0x2462, 0x3443, 0x0420, 0x1401, 0x64e6, 0x74c7, 0x44a4, 0x5485,
0xa56a, 0xb54b, 0x8528, 0x9509, 0xe5ee, 0xf5cf, 0xc5ac, 0xd58d,
0x3653, 0x2672, 0x1611, 0x0630, 0x76d7, 0x66f6, 0x5695, 0x46b4,
0xb75b, 0xa77a, 0x9719, 0x8738, 0xf7df, 0xe7fe, 0xd79d, 0xc7bc,
0x48c4, 0x58e5, 0x6886, 0x78a7, 0x0840, 0x1861, 0x2802, 0x3823,
0xc9cc, 0xd9ed, 0xe98e, 0xf9af, 0x8948, 0x9969, 0xa90a, 0xb92b,
0x5af5, 0x4ad4, 0x7ab7, 0x6a96, 0x1a71, 0x0a50, 0x3a33, 0x2a12,
0xdbfd, 0xcdbc, 0xfbff, 0xeb9e, 0x9b79, 0x8b58, 0xbb3b, 0xab1a,
0x6ca6, 0x7c87, 0x4ce4, 0x5cc5, 0x2c22, 0x3c03, 0x0c60, 0x1c41,
0xedae, 0xfd8f, 0xcdec, 0xddcd, 0xad2a, 0xbd0b, 0x8d68, 0x9d49,
0x7e97, 0x6eb6, 0x5ed5, 0x4ef4, 0x3e13, 0x2e32, 0x1e51, 0x0e70,
0xff9f, 0xefbe, 0xdfdd, 0xcffc, 0xbf1b, 0xaf3a, 0x9f59, 0x8f78,
0x9188, 0x81a9, 0xb1ca, 0xa1eb, 0xd10c, 0xc12d, 0xf14e, 0xe16f,
0x1080, 0x00a1, 0x30c2, 0x20e3, 0x5004, 0x4025, 0x7046, 0x6067,
0x83b9, 0x9398, 0xa3fb, 0xb3da, 0xc33d, 0xd31c, 0xe37f, 0xf35e,
0x02b1, 0x1290, 0x22f3, 0x32d2, 0x4235, 0x5214, 0x6277, 0x7256,
0xb5ea, 0xa5cb, 0x95a8, 0x8589, 0xf56e, 0xe54f, 0xd52c, 0xc50d,
0x34e2, 0x24c3, 0x14a0, 0x0481, 0x7466, 0x6447, 0x5424, 0x4405,
0xa7db, 0xb7fa, 0x8799, 0x97b8, 0xe75f, 0xf77e, 0xc71d, 0xd73c,
0x26d3, 0x36f2, 0x0691, 0x16b0, 0x6657, 0x7676, 0x4615, 0x5634,
0xd94c, 0xc96d, 0xf90e, 0xe92f, 0x99c8, 0x89e9, 0xb98a, 0xa9ab,
0x5844, 0x4865, 0x7806, 0x6827, 0x18c0, 0x08e1, 0x3882, 0x28a3,
0xcb7d, 0xdb5c, 0xeb3f, 0xfb1e, 0x8bf9, 0x9bd8, 0xabbb, 0xbb9a,
0x4a75, 0x5a54, 0x6a37, 0x7a16, 0x0af1, 0x1ad0, 0x2ab3, 0x3a92,
0xfd2e, 0xed0f, 0xdd6c, 0xcd4d, 0xbdaa, 0xad8b, 0x9de8, 0x8dc9,
0x7c26, 0x6c07, 0x5c64, 0x4c45, 0x3ca2, 0x2c83, 0x1ce0, 0x0cc1,
0xef1f, 0xff3e, 0xcf5d, 0xdf7c, 0xaf9b, 0xbfba, 0x8fd9, 0x9ff8,
0x6e17, 0x7e36, 0x4e55, 0x5e74, 0x2e93, 0x3eb2, 0x0ed1, 0x1ef0
};
WORD CRC16 (BYTE *Data, DWORD length)
{
    WORD CRC_16 = 0xFFFF;
    WORD i;
    for (i = 0; i < length; i++)
    {
        CRC_16 = (CRC_16 << 8) ^ (crc_table[(CRC_16 >> 8) ^ (Data[i])]);
    }
    return CRC_16;
}

```

3 Kommunikationsarten

3.1 Geräte-Adressen

Die Kommunikationsadressen für S 3000_1 und S 3000_2 (Device Code) sind für die Kommunikation der Geräte im Remote-Betrieb notwendig.

Für Stand-Alone-Applikationen und Applikationen mit UE100 sowie UE1000-Familie ist die Geräteadresse obligatorisch 0x07.

Im Remote-Betrieb sind die Adressen wie folgt aufgeteilt:

S 3000_1 Adresse 0x07

S 3000_2 Adresse 0x08

Diese Adresse wird im RK512 Telegram Header Device Code angegeben.

3.2 System Token

Der Token verwaltet die unterschiedlichen Kommunikationsportale (RS 232, RS 422, EFI). Ein Gerät erlaubt keinen zeitgleichen Zugriff von verschiedenen Schnittstellen. Dies wird dadurch sichergestellt, dass vor dem Lesen oder Schreiben der Daten der Token angefordert und erfolgreich zugewiesen worden sein muss.

Falls auf der Messdatenschnittstelle (RS 422) kontinuierliche Datenausgabe erfolgt, kann gleichzeitig über RS 232 oder EFI eine Kommunikation durchgeführt werden. Die kontinuierliche Datenausgabe wird dadurch nicht beeinflusst.

Ausnahme:

Um ein Request-Telegramm bei gleichzeitiger Datenausgabe auf der RS 422 durchzuführen, muss zuerst die kontinuierliche Ausgabe von Messdaten gestoppt werden.

Von einer seriellen Schnittstelle muss immer das angeschlossene Gerät adressiert werden, um den Token zu erhalten. Ohne den Token zugewiesen bekommen zu haben ist es nicht möglich auf Daten eines Geräts zuzugreifen. Der Token wird bei einem Power Reset nicht gespeichert und muss danach neu angefordert werden.

3.3 Request Mode

Im Request Mode werden die Messdaten durch den Hostrechner angefordert (z. B. Block 12 Scandaten). Es werden Send- und Fetch-Telegramme verwendet.

Es können sowohl Messdatenblöcke mit statischer Länge (Block 12) als auch Messdatenblöcke mit flexibler Länge (Block 112) angefordert werden.

Beispiel 1: Es existiert nur ein S3000 und es sollen Messdaten empfangen werden

1. Power On
2. Get Token (Send-Telegramm mit Device Code 0x07)

3. Read Scandatensatz 1 (Fetch-Telegramm mit Device Code 0x07)
4. Read Scandatensatz 2 (Fetch-Telegramm mit Device Code 0x07)
5. Read Scandatensatz n (Fetch-Telegramm mit Device Code 0x07)
6. Release Token (Send-Telegramm mit Device Code 0x07)

Beispiel 2: Es existiert ein S3000 EFI-Verbund und es sollen abwechselnd Messdaten von beiden Scannern empfangen werden. Es werden 2 getrennte RS 422-Verbindungen vorausgesetzt.

1. Power On
2. Get Token S 3000_1 (Send-Telegramm mit Device Code 0x07)
3. Read Scandatensatz von S 3000_1 (Fetch-Telegramm mit Device Code 0x07)
4. Release Token S 3000_1 (Send-Telegramm mit Device Code 0x07)
5. Get Token S 3000_2 (Send-Telegramm mit Device Code 0x08)
6. Read Scandatensatz von S 3000_2 (Fetch-Telegramm mit Device Code 0x08)
7. Release Token S 3000_2 (Send-Telegramm mit Device Code 0x08)
8. Get Token S 3000_1 (Send-Telegramm mit Device Code 0x07)

Beispiel für Request Mode

Get Token:

Hostrechner Send: 00 00 41 44 19 00 00 05 FF 07 19 00 00 05 FF 07 07 0F 9F D0

S 3000 Reply: 00 00 00 00

Read Scandata (Block 12):

Hostrechner Send: 00 00 45 44 0C 00 02 FE FF 07

S 3000 Reply: 00 00 00 00 0C 00 02 FE FF 07 00 08 3B 00 3D 00 29 00 FE E9

Read Extended Scandata (Block 112):

Hostrechner Send: 00 00 45 44 70 00 03 02 FF 07

S 3000 Reply: 00 00 00 00 70 00 03 02 FF 07 01 00 00 00 14 4B 00 26 8B

Release Token:

Hostrechner Send: 00 00 41 44 19 00 00 05 FF 07 19 00 00 05 FF 07 00 00 E7 B8

S 3000 Reply: 00 00 00 00

3.4 Kontinuierliche Datenausgabe

Bei der kontinuierlichen Datenausgabe erfolgt die Ausgabe der Messdaten automatisch, die Telegramme müssen nur gelesen werden.

Die kontinuierliche Datenausgabe wird bei der Konfiguration aktiviert.

Aufbau der kontinuierlichen Datenausgabe :

Die Datenausgabe beginnt immer wie folgt:

00 00 00 00 4 Byte Reply Header

00 00 Datenblocknummer 0x0000 für Datenausgabe

xx xx Größe des Telegramms

FF 07 Coordination Flag und Device Code, hier 0x07

02 01 Protokoll-Versionsnummer 0x0102

0x 00 Status Normal 0x0000 oder Lockout 0x0001

17 01 00 00 Scan-Nummer (Zeitstempel), hier 0x00000117

02 00 Telegramm-Nummer, hier 0x0002

Falls Messdatenausgabe konfiguriert wurde, wird folgender Block mit der Kennung BBBB ausgegeben:

BB BB ID für Messdaten, falls Messdaten konfiguriert

11 11 ID für Messdaten aus Winkelbereich 1

xx xx Messwerte aus Winkelbereich 1

..

Abschließend wird die CRC ausgegeben:

xx xx 16 Bit CRC

Datenformat der Messdaten für S3000 (2 Byte):

Bit 15-13	Bit 12-0
Status Bits Bit 15: Messwert innerhalb simultanem Schutzfeld B erkannt Bit 14: Messwert innerhalb Schutzfeld A erkannt Bit 13: Blendung erkannt	Gemessene Entfernung in Zentimetern

Datenformat der Messdaten für S300 (2 Byte):

Bit 15-13	Bit 12-0
Status Bits Bit 15: Messwert innerhalb Warnfeld erkannt Bit 14: Messwert innerhalb Schutzfeld erkannt Bit 13: Blendung erkannt	Gemessene Entfernung in Zentimetern

Protokoll-Versionsnummer:

S3000: 0x0102

S300: 0x0102 (ab Firmware-Version 1.32)

Die Protokoll -Versionsnummer sollte vom Hostrechner geprüft werden und ist abhängig von der Firmware im Sensor und Gerätetyp (S3000 bzw. S300).

Bei einer anderen Versionsnummer kann der Aufbau der kontinuierlichen Datenausgabe vom Beispiel oben abweichen.

3.5 Beispiel für kontinuierliche Datenausgabe

Konfiguration: Ausgabe aller Messwerte aus einem kompletten Scan

Empfang eines Messdatensatzes:

...

00 00 00 00 00 00 03 04 FF 07 02 01 00 00 17 01 00 00 00 00 BB BB 11 11 E8 03
E8 03 E8 03 E8 03 E8 03 E8 03 E8 03 E8 03 E8 03 E8 03 E8 03 E8 03 ... E8 03 FB B7

...

3.6 Anhalten der kontinuierlichen Datenausgabe

Sendet ein S3000/ S300 kontinuierlich Messdaten, kann nicht gleichzeitig ein Request-Telegramm vom Hostrechner gesendet werden.

Der Hostrechner muss zuerst die kontinuierliche Datenausgabe stoppen. Dies erfolgt durch das Senden eines Zeichens 0x41. Jetzt kann der Hostrechner im Request Mode kommunizieren, solange die konfigurierbare Silent Time noch nicht verstrichen ist. Danach setzt die Ausgabe der Messdaten automatisch wieder ein.

3.7 Fest konfigurierte Modi der kontinuierlichen Datenausgabe

Mit der CDS wird die Messdatenausgabe zunächst fest konfiguriert.

Diese Konfiguration bleibt permanent im Gerät erhalten, solange keine neue Konfiguration an das Gerät übertragen wird.

Bei jedem Power Up wird diese Konfiguration aus dem Speicher des Systemsteckers geladen.

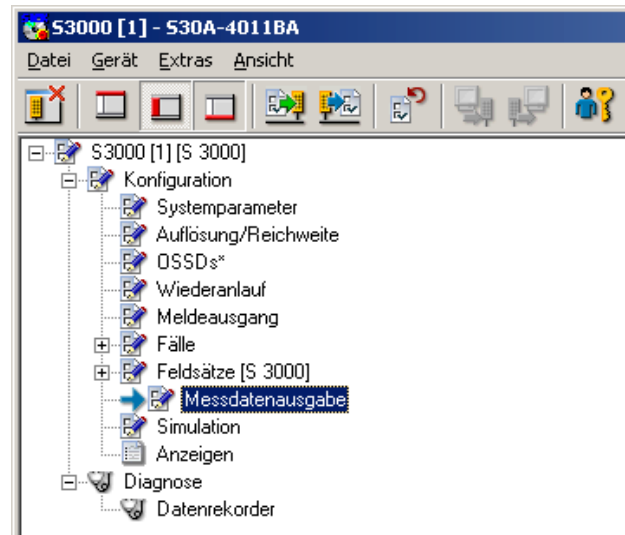
3.8 Zusätzliche Zeitstempel und Telegramm-Nummern

In einem bewegten Transportfahrzeug ist die Zuordnung eines Messwertsatzes über eine Zeitmarke notwendig, damit der Hostrechner die Messwertsätze mit der richtigen Position und der richtigen Orientierung des Fahrzeugs auswerten kann. Deshalb ist im Sensor ein globaler Zähler (32-Bit) realisiert, der beim Block 112 (Request Mode) sowie der kontinuierlichen Messwertausgabe mit ausgegeben wird. Dieser Zähler wird bei jedem Scan, also je nach Konfiguration alle 30 ms oder 60 ms, intern inkrementiert.

Zusätzlich besitzen diese Blöcke je eine eigene Telegramm-Nummer, die nur bei Ausgabe des Blocks inkrementiert wird.

4 Messdatenausgabe

4.1 CDS-Bedienoberfläche



Die *Konfiguration der Messdatenausgabe* erfolgt über das Gerätesymbol S3000 bzw. S300, im Kontextmenü *Messdatenausgabe*.

4.2 Konfiguration der Messdatenausgabe



Baudrate

Die Baudrate der RS422 Schnittstelle lässt sich entsprechend der Menüauswahl einstellen

Silent - Zeit individuell

Die Silent - Zeit legt den zeitlichen Abstand fest, für den nach Aussenden des Silent Bytes, die kontinuierliche Datenausgabe unterbrochen werden kann, um einen Zugriff auf die Schnittstelle zu ermöglichen.

Im Auslieferungszustand ist die Silent Time auf 5000 ms eingestellt.

Sendemodus

Datenausgabe auf Anfrage: Daten werden nur auf Anfrage ausgegeben

Kontinuierliche Datenausgabe: Daten werden permanent ausgegeben

5 Fehlerdiagnose

Für die Diagnose der Einstellungen an Ihrem Sensor stehen Ihnen folgende Möglichkeiten zur Verfügung:

Sie können die Blöcke, die sie online ändern, jederzeit wieder aus dem Sensor zurücklesen, um sicherzustellen, dass die Einstellungen im Gerät korrekt sind.

Mit Hilfe der CDS stehen Ihnen folgende Diagnose-Möglichkeiten zur Verfügung:

- die im Gerät abgespeicherte Konfiguration anzeigen lassen
- im Betriebszustandsbericht die aktuelle Einstellung des Sensors darstellen lassen

6 Glossar

CDS	SICK Configuration & Diagnostic Software
Remote-Betrieb	zwei S3000 bzw. S300 im EFI-Verbund
RK512 Telegram Header	Protokoll-Definition für Datenkommunikation
Device Code	Geräteadresse für Kommunikation
Protokoll-Versionsnummer:	Firmware-/ Geräteerkennung für den Hostrechner
Continuous Mode	permanenter Sende-Modus der Messdaten
Messdaten	vom Sensor gemessene Entfernungswerte
Silent Time	Pausezeit bei kontinuierliche Datenausgabe
Enhanced Function Interface	sichere SICK-Gerätekommunikation
Scandatensatz	Entfernungswerte eines Messumlaufs (0...190° bei S3000, 0...270° bei S300)
Roh-Geschwindigkeitsdaten	aktuell gemessene Inkrementalgeberwerte
Winkelbereich	Segment im Scanfeld
Zeitstempel	Zeitmarke, die bei der Datengenerierung ermittelt wird
Volltreffer	komplette Fläche des Laserstrahls trifft auf ein Objekt

This document is protected by the law of copyright, whereby all rights established therein remain with the company SICK AG. Reproduction of this document or parts of this document is only permissible within the limits of the legal determination of Copyright Law. Alteration or abridgement of the document is not permitted without the explicit written approval of the company SICK AG.

Contents

1	About this document	24
1.1	The function of this document	24
1.2	For whom this document is intended	24
1.3	Depth of information	24
1.4	Scope of validity	24
2	System description	25
2.1	System structure	25
2.2	Electrical interface	26
2.2.1	Transmission and data format	26
2.3	Telegram structure.....	26
2.3.1	The command telegram.....	27
2.3.2	The reply telegram	27
2.3.3	Reply telegram error codes	27
2.4	Standard communication	29
2.5	Flexible telegrams	30
2.6	Continuous output.....	30
2.7	Data confidence	31
3	Types of communication	33
3.1	Device addresses	33
3.2	The system token	33
3.3	Request Mode	33
3.4	Continuous data output	35
3.5	Example of continuous data output.....	36
3.6	Stopping continuous data output.....	37
3.7	Fixed configured continuous data output modes.....	37
3.8	Additional time stamps and telegram numbers.....	37
4	Measuring data output	38
4.1	CDS measurement data configuration	38
4.2	Configuration of measurement data	38
5	Error diagnosis	39
6	Glossary	40
7	Appendix	41

1 About this document

Please read this chapter carefully before you work with the documentation and the S3000 or S300.

1.1 The function of this document

This Telegram Listing Standard describes the measured-data-specific expansion of S3000 functions in combination with the Standard, Advanced, Professional and Remote modules as well as S300 Professional. It is to be considered as a supplement to the S3000 resp. S300 Operating Instructions.



WARNING

Please see the S3000 resp. S300 Operating Instructions, and read them carefully, for general information on, for example, mounting, installing and commissioning the safety laser scanner.

Pay attention on the safety instructions in chapter 2 and 8 before you operate the system for the first time!

The available data communication may not be used for safety purposes!

1.2 For whom this document is intended

This Telegram Listing Standard is intended for system specialists in hardware and software development who want to integrate and evaluate the scanner's own measured value data within their host application.

1.3 Depth of information

This Telegram Listing Standard contains information on the following topics:

- Description of the RS-422 interface
- Description of the RK512 protocol used
- Description of special scanner-specific functions
- Information on use
- Error diagnosis

1.4 Scope of validity

This Telegram Listing Standard is applicable for the laser scanner S3000 Standard, Advanced, Professional and Remote or for the laser scanner S300 Professional with the following type labels

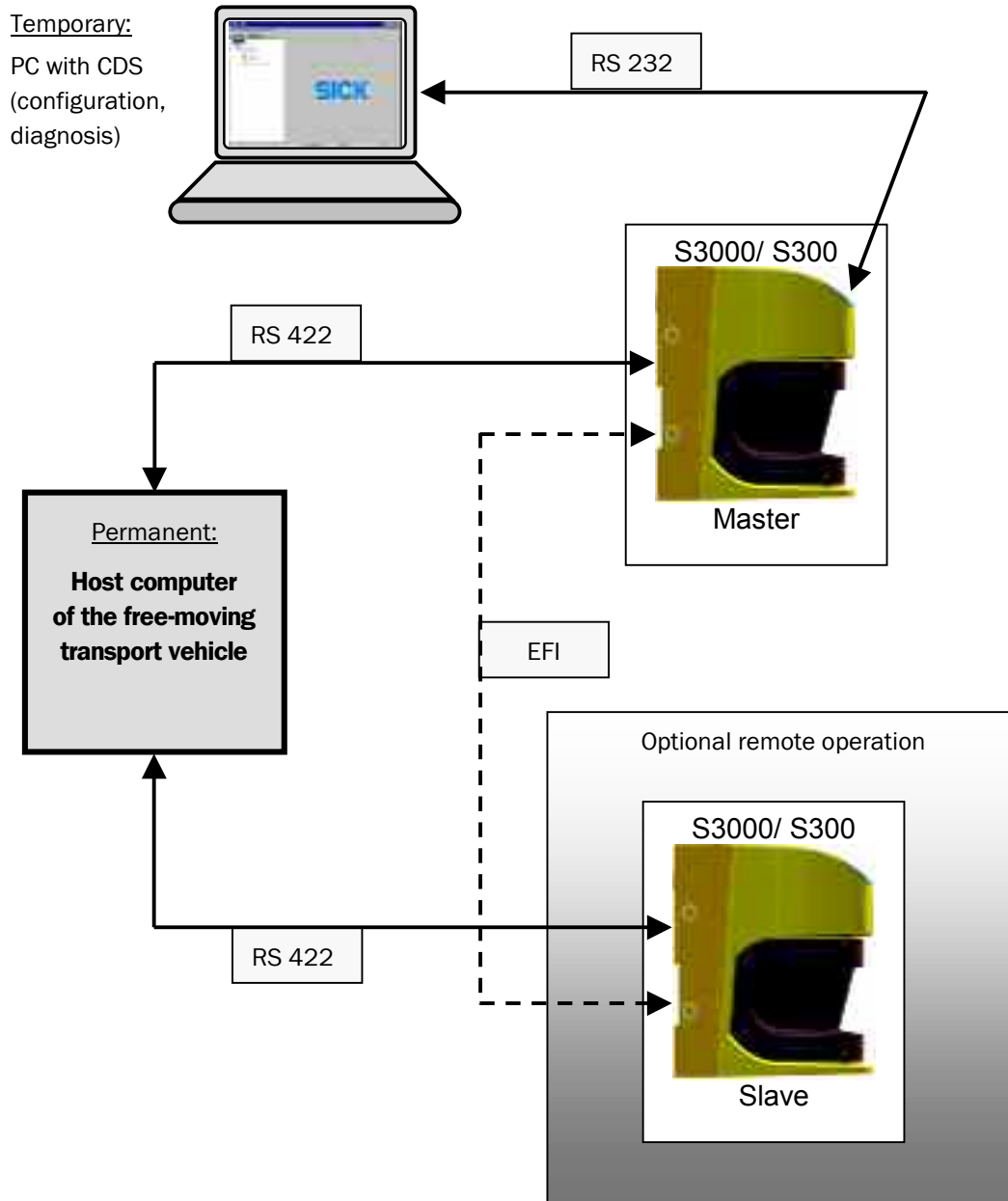
S3000: S30A-XXXX XA

S300: S30B-XXXX XA.

.

2 System description

2.1 System structure



2.2 Electrical interface

The electrical interface is implemented according to the EIA RS-422-A Standard. Electrical connection is described in the “Electrical installation” chapter of the S3000/S300 Operating Instructions.

2.2.1 Transmission and data format

The baud rate of RS-422 measurement data output can be selected with the CDS from among the following baud rates:

- 9600 baud
- 19200 baud
- 38400 baud
- 115.2 kbaud (S300 only)
- 125 Kbaud
- 230.4 kbaud (S300 only)
- 250 Kbaud
- 460.8 kbaud (S300 only)
- 500 Kbaud

In the default state the interface is pre-configured with a baud rate of 125 Kbaud.

One data byte consists of 1 start bit, 8 data bits, 1 stop bit, no parity bit.

2.3 Telegram structure

Communication according to the RK512 standard is based on “command” and “reply” telegrams. A command telegram is either a “send” or a “fetch” telegram.

The host computer transmits send telegrams with the data to be transferred after the telegram header; the recipient answers with a reply telegram without any further data. The host computer transmits fetch telegrams with the header of a fetch telegram without any subsequent data, and the sensor responds with a reply telegram which contains the requested data after the telegram header.

The host computer is also always the active participant in EFI combinations. The sensors do not transmit any RK512 telegrams on their own account. This prevents any possible initialisation conflicts arising from simultaneous communication.

The value in the size field always describes the number of 16-bit words; individual bytes cannot be accessed.

In the telegram header, values that consist of 2 bytes are transmitted with the high byte (HB) first. In the case of data bytes, the 16-bit words with the low byte (LB) are transferred first.

2.3.1 The command telegram

The telegram header of the command telegram consist of 10 bytes with the following meanings:

Byte	Telegram fields	Content	Meaning
1	Telegram identifier	0x00	
2		0x00	
3	Command telegram type	'A' (0x41) or 'E' (0x45)	Send telegram or fetch telegram
4	Command data type	'D' (0x44)	
5	Destination address/ source address	0x00 to 0xFF	Data block number
6		0x00	
7	Size	0 to 65535	Block size in word
8			
9	Co-ordination flag (byte number)	0xFF	
10	Device address	0 to 15	0x07 S 3000_1 (S 300_1) 0x08 S 3000_2 (S 300_2)

Table: Structure of command telegram

2.3.2 The reply telegram

The telegram header of the reply telegram consists of 4 bytes with the following meanings:

Byte	Telegram fields	Content	Meaning
1	Telegram identifier	0x00 (always)	
2		0x00 (always)	
3	Reply telegram type	0x00 (always)	Reply telegram
4	Reply error number	0x00 0x01 to 0xFF	No error See Error Table for error numbers

Table: Structure of reply telegram

2.3.3 Reply telegram error codes

The reply telegram is the answer of the S3000/ S300 to a send or fetch telegram.

If the S3000/ S300 detects an error it shows this in the error number of the reply telegram, and no data is sent after the reply telegram header.

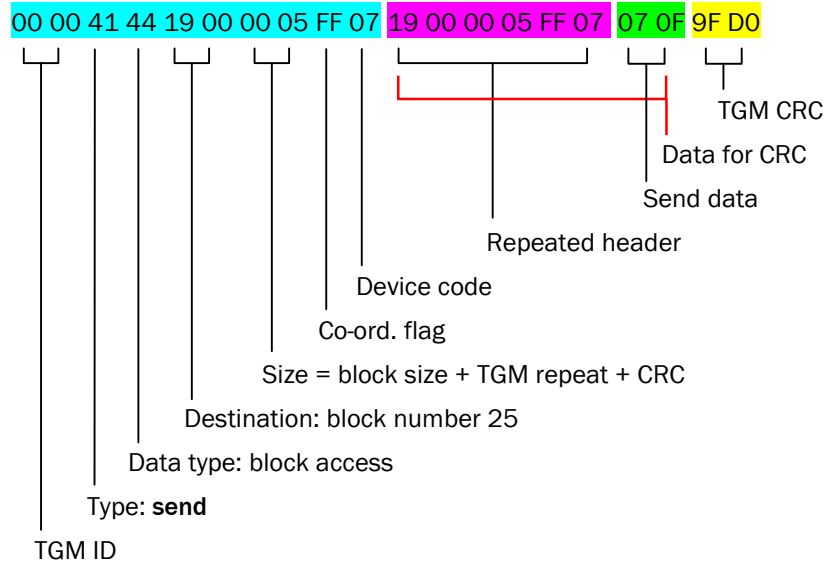
Error code in reply telegram	RK512 protocol communication error
0x00	No error
0x0A	One of the communication monitoring processes failed (e.g. timeout of EFI-RK512 packages / EFI-RK512 acknowledgement or failed transmission of EFI-RK512 packages / EFI-RK512 acknowledgement)
0x0C	The data word number of the destination address or source address in command telegram (byte 6) is impermissible (not defined in Interface Register)
	The co-ordination flag (byte number) in command telegram (byte 9) does not equal 0xFF
	The device code in the command telegram (byte 10, bits 0 to 3) is invalid (i.e. equals 0)
	The CPU number in the command telegram (byte 10, bits 5 to 7) is impermissible
0x10	The telegram identifier in the command telegram (byte 1) is not equal to 0x00 or 0xFF or is not followed by a further 0x00 byte (byte 2)
	The command data type in the command telegram (byte 4) is impermissible
0x14	The data block number of the destination address or source address in the command telegram (byte 5) is impermissible (not defined in the Interface Register)
0x16	The command telegram type in the command telegram (byte 3) is impermissible
0x34	Telegram format error Possible causes: <ul style="list-style-type: none"> • The length quoted in the size field of the command telegram (bytes 7 and 8) exceeds the number of entries after the defined address in the data block • The length of the send telegram data was greater than that defined in the size field • The length of the send telegram data was smaller than that defined in the size field • A fetch telegram with useful data has been received • The send telegram was correctly received but the CRC on the data is wrong or the first six data bytes do not match bytes 5 to 10 from the telegram header
0x36	A command telegram has been received though no reply telegram has been received yet
0x01	The current device status does not permit access to the data block
0x02	Access to the data block by the current user group is not permitted
0x03	Incorrect password
0x04	System token is occupied
0x05	Incorrect parameter

Table: Reply telegram error codes

2.4 Standard communication

Example of a send telegram: "Write the token in block 25"

Host computer sends:

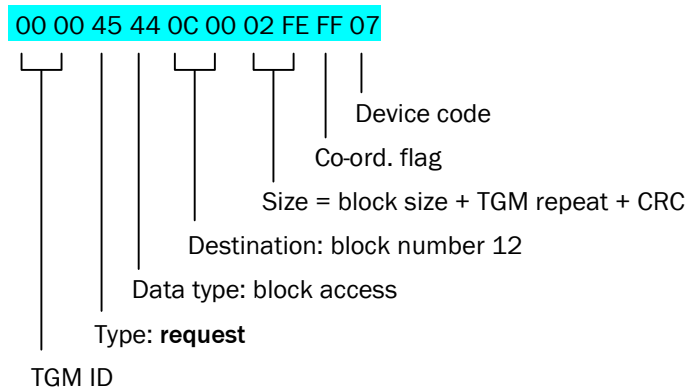


S3000 reply:

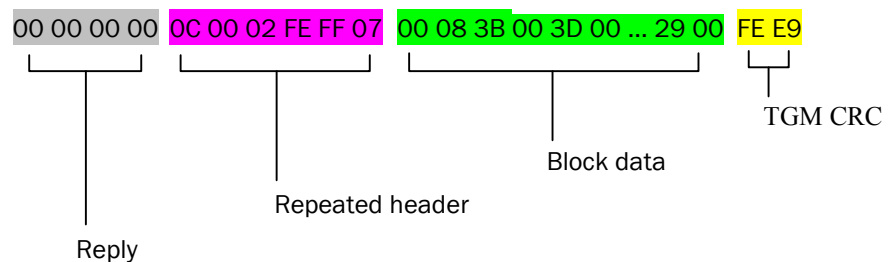
00 00 00 00

Example of a fetch telegram: "Read operating data block 12"

Host computer sends:



S3000 reply:



2.5 Flexible telegrams (S3000 only)

Some host computers require the interrupt burden for the UART to be kept as low as possible. In such cases (Request Mode) a data block of flexible length can be used in the reply telegram (block 112).

Please note In EFI combination, block 112 is only available via a sensor connection directly at the local RS-422 port.

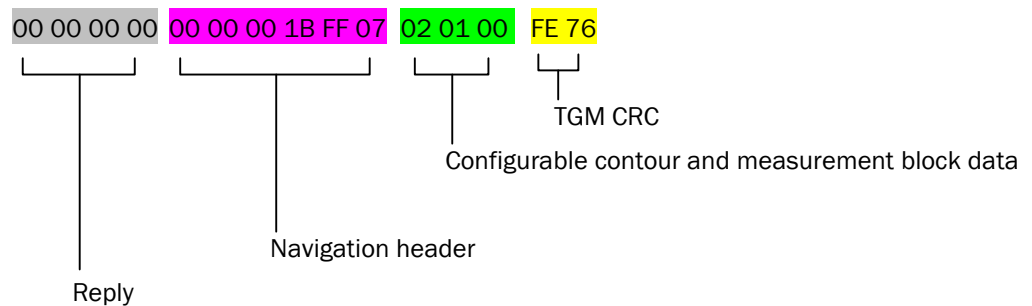
2.6 Continuous output

In order to be able to build up telegram traffic that is as efficient as possible, the S3000/S300 can be configured in such a way that it permanently transmits the measured values and some additional information at the RS-422 interface. Continuous output is not affected by telegram traffic via the other interfaces available. It is not necessary to reserve the token for the RS-422 interface for continuous data output (in Request Mode only one interface can possess the token at any time).

Continuous data output must first be stopped (see Chap. 3.6: Stopping continuous data output) in order to be able to transmit a request telegram from the host computer via the RS-422 interface during continuous data output (e.g. changing Trigger Modes).

Example: Continuous data output telegram structure

```
00 00 00 00 00 00 00 00 1B FF 07 02 01 00 00 17 01 00 00 00 00
CC CC 08 00 00 00 00 E8 23 19 00 E8 23 32 00 E8 23 7D 00 E8 23 96 00 E8 23 C8 00 E8
23 E1 00 E8 23 38 4A E8 23 FE 76
```



Please note In EFI combination, continuous output of measured values is only possible directly via one of the sensors connected to the RS 422.

2.7 Data confidence

As the RK512 standard does not offer any mechanism for checking the integrity of the data received, such a mechanism is supplemented in the data of the RK512 telegram. This involves the exact repetition of bytes 5 to 10 from the header of a command telegram in the first six data bytes and a CRC that is formed via the data bytes and added to them. Accordingly, the number of words defined in the size field of the telegram header of a command telegram is raised by 4 words.

The CRC is 16 bits and is formed according to the polynomial $x^{16}+x^{12}+x^5+x^0$ (0x1021). This CCITT-CRC is applied in the entire communication stack where a CRC is required for safeguarding the data transmitted.

A simple routine for CRC calculation follows as an example:

```

static const unsigned short crc_table[256] = {
0x0000, 0x1021, 0x2042, 0x3063, 0x4084, 0x50a5, 0x60c6, 0x70e7,
0x8108, 0x9129, 0xa14a, 0xb16b, 0xc18c, 0xd1ad, 0xe1ce, 0xf1ef,
0x1231, 0x0210, 0x3273, 0x2252, 0x52b5, 0x4294, 0x72f7, 0x62d6,
0x9339, 0x8318, 0xb37b, 0xa35a, 0xd3bd, 0xc39c, 0xf3ff, 0xe3de,
0x2462, 0x3443, 0x0420, 0x1401, 0x64e6, 0x74c7, 0x44a4, 0x5485,
0xa56a, 0xb54b, 0x8528, 0x9509, 0xe5ee, 0xf5cf, 0xc5ac, 0xd58d,
0x3653, 0x2672, 0x1611, 0x0630, 0x76d7, 0x66f6, 0x5695, 0x46b4,
0xb75b, 0xa77a, 0x9719, 0x8738, 0xf7df, 0xe7fe, 0xd79d, 0xc7bc,
0x48c4, 0x58e5, 0x6886, 0x78a7, 0x0840, 0x1861, 0x2802, 0x3823,
0xc9cc, 0xd9ed, 0xe98e, 0xf9af, 0x8948, 0x9969, 0xa90a, 0xb92b,
0x5af5, 0x4ad4, 0x7ab7, 0x6a96, 0x1a71, 0x0a50, 0x3a33, 0x2a12,
0xdbfd, 0xcdbc, 0xfbff, 0xeb9e, 0x9b79, 0x8b58, 0xbb3b, 0xab1a,
0x6ca6, 0x7c87, 0x4ce4, 0x5cc5, 0x2c22, 0x3c03, 0x0c60, 0x1c41,
0xedae, 0xfd8f, 0xcdec, 0xddcd, 0xad2a, 0xbd0b, 0x8d68, 0x9d49,
0x7e97, 0x6eb6, 0x5ed5, 0x4ef4, 0x3e13, 0x2e32, 0x1e51, 0x0e70,
0xff9f, 0xefbe, 0xdfdd, 0xcffc, 0xbf1b, 0xaf3a, 0x9f59, 0x8f78,
0x9188, 0x81a9, 0xb1ca, 0xa1eb, 0xd10c, 0xc12d, 0xf14e, 0xe16f,
0x1080, 0x00a1, 0x30c2, 0x20e3, 0x5004, 0x4025, 0x7046, 0x6067,
0x83b9, 0x9398, 0xa3fb, 0xb3da, 0xc33d, 0xd31c, 0xe37f, 0xf35e,
0x02b1, 0x1290, 0x22f3, 0x32d2, 0x4235, 0x5214, 0x6277, 0x7256,
0xb5ea, 0xa5cb, 0x95a8, 0x8589, 0xf56e, 0xe54f, 0xd52c, 0xc50d,
0x34e2, 0x24c3, 0x14a0, 0x0481, 0x7466, 0x6447, 0x5424, 0x4405,
0xa7db, 0xb7fa, 0x8799, 0x97b8, 0xe75f, 0xf77e, 0xc71d, 0xd73c,
0x26d3, 0x36f2, 0x0691, 0x16b0, 0x6657, 0x7676, 0x4615, 0x5634,
0xd94c, 0xc96d, 0xf90e, 0xe92f, 0x99c8, 0x89e9, 0xb98a, 0xa9ab,
0x5844, 0x4865, 0x7806, 0x6827, 0x18c0, 0x08e1, 0x3882, 0x28a3,
0xcb7d, 0xdb5c, 0xeb3f, 0xfb1e, 0x8bf9, 0x9bd8, 0xabbb, 0xbb9a,
0x4a75, 0x5a54, 0x6a37, 0x7a16, 0x0af1, 0x1ad0, 0x2ab3, 0x3a92,
0xfd2e, 0xed0f, 0xdd6c, 0xcd4d, 0xbdaa, 0xad8b, 0x9de8, 0x8dc9,
0x7c26, 0x6c07, 0x5c64, 0x4c45, 0x3ca2, 0x2c83, 0x1ce0, 0x0cc1,
0xef1f, 0xff3e, 0xcf5d, 0xdf7c, 0xaf9b, 0xbfba, 0x8fd9, 0x9ff8,
0x6e17, 0x7e36, 0x4e55, 0x5e74, 0x2e93, 0x3eb2, 0x0ed1, 0x1ef0
};
WORD CRC16 (BYTE *Data, DWORD length)
{
    WORD CRC_16 = 0xFFFF;
    WORD i;
    for (i = 0; i < length; i++)
    {
        CRC_16 = (CRC_16 << 8) ^ (crc_table[(CRC_16 >> 8) ^ (Data[i])]);
    }
    return CRC_16;
}

```

3 Types of communication

3.1 Device addresses

The communications addresses for S 3000_1 and S 3000_2 (the device code) are necessary for device communications in Remote Mode.

The device address must be 0x07 for stand-alone applications and applications with the UE100 and UE1000 family.

The addresses are assigned as follows in Remote Mode:

S 3000_1 address 0x07

S 3000_2 address 0x08

These addresses are defined in the RK512 telegram header device code.

3.2 The system token

The token administrates the various communication portals (RS 232, RS 422, EFI). A device does not permit simultaneous access to different interfaces. The token is demanded and must be successfully allocated before any data can be read or written.

If continuous data output is taking place at the measurement data interface (RS 422), communication can take place simultaneously via the RS 232 or EFI. Continuous data output is not affected by this.

Exception:

In order to carry out a request telegram during simultaneous data output at the RS 422, the continuous output of measurement must be stopped first.

The connected device must always be addressed from a serial interface in order to receive the token. Without having been assigned the token, it is impossible to access the data of a device. The token is not saved during a power reset and must be requested again thereafter.

3.3 Request Mode

In Request Mode the measurement data is requested by the host computer (e.g. block 12 scan data). Send and fetch telegrams are used.

Measurement blocks with static lengths (block 12) and measurement blocks with flexible lengths (block 112) can be requested.

Example1: There is only one S3000 and measurement data is to be received

1. Power on
2. Get token (send telegram with device code 0x07)
3. Read scan data set 1 (fetch telegram with device code 0x07)

4. Read scan data set 2 (fetch telegram with device code 0x07)
5. Read scan data set n (fetch telegram with device code 0x07)
6. Release token (send telegram with device code 0x07)

Example 2: There is an S3000 EFI combination and measurement data from both scanners is to be received alternately. 2 separate RS 422 connections are a prerequisite.

1. Power on
2. Get token S 3000_1 (send telegram with device code 0x07)
3. Read scan data set from S 3000_1 (fetch telegram with device code 0x07)
4. Release token S 3000_1 (send telegram with device code 0x07)
5. Get token S 3000_2 (send telegram with device code 0x08)
6. Read scan data set from S 3000_2 (fetch telegram with device code 0x08)
7. Release token S 3000_2 (send telegram with device code 0x08)
8. Get token S 3000_1 (send telegram with device code 0x07)

Example for Request Mode

Get token:

Host computer, send: 00 00 41 44 19 00 00 05 FF 07 19 00 00 05 FF 07 07 0F 9F D0

S3000, reply: 00 00 00 00

Read scan data (block 12):

Host computer, send: 00 00 45 44 0C 00 02 FE FF 07

S3000, reply: 00 00 00 00 0C 00 02 FE FF 07 00 08 3B 00 3D 00 29 00 FE E9

Read extended scan data (block 112):

Host computer, send: 00 00 45 44 70 00 03 02 FF 07

S3000, reply: 00 00 00 00 70 00 03 02 FF 07 01 00 00 00 14 4B 00 26 8B

Release token:

Host computer, send: 00 00 41 44 19 00 00 05 FF 07 19 00 00 05 FF 07 00 00 E7 B8

S3000, reply: 00 00 00 00

3.4 Continuous data output

During continuous data output, the output of measurement data takes place automatically, the telegrams only need to be read.

Continuous data output is activated during configuration.

Structure of continuous data output:

data output always starts as follows:

00 00 00 00 4 byte reply header

00 00 data block number 0x0000 for data output

xx xx size of telegram

FF 07 co-ordination flag and device code, here 0x07

02 01 protocol version number 0x0102

0x 00 status normal 0x0000 or lockout 0x0001

17 01 00 00 scan number (time stamp), here 0x00000117

02 00 telegram number, here 0x0002

The following block measurement data with the BBBB flag is output if measurement data output is configured:

BB BB ID for measurement data, if measurement data configured

11 11 ID for measurement data from angular range 1

xx xx measured values from angular range 1

...

The CRC is output in conclusion:

xx xx 16 bit CRC

3.6 Stopping continuous data output

A request telegram cannot be sent from the host computer while an S3000/ S300 is transmitting continuous measurement data.

The host computer must first stop the continuous data output. This takes place through the transmission of a character 0x41. Now the host computer can communicate in Request Mode, as long as the configurable silent time has not yet elapsed. Thereafter the output of measurement data automatically restarts.

3.7 Fixed configured continuous data output modes

Measurement data output is initially configured as fixed with the CDS.

This configuration is permanently retained in the device, as long as no new configuration is transferred to the device.

This configuration is loaded from the system plug's memory at each Power Up.

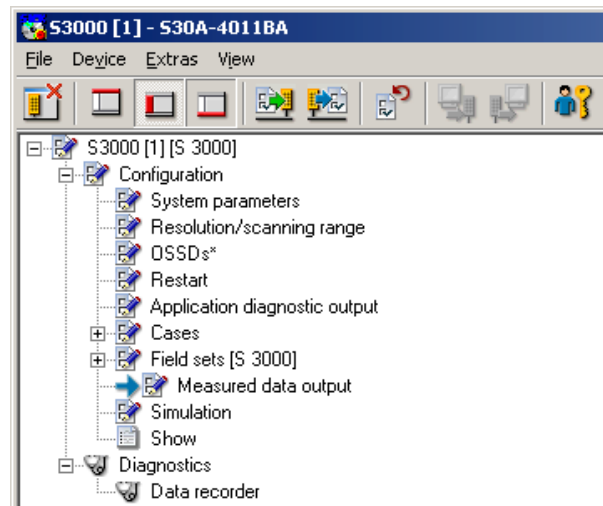
3.8 Additional time stamps and telegram numbers

In a moving transport vehicle, the allocation of a measured value set with a time stamp for the host computer is necessary so that the measured value set is evaluated with the correct position and the correct orientation of the vehicle. For this reason, there is a global counter in the sensor (32-bit) that is output with block 112 (Request Mode) in addition to the continuous measured value output. This counter is internally incremented at each scan, i.e. every 30 ms or 60 ms depending on the configuration.

In addition, these blocks each have their own telegram number that is only incremented on output of the block.

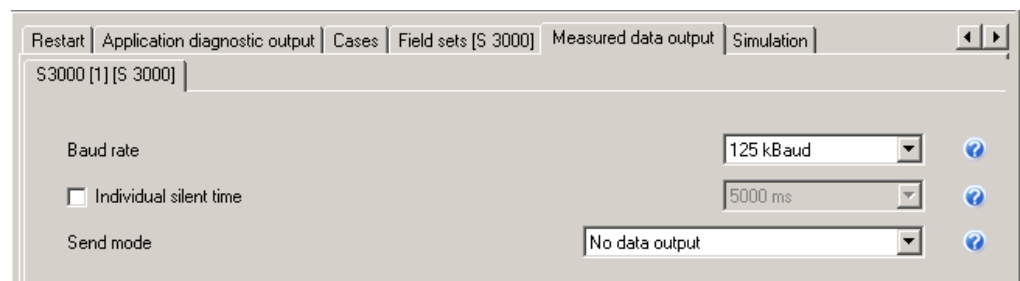
4 Measuring data output

4.1 CDS measurement data configuration



Configuration of measuring data output takes place via the S3000/ S300 device symbol in the Measuring data contextual menu.

4.2 Configuration of measurement data



Baud rate

The baud rate of the RS-422 interface can be set within the Baud rate box!

Individual Silent time

The silent time defines the length of the time period for transmitting the silent bytes, with which continuous data output can be interrupted to allow access to the interface.

The default silent time is 5000 ms.

Send mode

Data output on request: data is only provided on request

Continuous data output: data is provided continuously

5 Error diagnosis

The following possibilities are available for the diagnosis of settings on your sensor:

You can read blocks, that you have changed online, back from the sensor at any time to ensure that the settings in the device are correct.

The following diagnostic possibilities are available with the help of CDS:

- you can display the configuration saved in the device
- you can show the current setting of the sensor in the operating state report

6 Glossary

CDS	SICK Configuration & Diagnostic Software
Remote Mode	Two S3000/ S300 in an EFI combination
RK512 telegram header	Protocol definition for data communication
Device code	Device address for communication
Protocol version number	Firmware/ Device identification to the host computer
Continuous Mode	Permanent measurement data transmission mode
Measurement data	Distance values measured by sensor
Silent time	Pause period during continuous data output
Enhanced function interface	Safe SICK device communication
Scan data set	Distance values of a measurement range (0...190° for S3000; 0...270° for S300)
Raw speed data	Shaft encoder value currently measured
Angular range	Segment in the scanning field
Time stamp	A time mark that is determined on generation of data
Bull's eye	Complete area of the laser beam hits an object

7 Appendix

7.1 Description of the data blocks used

7.1.1 Scan Data Block (Block No. 12) for S300

Block Description

Data Block Name	Block No.	Block Size	Access		Non-volatile
			S 300	Ext. Device	
Scan Data Block	12	1084 Byte	W	R 0-4	No
	Block Registers	Block Words	Register Size	RI Identifier	
	Monitoring Data	0	2 Byte	RI_SCAN_STATUS	
	Scan Data Pulse 1-542	1-542	1082 Byte	RI_SCAN_DATA	

Register Description

Register Name	Block Word	Field Bits	Field Name	Field Description
Monitoring Data	0	0-3	Monitoring Case	0-3: Active Monitoring Case
		4-7	n/u	n/u
		8-10	Control Area	0-3: Monitoring Area
		11-15	n/u	n/u
Scan Data Pulse 1	1	0-12	Distance	Distance [cm]
		13-15	Status Flags	Bit 15:n/u Bit 14: Pulse within Control Area detected Bit 13: glare (dazzling) detected
Scan Data Pulse 2-542	2-542	See Scan Data Pulse 1		

7.1.2 Scan data block (block no. 12) for S3000**Block description**

Data Block Name	Block No.	Block Size	Access		Non-volatile
			S3000	Ext. Device	
Scan data block	12	1524 Byte	W	R 0-4	No
	Block Registers	Block Words	Register Size	RI Identifier	
	Monitoring data	0	2 Byte	RI_SCAN_STATUS	
	Scan data pulse 1-761	1-762	1522 Byte	RI_SCAN_DATA	

Register description

Register Name	Block Word	Field Bits	Field Name	Field Description
Monitoring data	0	0-3	Monitoring case	0-15: Active monitoring case
		4-7	n/u	n/u
		8-10	Control area A	0-7: Monitoring area of control area A
		11	Control area A activated	0: Control area A inactive 1: Control area A active
		12-14	Control area B	0-7: Monitoring area of control area B
		15	Control area B activated	0: Control area B inactive 1: Control area B active
Scan data pulse 1	1	0-12	Distance	Distance [cm]
		13-15	Status flags	Bit 15: Pulse within control area B detected Bit 14: Pulse within control area A detected Bit 13: glare (dazzling) detected
Scan data pulses 2-761	2-762	See scan data pulse 1		

7.1.3 Configuration master block (block no. 25)

Block description

Data Block Name	Block No.	Block Size	Access		Non-volatile
			S3000	Ext. Device	
Config. master block	25	2 Byte	R	R 0-4	No
			W	W 0-4	
			Block Registers	Block Words	
Master ID register	0	2 Byte	RI_MSTR_ID		

Register description

Register Name	Block Word	Field Bits	Field Name	Field Description
Master ID register	0	0-3	Device address	Device address that passed request to host of master token: 0x7: S 3000_1 0x8: S 3000_2 Other: prohibited
		4-7	n/u	n/u
		8-11	Token requester	0xF: PC 7 host PC Other: prohibited
		12-15	Interface of token requester	Valid only for PC as token requester 0x0: External serial interface 0x1: Navigation data interface Other: prohibited

Remark:

By using the physical device address this block can be accessed without being assigned the token (only read access).

Write 0x0000 to the master ID register to return the token.

7.14 Extended scan data block (block no. 112, S3000 only)**Block description**

Data Block Name	Block No.	Block Size	Access		Non-volatile
			S3000	Ext. Device	
Extended scan data block	112	772 or 1532 Byte	W	R 0-4	No
	Block Registers	Block Words	Register Size	RI Identifier	
	Telegram number	0-1	4 Byte	RI_EX_SCAN_TELEGRAM_NO	
	Scan number	2-3	4 Byte	RI_EX_SCAN_SCAN_NO	
	Monitoring data	4	2 Byte	RI_EX_SCAN_STATUS	
	Scan data pulse 1-381 or 1-761	5-765	762 or 1522 Byte	RI_EX_SCAN_DATA	

Register description

Register Name	Block Word	Field Bits	Field Name	Field Description
Telegram number	0-1	0-31	Telegram number	Current telegram number
Scan number	2-3	0-31	Scan number	Scan number since power-up
Monitoring data	4	0-3	Monitoring case	0-15: Active monitoring case
		4-7	n/u	n/u
		8-10	Control area A	0-7: Monitoring area of control area A
		11	Control area A activated	0: Control area A inactive 1: Control area A active
		12-14	Control area B	0-7: Monitoring area of control area B
		15	Control area B activated	0: Control area B inactive 1: Control area B active
Scan data pulse 1	5	0-12	Distance	Distance [cm]
		13-15	Status flags	Bit 15: Pulse detected within control area B Bit 14: Pulse detected within control area A Bit 13: glare (dazzling) detected
Scan data pulse 2-381 or 2-761	5-385 or 5-765	See scan data pulse 1		

Remark:

Telegram size depends on scan resolution.

Australia

Phone +61 3 9497 4100
1800 33 48 02 - tollfree
E-Mail sales@sick.com.au

Belgium/Luxembourg

Phone +32 (0)2 466 55 66
E-Mail info@sick.be

Brasil

Phone +55 11 5091-4900
E-Mail sac@sick.com.br

Ceská Republika

Phone +420 2 57 91 18 50
E-Mail sick@sick.cz

China

Phone +852-2763 6966
E-Mail ghk@sick.com.hk

Danmark

Phone +45 45 82 64 00
E-Mail sick@sick.dk

Deutschland

Phone +49 (0)2 11 53 01-260
E-Mail info@sick.de

España

Phone +34 93 480 31 00
E-Mail info@sick.es

France

Phone +33 1 64 62 35 00
E-Mail info@sick.fr

Great Britain

Phone +44 (0)1727 831121
E-Mail info@sick.co.uk

India

Phone +91-22-2822 7084
E-Mail info@sick-india.com

Italia

Phone +39 02 27 40 93 19
E-Mail info@sick.it

Japan

Phone +81 (0)3 3358 1341
E-Mail info@sick.jp

Nederlands

Phone +31 (0)30 229 25 44
E-Mail info@sick.nl

Norge

Phone +47 67 81 50 00
E-Mail austefjord@sick.no

Österreich

Phone +43 (0)22 36 62 28 8-0
E-Mail office@sick.at

Polska

Phone +48 22 837 40 50
E-Mail info@sick.pl

Republic of Korea

Phone +82-2 786 6321/4
E-Mail kang@sickkorea.net

Republika Slovenija

Phone +386 (0)1-47 69 990
E-Mail office@sick.si

Russia

Phone +7 95 775 05 30
E-Mail info@sick-automation.ru

Schweiz

Phone +41 41 619 29 39
E-Mail contact@sick.ch

Singapore

Phone +65 6744 3732
E-Mail admin@sicksgp.com.sg

Suomi

Phone +358-9-25 15 800
E-Mail sick@sick.fi

Sverige

Phone +46 8 680 64 50
E-Mail info@sick.se

Taiwan

Phone +886 2 2365-6292
E-Mail sickgrc@ms6.hinet.net

Türkiye

Phone +90 216 587 74 00
E-Mail info@sick.com.tr

USA/Canada/México

Phone +1(952) 941-6780
1 800-325-7425 - tollfree
E-Mail info@sickusa.com

More representatives and agencies
in all major industrial nations at
www.sick.com